

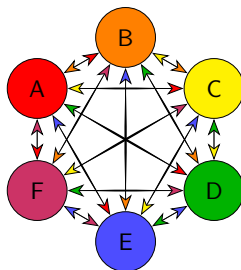
# Cryptography: asymmetric encryption

**Myrto Arapinis** and Markulf Kohlweiss  
School of Informatics  
University of Edinburgh

# Introduction

So far: how two users can protect data using a shared secret key

- ▶ One shared secret key per pair of users that want to communicate

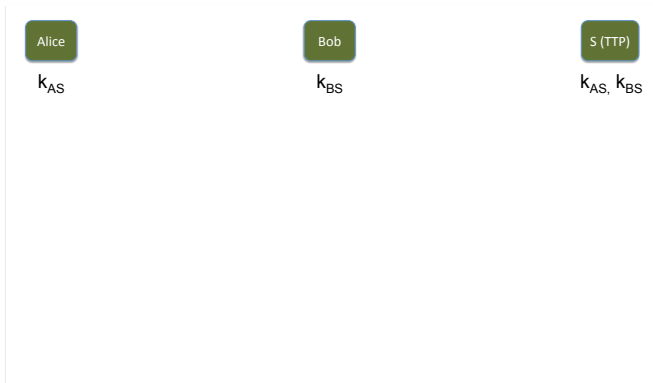


Our goal now: how to establish a shared secret key to begin with?

- ▶ Trusted Third Party (TTP)
- ▶ Diffie-Hellman (DH) protocol
- ▶ *RSA*
- ▶ ElGamal (EG)

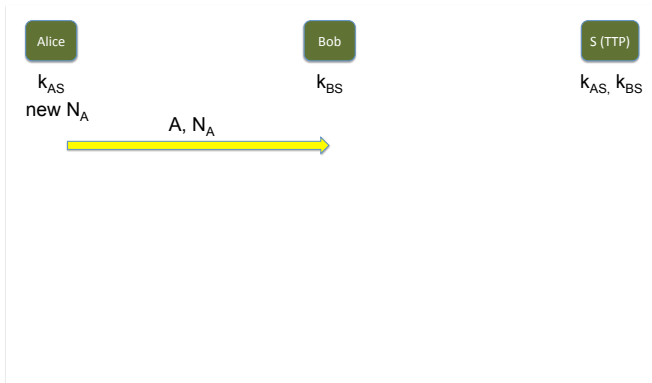
## Online Trusted Third Party (TTP)

- ▶ Users  $U_1, U_2, U_3, \dots, U_n, \dots$
- ▶ Each user  $U_i$  has a shared secret key  $K_i$  with the TTP
- ▶  $U_i$  and  $U_j$  can establish a key  $K_{i,j}$  with the help of the TTP
- ▶  $\{m\}_k$  denotes the symmetric encryption of  $m$  under the key  $k$   
ex: using Paulson's variant of the Yahalom protocol



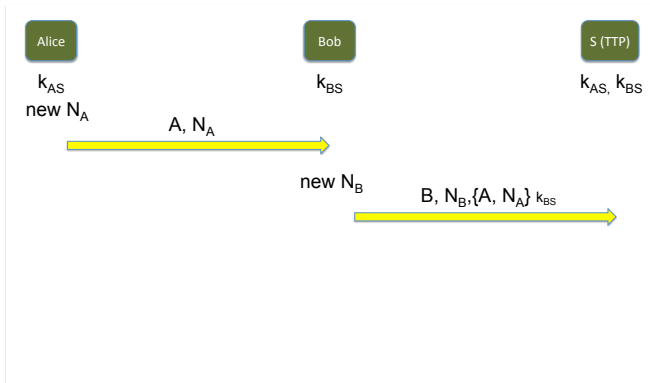
# Online Trusted Third Party (TTP)

- ▶ Users  $U_1, U_2, U_3, \dots, U_n, \dots$
- ▶ Each user  $U_i$  has a shared secret key  $K_i$  with the TTP
- ▶  $U_i$  and  $U_j$  can establish a key  $K_{i,j}$  with the help of the TTP
- ▶  $\{m\}_k$  denotes the symmetric encryption of  $m$  under the key  $k$   
ex: using Paulson's variant of the Yahalom protocol



