Primes & Secrets

A gentle introduction to public-key cryptography

Alberto Trombetta **Insubria University**

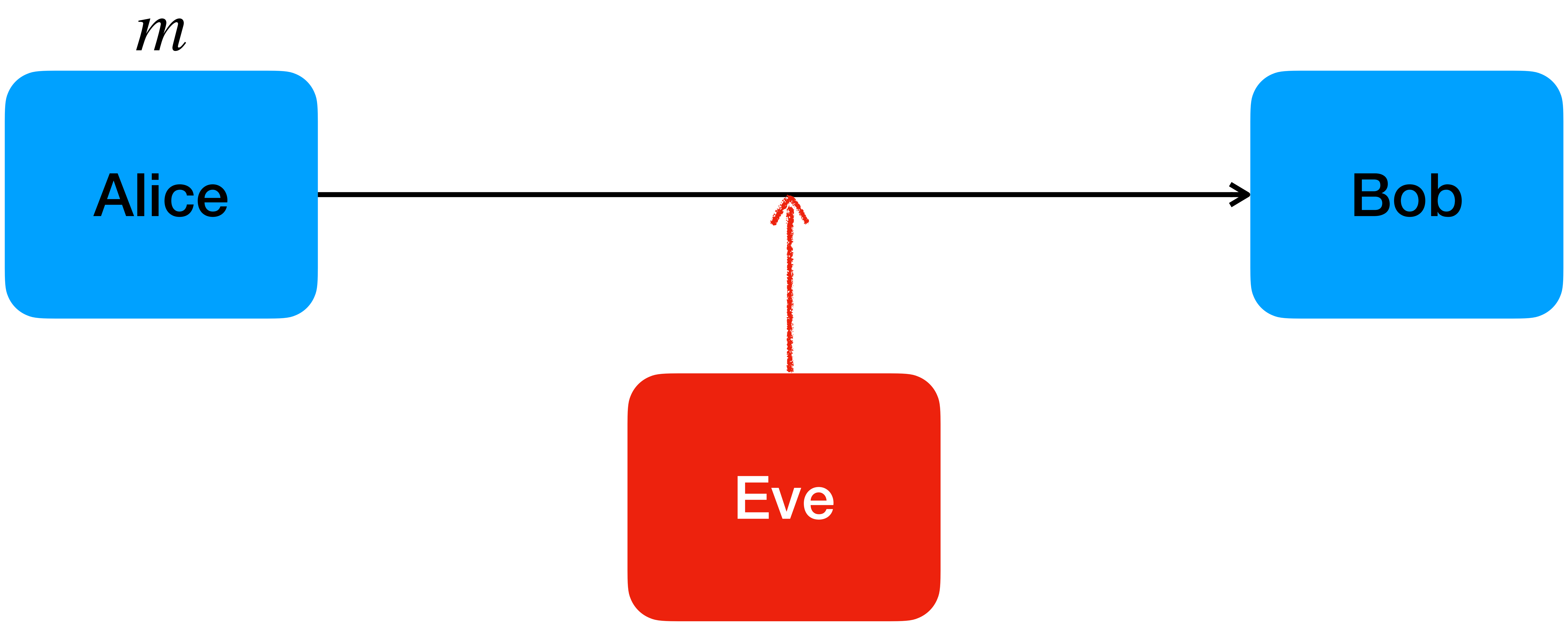
Introduction

The Plan

- What is public-key cryptography
- Some tools from Number Theory
- The Rivest-Shamir-Adleman encryption scheme
- Uses of the RSA scheme

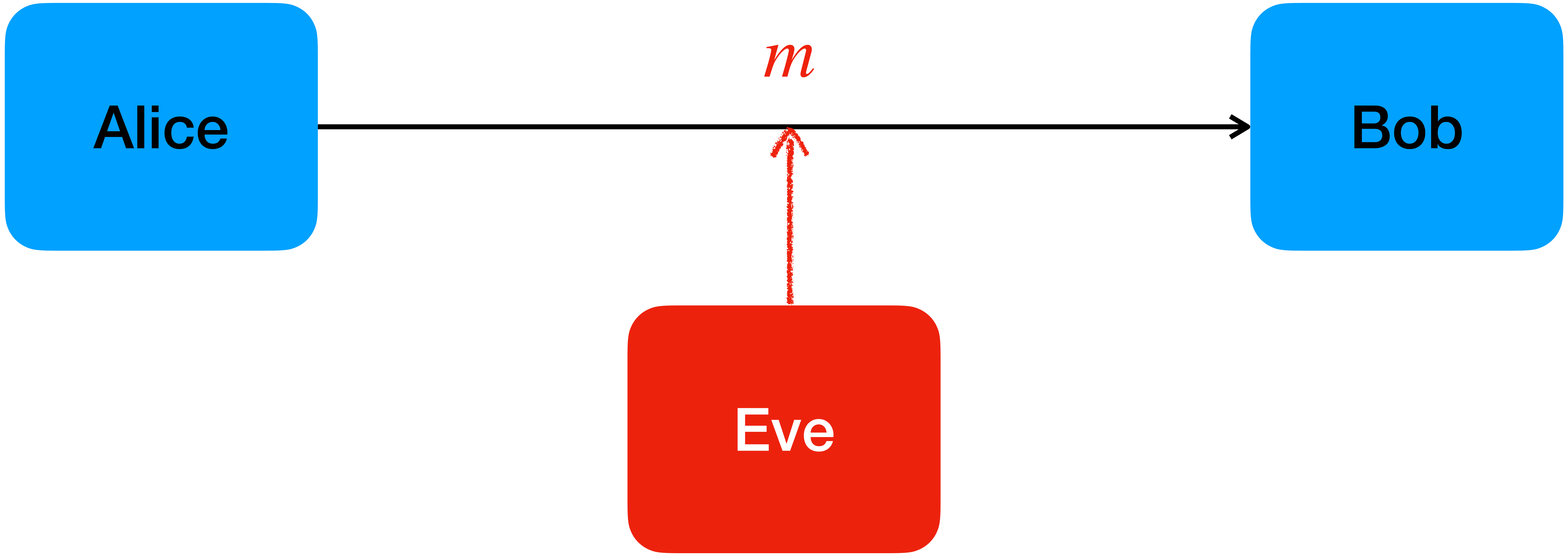
Symmetric encryption

Alice, Bob, etc.