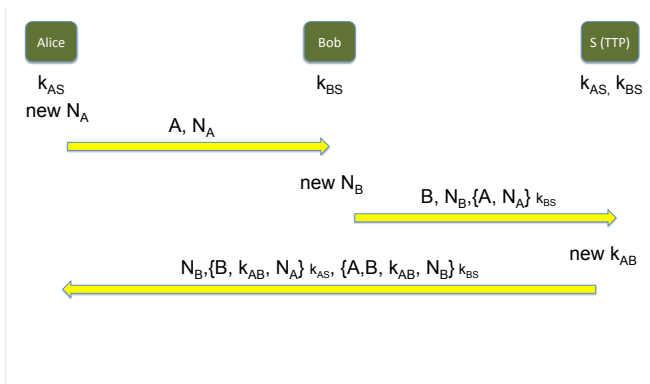
# Online Trusted Third Party (TTP)

- ▶ Users  $U_1, U_2, U_3, \dots, U_n, \dots$
- ▶ Each user  $U_i$  has a shared secret key  $K_i$  with the TTP
- ▶  $U_i$  and  $U_j$  can establish a key  $K_{i,j}$  with the help of the TTP
- ▶  $\{m\}_k$  denotes the symmetric encryption of  $m$  under the key  $k$   
ex: using Paulson's variant of the Yahalom protocol



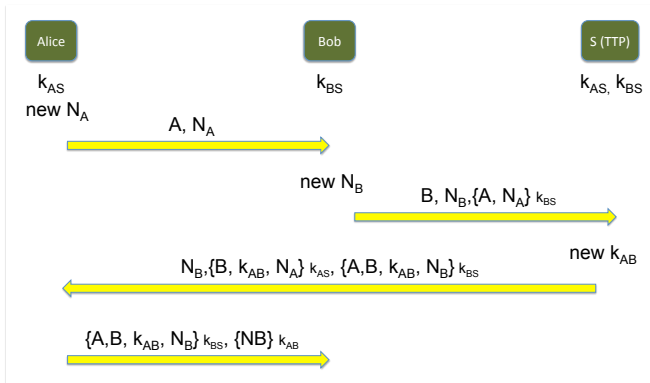
# Online Trusted Third Party (TTP)

- ▶ Users  $U_1, U_2, U_3, \dots, U_n, \dots$
- ▶ Each user  $U_i$  has a shared secret key  $K_i$  with the TTP
- ▶  $U_i$  and  $U_j$  can establish a key  $K_{i,j}$  with the help of the TTP
- ▶  $\{m\}_k$  denotes the symmetric encryption of  $m$  under the key  $k$   
ex: using Paulson's variant of the Yahalom protocol



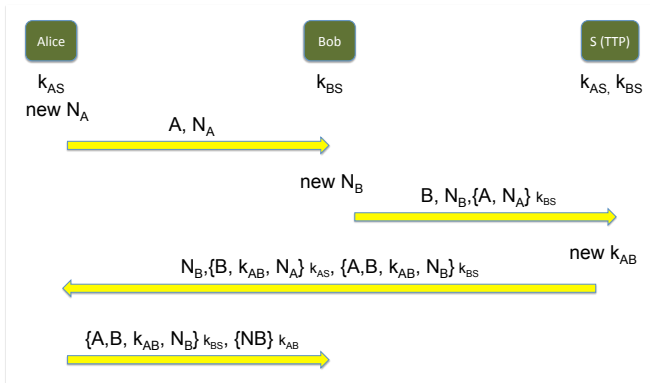
# Online Trusted Third Party (TTP)

- ▶ Users  $U_1, U_2, U_3, \dots, U_n, \dots$
- ▶ Each user  $U_i$  has a shared secret key  $K_i$  with the TTP
- ▶  $U_i$  and  $U_j$  can establish a key  $K_{i,j}$  with the help of the TTP
- ▶  $\{m\}_k$  denotes the symmetric encryption of  $m$  under the key  $k$   
ex: using Paulson's variant of the Yahalom protocol



# Online Trusted Third Party (TTP)

- ▶ Users  $U_1, U_2, U_3, \dots, U_n, \dots$
- ▶ Each user  $U_i$  has a shared secret key  $K_i$  with the TTP
- ▶  $U_i$  and  $U_j$  can establish a key  $K_{i,j}$  with the help of the TTP
- ▶  $\{m\}_k$  denotes the symmetric encryption of  $m$  under the key  $k$   
ex: using Paulson's variant of the Yahalom protocol



Question: can we establish a shared secret key without a TTP?