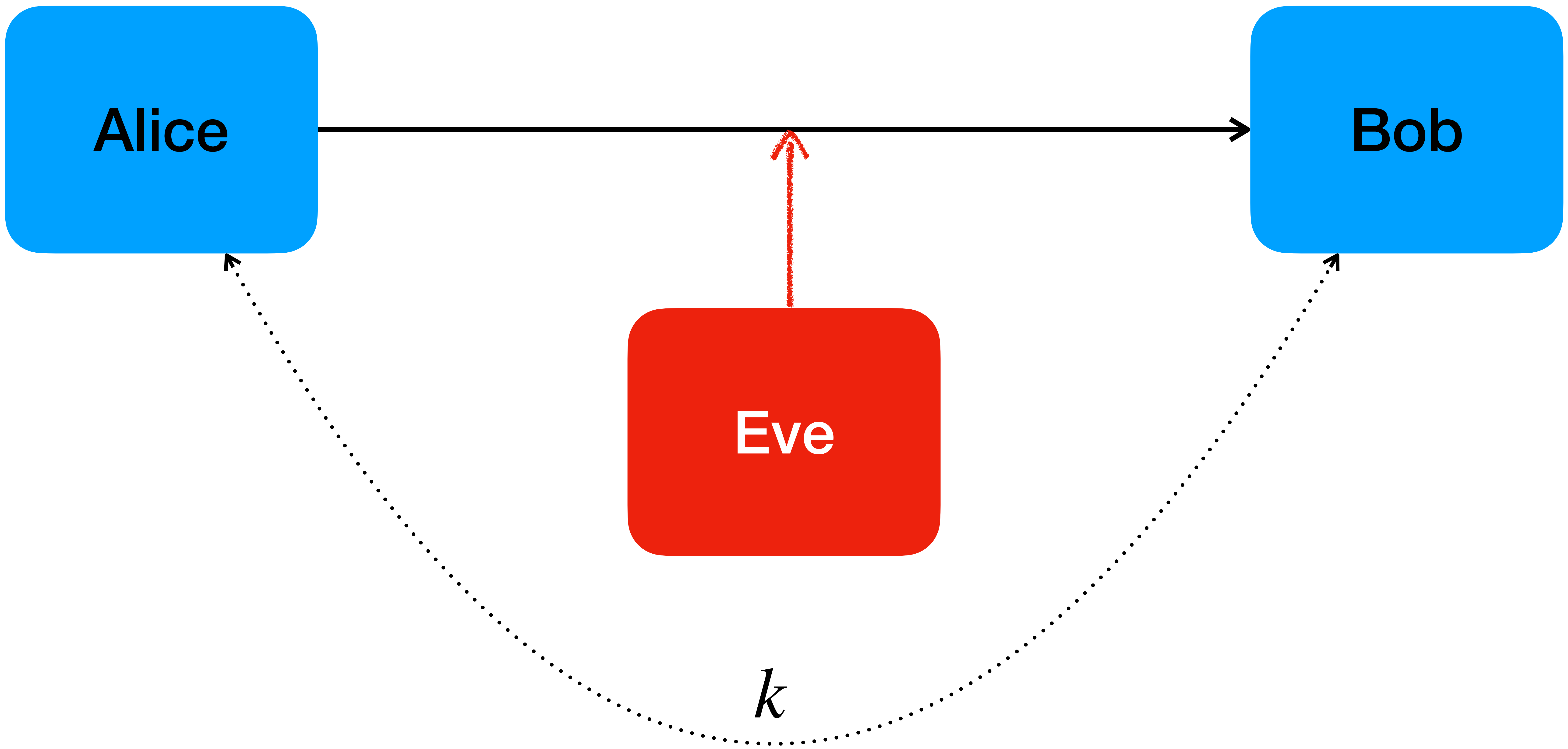
Symmetric encryption

Alice, Bob, etc.



Symmetric encryption

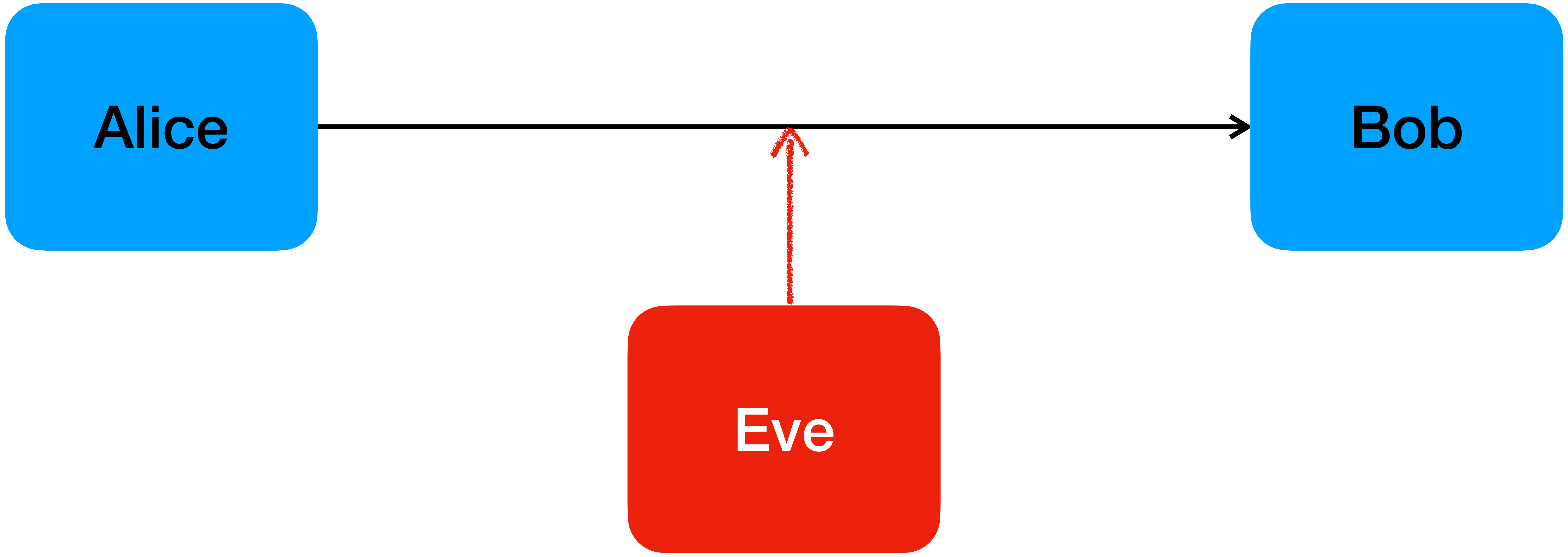
Alice, Bob, etc.



Symmetric encryption

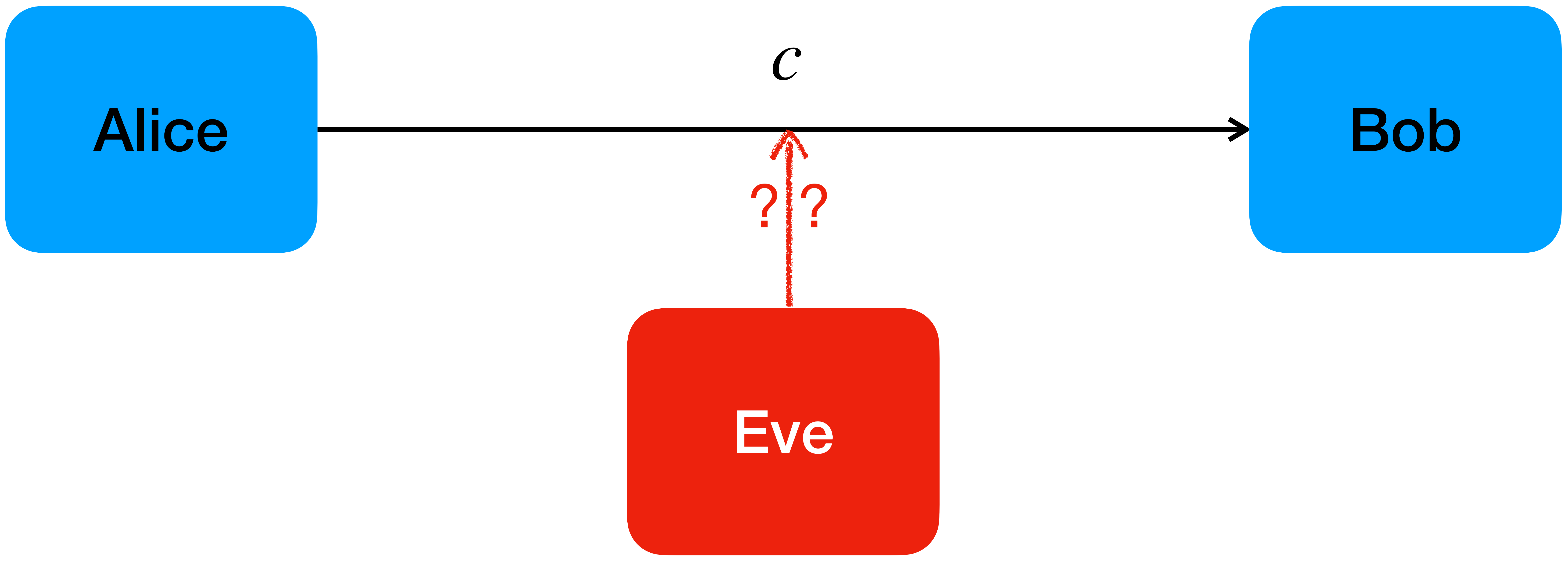
Alice, Bob, etc.

$$c \leftarrow Enc_k(m)$$



Symmetric encryption

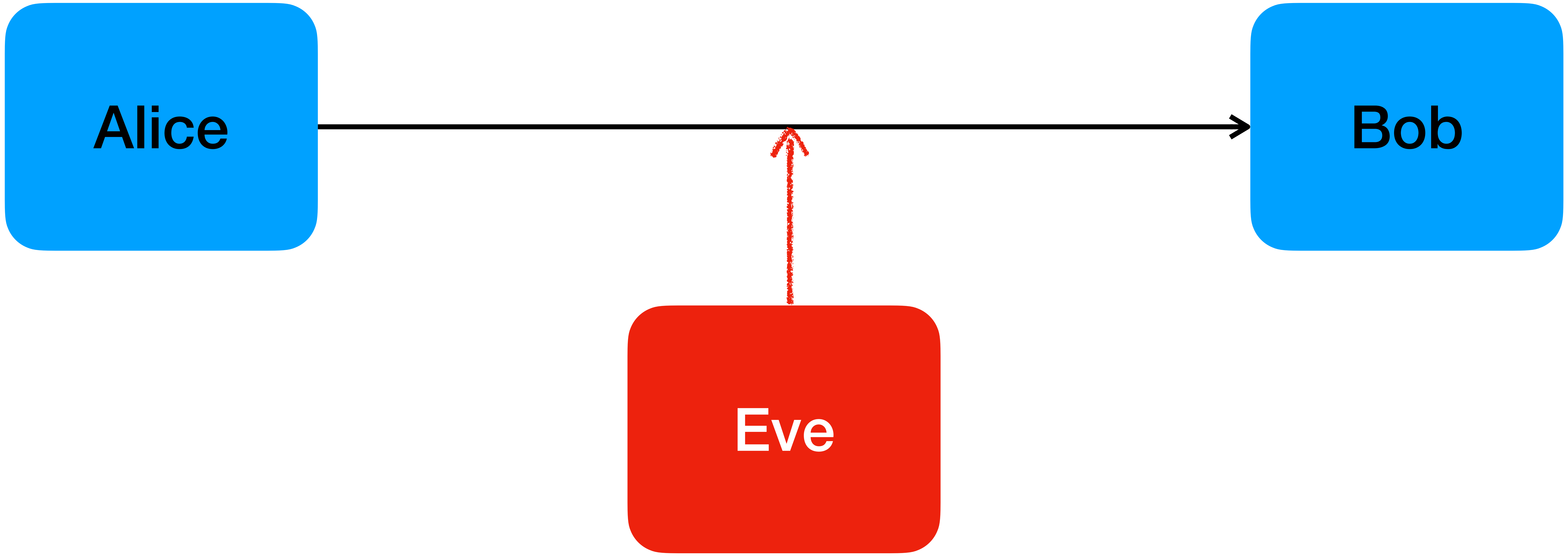
Alice, Bob, etc.



Symmetric encryption

Alice, Bob, etc.