Answer: Yes! Using public key cryptography.



# Goal of public-key encryption



Alice



# Goal of public-key encryption



Alice

Bob



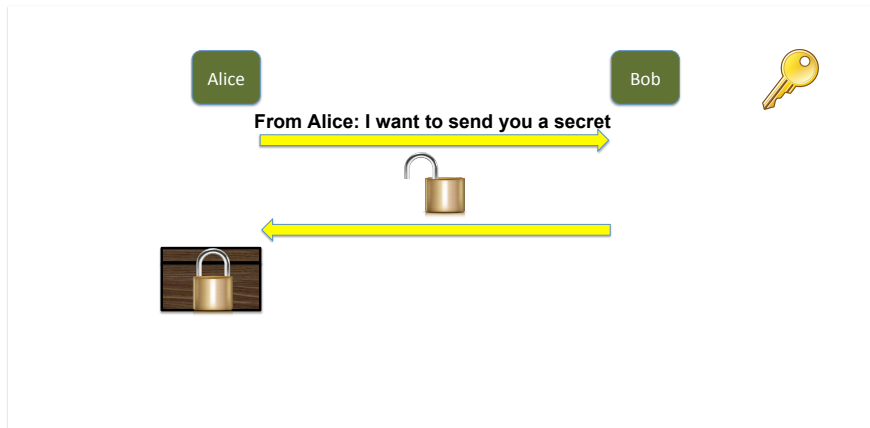
**From Alice: I want to send you a secret**



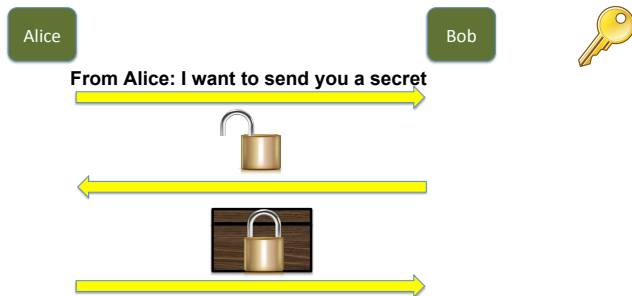
# Goal of public-key encryption



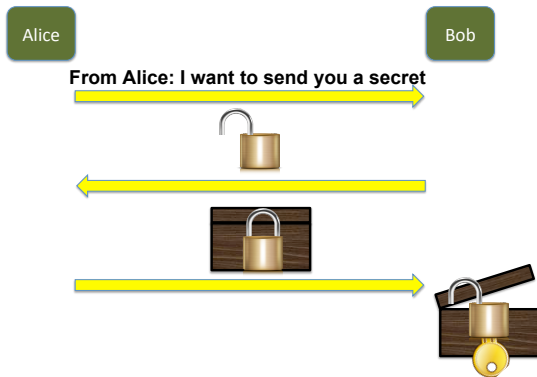
# Goal of public-key encryption



# Goal of public-key encryption

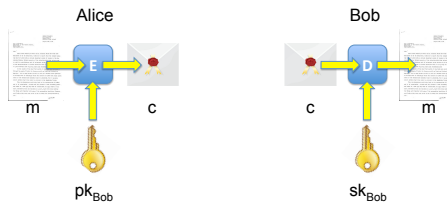


# Goal of public-key encryption



# Public-key encryption - Definition

- key generation algorithm:  $G : \rightarrow \mathcal{K} \times \mathcal{K}$   
encryption algorithm  $E : \mathcal{K} \times \mathcal{M} \rightarrow \mathcal{C}$   
decryption algorithm  $D : \mathcal{K} \times \mathcal{C} \rightarrow \mathcal{M}$   
st.  $\forall (sk, pk) \in G$ , and  $\forall m \in \mathcal{M}$ ,  $D(sk, E(pk, m)) = m$



- the decryption key  $sk_{Bob}$  is secret (only known to Bob). The encryption key  $pk_{Bob}$  is known to everyone. And  $sk_{Bob} \neq pk_{Bob}$