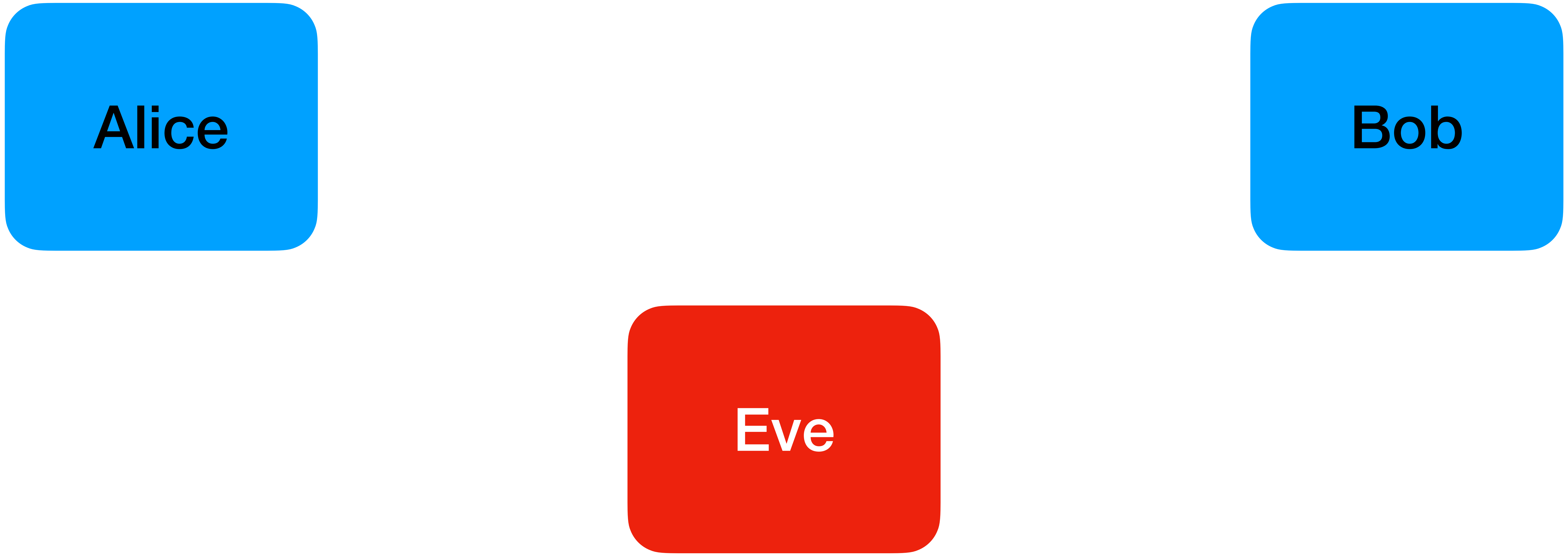
$m \leftarrow Dec_k(c)$



Public-key encryption

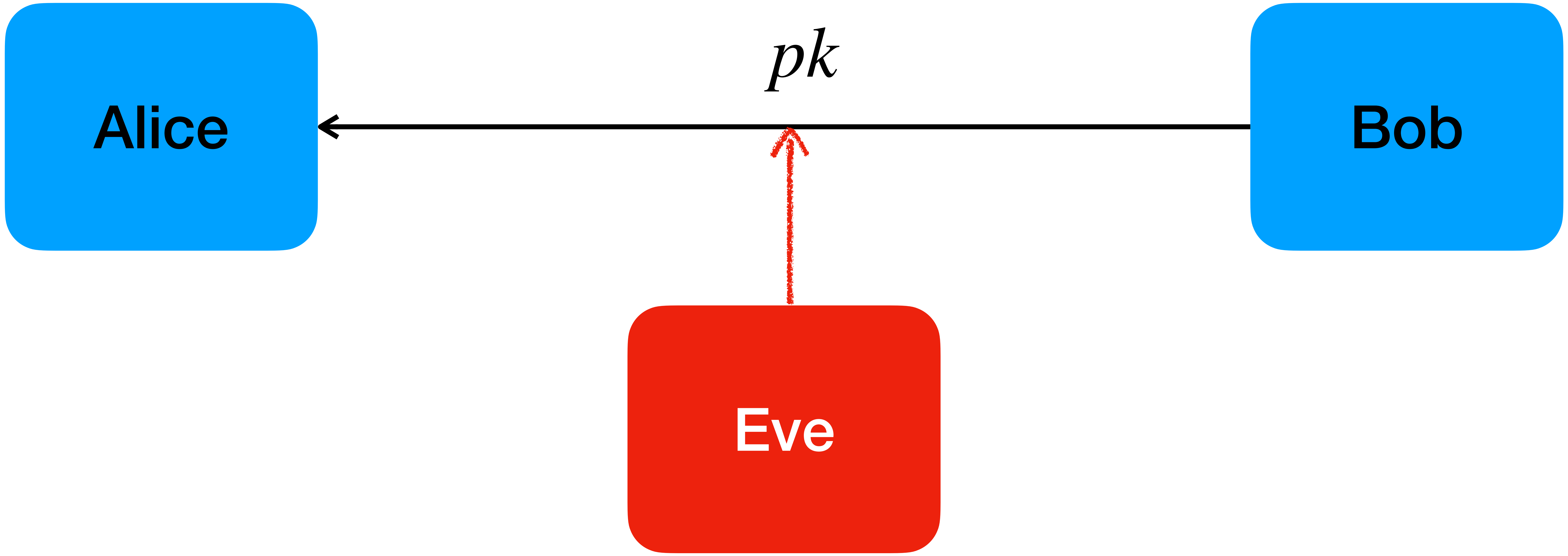
Or - *asymmetric* encryption

$(pk, sk) \leftarrow \mathrm{KeyGen}()$



Public-key encryption

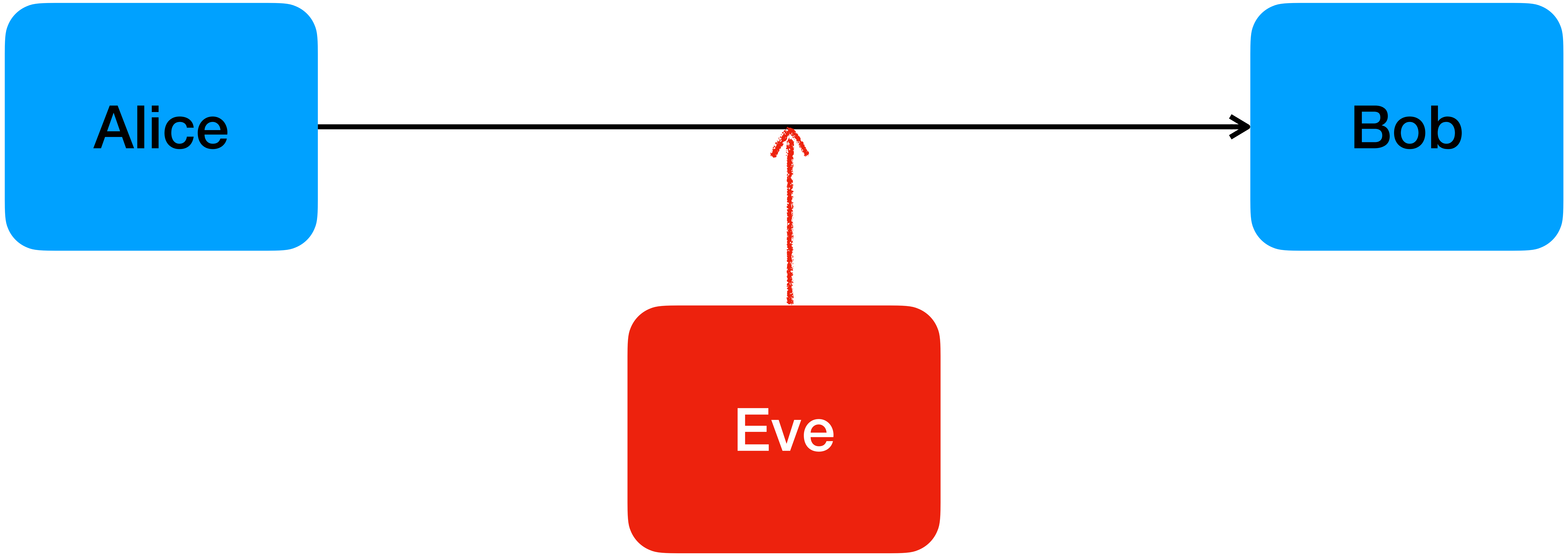
Or - *asymmetric* encryption



Public-key encryption

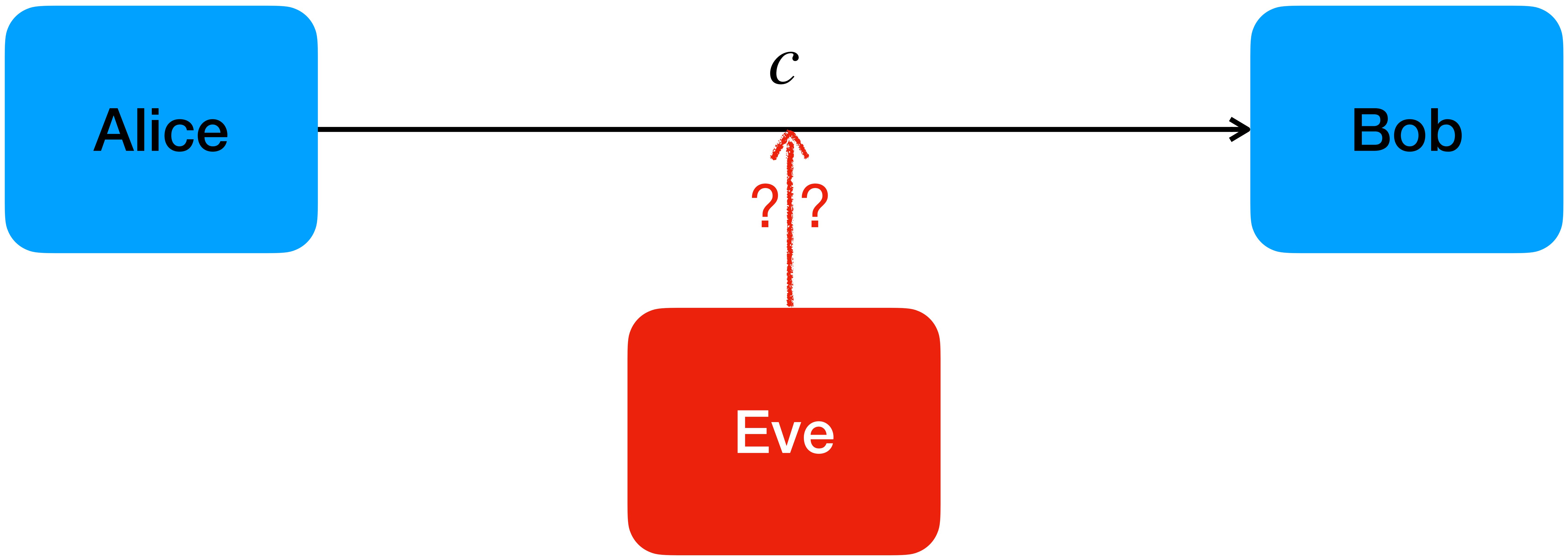
Or - *asymmetric* encryption

$c \leftarrow \mathsf{Enc}_{pk}(m)$



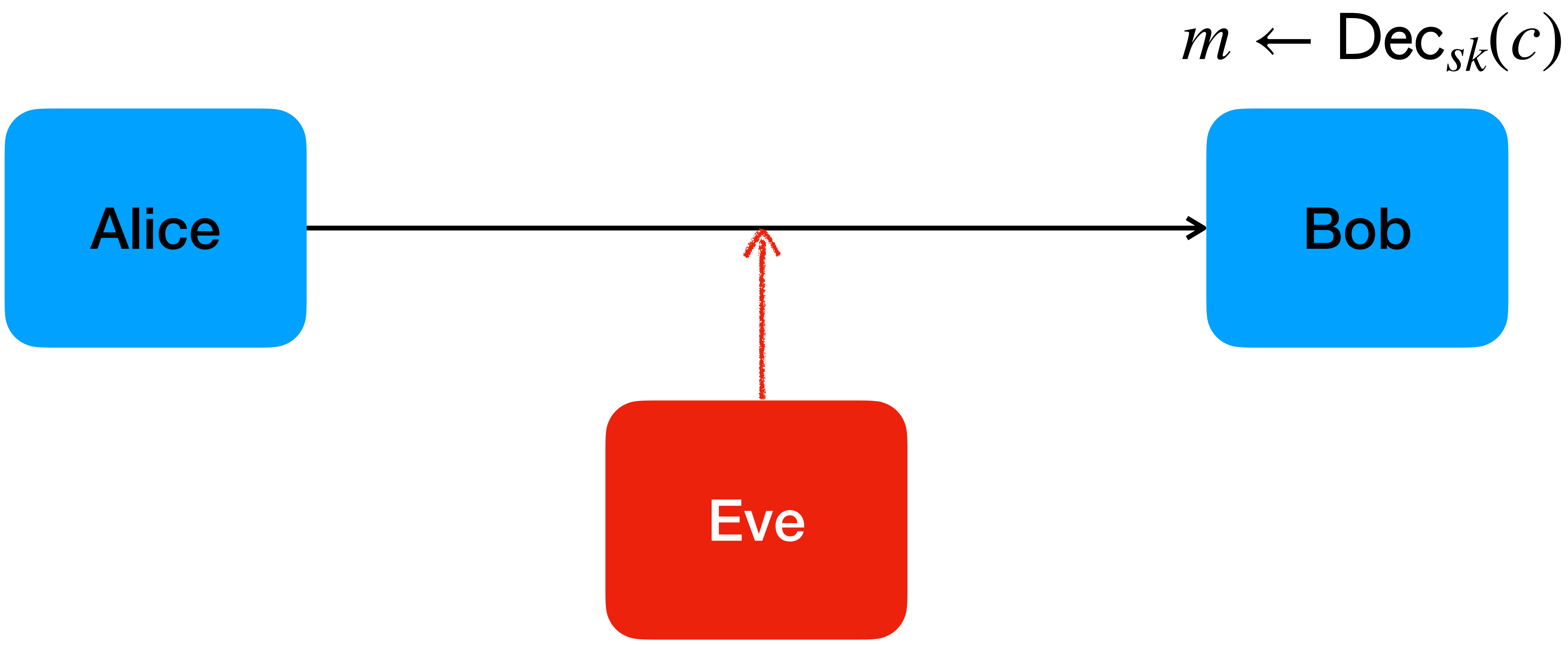
Public-key encryption

Or - *asymmetric* encryption



Public-key encryption

Or - *asymmetric* encryption



Mathematics kicks in

Number Theory, precisely

- Messages are integer numbers
- Fix a natural number n. Consider the integers modulo n

Mathematics kicks in

Number Theory, precisely

- Messages are integer numbers
- Fix a natural number n. Consider the integers modulo n
- Take $n = 12$:
- $0 \mod 12 = 0, 1 \mod 12 = 1,...,11 \mod 12 = 11,12 \mod 12 = 0,...$

Mathematics kicks in

Number Theory, precisely

- Messages are integer numbers
- Fix a natural number n. Consider the integers modulo n
- Take $n = 12$:
- $0 \mod 12 = 0, 1 \mod 12 = 1, ..., 11 \mod 12 = 11, 12 \mod 12 = 0, ...$
- Integers modulo n can be added and subtracted

Mathematics kicks in

Number Theory, precisely

- Messages are integer numbers
- Fix a natural number n. Consider the integers modulo n
- Take $n = 12$:
- $0 \mod 12 = 0, 1 \mod 12 = 1,\ldots,11 \mod 12 = 11,12 \mod 12 = 0,\ldots$
- Integers modulo n can be added and subtracted
- Formally: $\mathbb{Z}_n = \{0,1,\ldots,n-1\}$ is a group under addition

Modular arithmetic

Prime moduli

- Is $\mathbb{Z}_n$ a multiplicative group?

Modular arithmetic

Prime moduli

- Is $\mathbb{Z}_n$ a multiplicative group?
- In general, no.
- Take m in $\mathbb{Z}_n$. It has multiplicative inverse (another $m' \in \mathbb{Z}_n$ such that $m \cdot m' = 1 \mod n$) only if $\gcd(m, n) = 1$

Modular arithmetic

Considering multiplication

- Is $\mathbb{Z}_n$ a multiplicative group?
- In general, no.
- Take m in $\mathbb{Z}_n$. It has multiplicative inverse (another $m' \in \mathbb{Z}_n$ such that $m \cdot m' = 1 \mod n$) only if $\gcd(m, n) = 1$
- How many elements in $\mathbb{Z}_n$ have multiplicative inverse?

Modular arithmetic

Considering multiplication

- Is $\mathbb{Z}_n$ a multiplicative group?
- In general, no.
- Take m in $\mathbb{Z}_n$. It has multiplicative inverse (another $m' \in \mathbb{Z}_n$ such that $m \cdot m' = 1 \mod n$) only if $\gcd(m, n) = 1$
- How many elements in $\mathbb{Z}_n$ have multiplicative inverse?
- $\phi(n)$: counts the integers $\leq n$ coprimes with n
- p prime $\implies \phi(n) = p - 1$

Modular arithmetic

A fundamental theorem

- Theorem (Euler): if a is coprime with n then $a^{\phi(n)} = 1 \mod n$

Modular arithmetic

A fundamental theorem

- Theorem (Euler): if a is coprime with n then $a^{\phi(n)} = 1 \mod n$
- Example: $n = 100$, choose a coprime with 100

Modular arithmetic

A fundamental theorem

- Theorem (Euler): if a is coprime with n then $a^{\phi(n)} = 1 \mod n$
- Example: $n = 100$, choose a coprime with 100
- $\phi(100) = 40$. Then $a^{40} = 1 \mod 100$

Modular arithmetic

A fundamental theorem

- Theorem (Euler): if a is coprime with n then $a^{\phi(n)} = 1 \mod n$
- Example: $n = 100$, choose a coprime with 100
- $\phi(100) = 40$. Then $a^{40} = 1 \mod 100$
- This implies $a^{80} = 1 \mod 100$, and $a^{81} = a \mod 100$

Modular arithmetic

A fundamental theorem

- Theorem (Euler): if a is coprime with n then $a^{\phi(n)} = 1 \mod n$
- Example: $n = 100$, choose a coprime with 100
- $\phi(100) = 40$. Then $a^{40} = 1 \mod 100$
- This implies $a^{80} = 1 \mod 100$, and $a^{81} = a \mod 100$
- $a^{81} = a \mod 100 \Rightarrow a^{27 \cdot 3} = a \mod 100 \Rightarrow (a^3)^{27} = a \mod 100$

Modular arithmetic

A fundamental theorem

- Theorem (Euler): if a is coprime with n then $a^{\phi(n)} = 1 \mod n$
- Example: $n = 100$, choose a coprime with 100
- $\phi(100) = 40$. Then $a^{40} = 1 \mod 100$
- This implies $a^{80} = 1 \mod 100$, and $a^{81} = a \mod 100$
- $a^{81} = a \mod 100 \Rightarrow a^{27\cdot 3} = a \mod 100 \Rightarrow (a^3)^{27} = a \mod 100$

- Raising to the 27-power is the same as taking the cubic root (modulo 100)