We need a bit of number theory now



# Primes

## Definition

$p \in \mathbb{N}$  is a **prime** if its only divisors are 1 and  $p$

Ex: 2, 3, 5, 7, 11, 13, 17, 19, 23, 29

## Theorem

Every  $n \in \mathbb{N}$  has a **unique factorization** as a product of prime numbers (which are called its factors)

Ex:  $23244 = 2 \times 2 \times 3 \times 13 \times 149$

# Relative primes

## Definition

$a$  and  $b$  in  $\mathbb{Z}$  are **relative primes** if they have no common factors

## Definition

The Euler function  $\phi(n)$  is the number of elements that are relative primes with  $n$ :

$$\phi(n) = |\{m \mid 0 < m < n \text{ and } \gcd(m, n) = 1\}|$$

# Relative primes

## Definition

$a$  and  $b$  in  $\mathbb{Z}$  are **relative primes** if they have no common factors

## Definition

The Euler function  $\phi(n)$  is the number of elements that are relative primes with  $n$ :

$$\phi(n) = |\{m \mid 0 < m < n \text{ and } \gcd(m, n) = 1\}|$$

► For  $p$  prime:  $\phi(p) =$

# Relative primes

## Definition

$a$  and  $b$  in  $\mathbb{Z}$  are **relative primes** if they have no common factors

## Definition

The Euler function  $\phi(n)$  is the number of elements that are relative primes with  $n$ :

$$\phi(n) = |\{m \mid 0 < m < n \text{ and } \gcd(m, n) = 1\}|$$

- For  $p$  prime:  $\phi(p) = p-1$

# Relative primes

## Definition

$a$  and  $b$  in  $\mathbb{Z}$  are **relative primes** if they have no common factors

## Definition

The Euler function  $\phi(n)$  is the number of elements that are relative primes with  $n$ :

$$\phi(n) = |\{m \mid 0 < m < n \text{ and } \gcd(m, n) = 1\}|$$

- ▶ For  $p$  prime:  $\phi(p) = p-1$
- ▶ For  $p$  and  $q$  primes:  $\phi(p \cdot q) =$

# Relative primes

## Definition

$a$  and  $b$  in  $\mathbb{Z}$  are **relative primes** if they have no common factors

## Definition

The Euler function  $\phi(n)$  is the number of elements that are relative primes with  $n$ :

$$\phi(n) = |\{m \mid 0 < m < n \text{ and } \gcd(m, n) = 1\}|$$

- ▶ For  $p$  prime:  $\phi(p) = p-1$
- ▶ For  $p$  and  $q$  primes:  $\phi(p \cdot q) = (p-1)(q-1)$

$\mathbb{Z}_n$ 

► Let  $n \in \mathbb{N}$ . We define  $\mathbb{Z}_n = \{0, \dots, n-1\}$

$$\forall a \in \mathbb{Z}, \forall b \in \mathbb{Z}_n, a \equiv b \pmod{n} \Leftrightarrow \exists k \in \mathbb{N}. a = b + k \cdot n$$

- ▶ Let  $n \in \mathbb{N}$ . We define  $\mathbb{Z}_n = \{0, \dots, n-1\}$

$$\forall a \in \mathbb{Z}, \forall b \in \mathbb{Z}_n, a \equiv b \pmod{n} \Leftrightarrow \exists k \in \mathbb{N}. a = b + k \cdot n$$

- ▶ Modular inversion: the inverse of  $x \in \mathbb{Z}_n$  is  $y \in \mathbb{Z}_n$  s.t.  
 $x \cdot y \equiv 1 \pmod{n}$ . We denote  $x^{-1}$  the inverse of  $x$  mod  $n$



- ▶ Let  $n \in \mathbb{N}$ . We define  $\mathbb{Z}_n = \{0, \dots, n-1\}$

$$\forall a \in \mathbb{Z}, \forall b \in \mathbb{Z}_n, a \equiv b \pmod{n} \Leftrightarrow \exists k \in \mathbb{N}. a = b + k \cdot n$$

- ▶ Modular inversion: the inverse of  $x \in \mathbb{Z}_n$  is  $y \in \mathbb{Z}_n$  s.t.  
 $x \cdot y \equiv 1 \pmod{n}$ . We denote  $x^{-1}$  the inverse of  $x$  mod  $n$   
Ex:  $7^{-1}$  in  $\mathbb{Z}_{12}$ :