Modular arithmetic

A fundamental theorem

- In general:
- a coprime with n and $e \cdot d = 1 \mod \phi(n) \Rightarrow a^{e \cdot d} = a \mod n$
- That is, taking the d-th power is equal to taking the e-th root modulo n

Modular arithmetic

A fundamental theorem

- In general
- a coprime with n and $e \cdot d = 1 \mod \phi(n) \Rightarrow a^{e \cdot d} = a \mod n$
- That is, taking the d-th power is equal to taking the e-th root modulo n
- We have all the ingredients for the RSA encryption scheme

RSA

An application of Euler theorem

- Keys generation:
  - $p, q \xleftarrow{R}$ large primes
  - compute $N = pq$

RSA

An application of Euler theorem

- Keys generation:
  - $p, q \xleftarrow{R}$ large primes
  - compute $N = pq$ Note that: $\phi(N) = (p-1)(q-1)$

RSA

An application of Euler theorem

- Keys generation:
  - $p, q \xleftarrow{R}$ large primes
  - compute $N = pq$ Note that: $\phi(N) = (p-1)(q-1)$
  - pick a positive integer e such that: $\gcd(e, (p-1)(q-1)) = 1$

RSA

An application of Euler theorem

- Keys generation:
  - $p, q \xleftarrow{R}$ large primes
  - compute $N = pq$ Note that: $\phi(N) = (p-1)(q-1)$
  - pick a positive integer e such that: $\gcd(e, (p-1)(q-1)) = 1$
  - compute integer d such that: $de = 1 \mod (p-1)(q-1)$

RSA

An application of Euler theorem

- Keys generation:
  - $p, q \xleftarrow{R}$ large primes
  - compute $N = pq$ Note that: $\phi(N) = (p-1)(q-1)$
  - pick a positive integer e such that: $\gcd(e, (p-1)(q-1)) = 1$
  - compute integer d such that: $de = 1 \mod (p-1)(q-1)$
  - $pk = (N, e)$ $sk = (d, p, q)$

RSA

Encryption & decryption

- a message is represented as an element m of $\mathbb{Z}_N$

RSA

Encryption & decryption

- a message is represented as an element m of $\mathbb{Z}_N$
- Encryption of m using pk:
  - $c \leftarrow m^e \mod N$

RSA

Encryption & decryption

- a message is represented as an element m of $\mathbb{Z}_N$
- Encryption of m using pk:
  - $c \leftarrow m^e \mod N$
- Decryption of c using sk:
  - $m \leftarrow c^d \mod N$

RSA

Encryption & decryption

- a message is represented as an element m of $\mathbb{Z}_N$
- Encryption of m using pk:
  - $c \leftarrow m^e \mod N$
- Decryption of c using sk:
  - $m \leftarrow c^d \mod N$

Does it works?

That is : $\mathsf{Dec}_{sk}(\mathsf{Enc}_{pk}(m)) \stackrel{?}{=} m$

RSA

Encryption & decryption

- a message is represented as an element m of $\mathbb{Z}_N$
- Encryption of m using pk:
  - $c \leftarrow m^e \mod N$
- Decryption of c using sk:
  - $m \leftarrow c^d \mod N$

Does it works?

That is : $\mathsf{Dec}_{sk}(\mathsf{Enc}_{pk}(m)) \stackrel{?}{=} m$

$$\mathsf{Dec}_{sk}(\mathsf{Enc}_{pk}(m)) = (m^e)^d \mod N$$

RSA

Encryption & decryption

- a message is represented as an element m of $\mathbb{Z}_N$
- Encryption of m using pk:
  - $c \leftarrow m^e \mod N$
- Decryption of c using sk:
  - $m \leftarrow c^d \mod N$

Does it works?

That is : $\mathsf{Dec}_{sk}(\mathsf{Enc}_{pk}(m)) \stackrel{?}{=} m$

$$\mathsf{Dec}_{sk}(\mathsf{Enc}_{pk}(m)) = (m^e)^d \mod N$$
$$= m^{ed} \mod N$$

RSA

Encryption & decryption

- a message is represented as an element m of $\mathbb{Z}_N$
- Encryption of m using pk:
  - $c \leftarrow m^e \mod N$
- Decryption of c using sk:
  - $m \leftarrow c^d \mod N$

Does it works?

That is : $\mathsf{Dec}_{sk}(\mathsf{Enc}_{pk}(m)) \stackrel{?}{=} m$

$$\mathsf{Dec}_{sk}(\mathsf{Enc}_{pk}(m)) = (m^e)^d \mod N$$

$$= m^{ed} \mod N$$

$ed = 1 \mod (p-1)(q-1)$

$$= m^{k(p-1)(q-1)+1} \mod N$$

RSA

Encryption & decryption

- a message is represented as an element m of $\mathbb{Z}_N$
- Encryption of m using pk:
  - $c \leftarrow m^e \mod N$
- Decryption of c using sk:
  - $m \leftarrow c^d \mod N$

Does it works?

That is : $\mathsf{Dec}_{sk}(\mathsf{Enc}_{pk}(m)) \stackrel{?}{=} m$

$$\mathsf{Dec}_{sk}(\mathsf{Enc}_{pk}(m)) = (m^e)^d \mod N$$

$$= m^{ed} \mod N$$

$ed = 1 \mod (p-1)(q-1)$

$$= m^{k(p-1)(q-1)+1} \mod N$$

$$= m \cdot m^{k(p-1)(q-1)} \mod N$$

RSA

Encryption & decryption

- a message is represented as an element m of $\mathbb{Z}_N$
- Encryption of m using pk:
  - $c \leftarrow m^e \mod N$
- Decryption of c using sk:
  - $m \leftarrow c^d \mod N$

Does it works?

That is : $\text{Dec}_{sk}(\text{Enc}_{pk}(m)) \stackrel{?}{=} m$

$$\begin{aligned}
\text{Dec}_{sk}(\text{Enc}_{pk}(m)) &= (m^e)^d \mod N \\
&= m^{ed} \mod N \\
&= m^{k(p-1)(q-1)+1} \mod N \\
&= m \; m^{k(p-1)(q-1)} \mod N \\
&= m \mod N
\end{aligned}$$

$ed = 1 \mod (p-1)(q-1)$

equal to 1 by Euler Thm

Is it secure?

It depends...

- RSA problem: given $c = m^e \mod N$ find m (without knowing d)

Is it secure?

It depends...

- RSA problem: given $c = m^e \mod N$ find m (without knowing d)
- That is: compute the e-th root of c modulo N

Is it secure?

It depends...

- RSA problem: given $c = m^e \mod N$ find m (without knowing d)
- That is: compute the e-th root of c modulo N
- It is widely believed to be a hard problem

Is it secure?

It depends...

- RSA problem: given $c = m^e \mod N$ find m (without knowing d)
- That is: compute the e-th root of c modulo N
- It is widely believed to be a hard problem
- Note: if one knows the factorization of N, then solving the RSA problem is easy

Is it secure?

It depends...

- RSA problem: given $c = m^e \mod N$ find m (without knowing d)
- That is: compute the e-th root of c modulo N
- It is widely believed to be a hard problem
- Note: if one knows the factorization of N, then solving the RSA problem is easy
- p, q are known $\Rightarrow$ $(p-1)(q-1)$ is known $\Rightarrow$ easy to compute d

Is it secure?

It depends...

- RSA problem: given $c = m^e \mod N$ find m (without knowing d)
- That is: compute the e-th root of c modulo N
- It is widely believed to be a hard problem
- Note: if one knows the factorization of N, then solving the RSA problem is easy
- p, q are known $\Rightarrow$ $(p-1)(q-1)$ is known $\Rightarrow$ easy to compute d
- Is it necessary to factor N to easily solve the RSA problem?

Is it secure?

It depends...

- RSA problem: given $c = m^e \mod N$ find m (without knowing d)
- That is: compute the e-th root of c modulo N
- It is widely believed to be a hard problem
- Note: if one knows the factorization of N, then solving the RSA problem is easy
- p, q are known $\Rightarrow$ $(p-1)(q-1)$ is known $\Rightarrow$ easy to compute d
- Is it necessary to factor N to easily solve the RSA problem? Nobody knows

Real-world RSA parameters

- $p =$
2852548559318792131996856928776907825119281774314763570567815857161180832220837519246298437849803897808133857804293068360720289689219293552787014914773341602205226485142445035331780826110954892820763749663847341009298597810475043222248992578332801443153983881114506335785938838763490070529273256179374411715693740352023288821141872517444832843142745124248167962975328995374423885701884560435346893852806427942025281363464456832501641858113503599539750773425462372913221908859606938360093958840350104476559208036630009503322492880333306863971638512561292424076321787354414847849812047565954709580803419355650482786511 7

- $q =$
2509785003812493366094206331512468181316605756771693905249883385880642709508878477068939652775190207339573525959319662037263317416923209740213635937018922141058811459917391399502361197229181706040899508893306387999326350207796710736265991703103379230835418488059153440114942935256332194856506022411021260589273249908630686846952779779617454980260997007245457420331241921253950673039150817147531381464668508844159047271084111532669050666200100584416669411441413444742531230931714539004213387169133262328065383797323397971665046699690044164577147301910153724302621814797515015487013570026365937192268613525458212916793

- $e = 65537$

Real-world use of RSA

A.k.a. certificates

- Alice and Bob may not know each other
- Bob puts its public key into a message that says:"hi, I'm Bob and this is my pk"
- Then this message is signed by an authority that is known and trusted by both Alice and Bob $\Rightarrow$ certificate of Bob

Real-world use of RSA

A.k.a. certificates

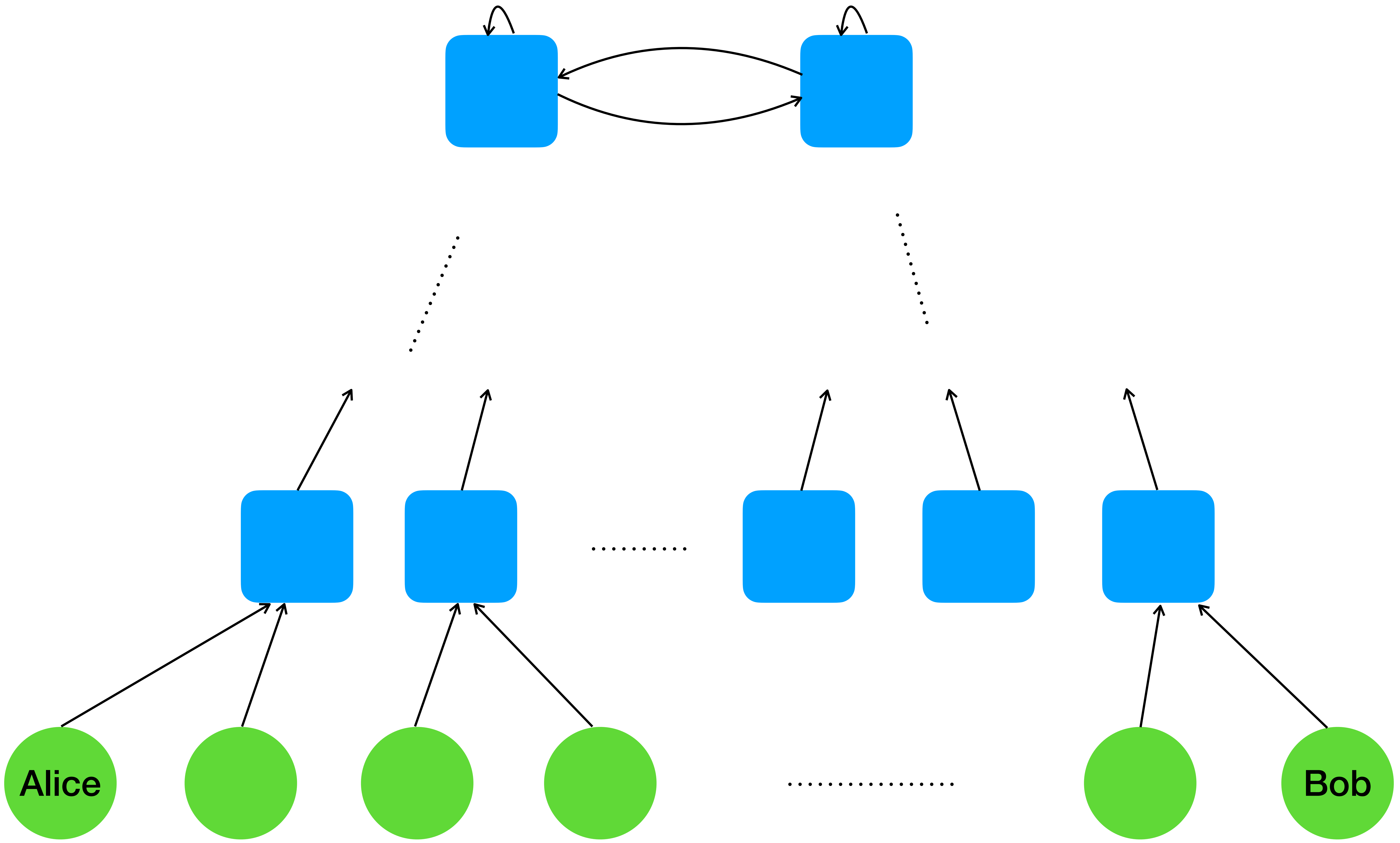
- Alice and Bob may not know each other
- Bob puts its public key into a message that says:"hi, I'm Bob and this is my pk"
- Then this message is signed by an authority that is known and trusted by both Alice and Bob $\Rightarrow$ certificate of Bob
- What if Alice and bob don't know a common authority?

Real-world use of RSA

A.k.a. certificates

- Alice and Bob may not know each other
- Bob puts its public key into a message that says:”hi, I’m Bob and this is my pk”
- Then this message is signed by an authority that is known and trusted by both Alice and Bob $\Rightarrow$ certificate of Bob
- What if Alice and bob don’t know a common authority?
- Idea: local authorities have certificates signed by upper level certificates

Public Key Infrastructure



Authentication

How to sign a certificate?

Authentication

How to sign a certificate?

- Use RSA
- Bob wants to sign a public message m
- “encrypt” with sk (that is, sign m), “decrypt” with pk (that is, verify the signature)