- ▶ Let  $n \in \mathbb{N}$ . We define  $\mathbb{Z}_n = \{0, \dots, n-1\}$

$$\forall a \in \mathbb{Z}, \forall b \in \mathbb{Z}_n, a \equiv b \pmod{n} \Leftrightarrow \exists k \in \mathbb{N}. a = b + k \cdot n$$

- ▶ Modular inversion: the inverse of  $x \in \mathbb{Z}_n$  is  $y \in \mathbb{Z}_n$  s.t.  
 $x \cdot y \equiv 1 \pmod{n}$ . We denote  $x^{-1}$  the inverse of  $x$  mod  $n$   
Ex:  $7^{-1}$  in  $\mathbb{Z}_{12}$ : 7

- ▶ Let  $n \in \mathbb{N}$ . We define  $\mathbb{Z}_n = \{0, \dots, n-1\}$

$$\forall a \in \mathbb{Z}, \forall b \in \mathbb{Z}_n, a \equiv b \pmod{n} \Leftrightarrow \exists k \in \mathbb{N}. a = b + k \cdot n$$

- ▶ Modular inversion: the inverse of  $x \in \mathbb{Z}_n$  is  $y \in \mathbb{Z}_n$  s.t.  
 $x \cdot y \equiv 1 \pmod{n}$ . We denote  $x^{-1}$  the inverse of  $x$  mod  $n$

Ex:  $7^{-1}$  in  $\mathbb{Z}_{12}$ : 7

$4^{-1}$  in  $\mathbb{Z}_{12}$ :

- ▶ Let  $n \in \mathbb{N}$ . We define  $\mathbb{Z}_n = \{0, \dots, n-1\}$

$$\forall a \in \mathbb{Z}, \forall b \in \mathbb{Z}_n, a \equiv b \pmod{n} \Leftrightarrow \exists k \in \mathbb{N}. a = b + k \cdot n$$

- ▶ Modular inversion: the inverse of  $x \in \mathbb{Z}_n$  is  $y \in \mathbb{Z}_n$  s.t.  
 $x \cdot y \equiv 1 \pmod{n}$ . We denote  $x^{-1}$  the inverse of  $x$  mod  $n$

Ex:  $7^{-1}$  in  $\mathbb{Z}_{12}$ : 7

$4^{-1}$  in  $\mathbb{Z}_{12}$ : 4 has no inverse in  $\mathbb{Z}_{12}$

- ▶ Let  $n \in \mathbb{N}$ . We define  $\mathbb{Z}_n = \{0, \dots, n-1\}$

$$\forall a \in \mathbb{Z}, \forall b \in \mathbb{Z}_n, a \equiv b \pmod{n} \Leftrightarrow \exists k \in \mathbb{N}. a = b + k \cdot n$$

- ▶ Modular inversion: the inverse of  $x \in \mathbb{Z}_n$  is  $y \in \mathbb{Z}_n$  s.t.  
 $x \cdot y \equiv 1 \pmod{n}$ . We denote  $x^{-1}$  the inverse of  $x$  mod  $n$

Ex:  $7^{-1}$  in  $\mathbb{Z}_{12}$ : 7

$4^{-1}$  in  $\mathbb{Z}_{12}$ : 4 has no inverse in  $\mathbb{Z}_{12}$

### Theorem

Let  $n \in \mathbb{N}$ . Let  $x \in \mathbb{Z}_n$ .  $x$  has a inverse in  $\mathbb{Z}_n$  iff  $\gcd(x, n) = 1$

$$\mathbb{Z}_N^*$$

- Let  $n \in \mathbb{N}$ . We define  $\mathbb{Z}_n^* = \{x \in \mathbb{Z}_n \mid \gcd(x, n) = 1\}$   
Ex:  $\mathbb{Z}_{12} = \{1, 5, 7, 11\}$

$$\mathbb{Z}_N^*$$

- ▶ Let  $n \in \mathbb{N}$ . We define  $\mathbb{Z}_n^* = \{x \in \mathbb{Z}_n \mid \gcd(x, n) = 1\}$   
Ex:  $\mathbb{Z}_{12} = \{1, 5, 7, 11\}$
- ▶ Note that  $|\mathbb{Z}_n^*| = \phi(n)$