Authentication

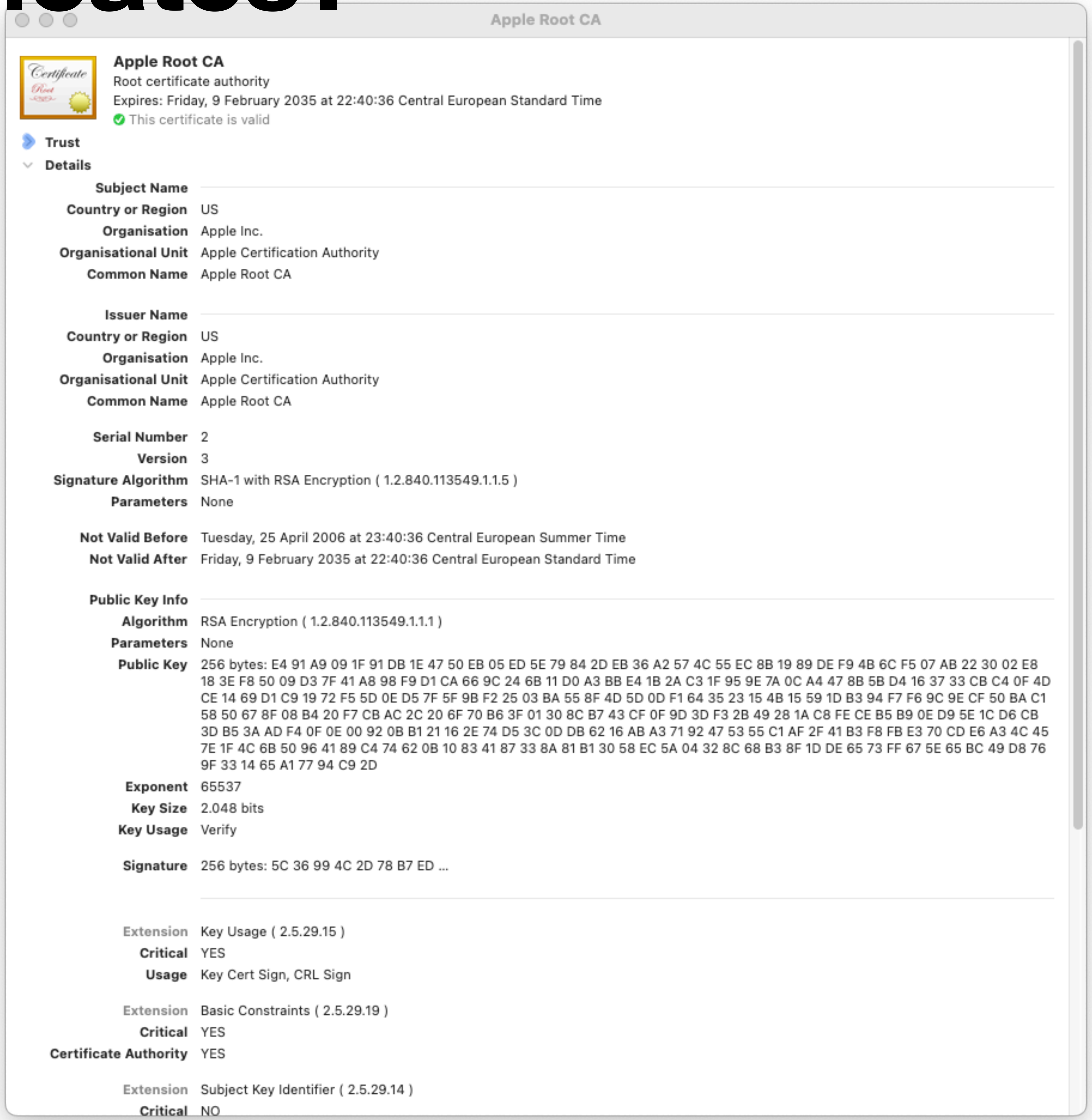
How to sign a certificate?

- Use RSA
- Bob wants to sign a public message m
- "encrypt" with sk (that is, sign m), "decrypt" with pk (that is, verify the signature)

- $\mathsf{Sign}(sk, m) : \sigma \leftarrow m^{sk} \mod N$
- $\mathsf{Verify}(pk, \sigma) : m \leftarrow \sigma^{pk} \mod N$

Where are root CA certificates?

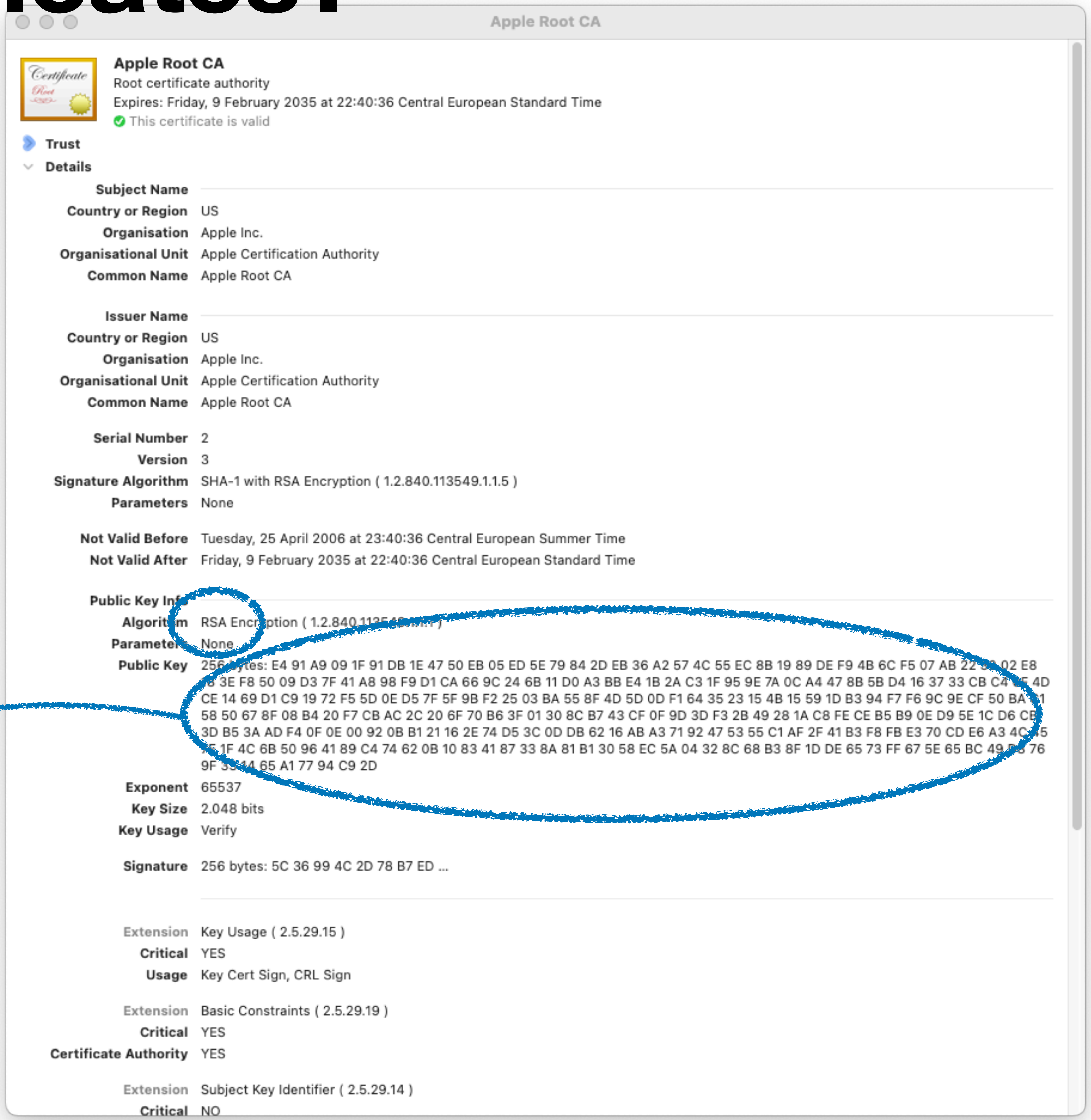
In your browser



Where are root CA certificates?

In your browser

RSA pk



Messages are numbers

Use Unicode, for example

- $'a' \to 97, \ldots, 'z' \to 122$
- ‘we meet at 11’ → 9459448527424017981640577921329

- In practice, RSA is used for encrypt short messages
- For example, 256 bits keys of symmetric encryption schemes

In the Lab

- Python basics
- Implementation of a simple symmetric cipher
- Brute force primality testing
- How to compute the greatest common divisor, efficiently
- Modular arithmetics
- Textbook RSA
- Diffie-Hellman (maybe)