- ▶ Let  $n \in \mathbb{N}$ . We define  $\mathbb{Z}_n^* = \{x \in \mathbb{Z}_n \mid \gcd(x, n) = 1\}$   
Ex:  $\mathbb{Z}_{12} = \{1, 5, 7, 11\}$
- ▶ Note that  $|\mathbb{Z}_n^*| = \phi(n)$

### Theorem (Euler)

$\forall n \in \mathbb{N}, \forall x \in \mathbb{Z}_n^*, \text{ if } \gcd(x, n) = 1 \text{ then } x^{\phi(n)} \equiv 1 \pmod{n}$



- ▶ Let  $n \in \mathbb{N}$ . We define  $\mathbb{Z}_n^* = \{x \in \mathbb{Z}_n \mid \gcd(x, n) = 1\}$   
Ex:  $\mathbb{Z}_{12}^* = \{1, 5, 7, 11\}$
- ▶ Note that  $|\mathbb{Z}_n^*| = \phi(n)$

### Theorem (Euler)

$\forall n \in \mathbb{N}, \forall x \in \mathbb{Z}_n^*, \text{ if } \gcd(x, n) = 1 \text{ then } x^{\phi(n)} \equiv 1 \pmod{n}$

### Theorem (Euler)

$\forall p \text{ prime}, \mathbb{Z}_p^* \text{ is a cyclic group, i.e.}$

$$\exists g \in \mathbb{Z}_p^*, \{1, g, g^2, g^3, \dots, g^{p-2}\} = \mathbb{Z}_p^*$$

# Intractable problems

# Intractable problems

► FACTORING:

input:  $n \in \mathbb{N}$

output:  $p_1, \dots, p_m$  primes *st.*  $n = p_1 \cdot \dots \cdot p_m$

# Intractable problems

## ► FACTORING:

input:  $n \in \mathbb{N}$

output:  $p_1, \dots, p_m$  primes st.  $n = p_1 \cdot \dots \cdot p_m$

## ► RSAP

input:  $n$  st.  $n = p \cdot q$  with  $2 \leq p, q$  primes

$e$  st.  $\gcd(e, \phi(n)) = 1$

$m^e \bmod n$

output:  $m$

# Intractable problems

- ▶ FACTORING:  
input:  $n \in \mathbb{N}$   
output:  $p_1, \dots, p_m$  primes *st.*  $n = p_1 \cdot \dots \cdot p_m$
- ▶ RSAP  
input:  $n$  *st.*  $n = p \cdot q$  with  $2 \leq p, q$  primes  
 $e$  *st.*  $\gcd(e, \phi(n)) = 1$   
 $m^e \bmod n$   
output:  $m$
- ▶ DISCRETE LOG:  
input: prime  $p$ , generator  $g$  of  $\mathbb{Z}_p^*$ ,  $y \in \mathbb{Z}_p^*$   
output:  $x$  such that  $y = g^x \pmod{p}$

# Intractable problems

► FACTORING:

input:  $n \in \mathbb{N}$

output:  $p_1, \dots, p_m$  primes st.  $n = p_1 \cdot \dots \cdot p_m$

► RSAP

input:  $n$  st.  $n = p \cdot q$  with  $2 \leq p, q$  primes

$e$  st.  $\gcd(e, \phi(n)) = 1$

$m^e \bmod n$

output:  $m$

► DISCRETE LOG:

input: prime  $p$ , generator  $g$  of  $\mathbb{Z}_p^*$ ,  $y \in \mathbb{Z}_p^*$

output:  $x$  such that  $y = g^x \pmod{p}$

► DHP:

input: prime  $p$ , generator  $g$  of  $\mathbb{Z}_p^*$ ,  $g^a \pmod{p}$ ,  $g^b \pmod{p}$

output:  $g^{ab} \pmod{p}$

We can now go back and see how to establish a key without a TTP

# The Diffie-Hellman (DH) protocol

- ▶ Assumption: the DHP is hard in  $\mathbb{Z}_p^*$
- ▶ Fix a very large prime  $p$ , and  $g$  generator of  $\mathbb{Z}_p^*$



# The Diffie-Hellman (DH) protocol

- ▶ Assumption: the DHP is hard in  $\mathbb{Z}_p^*$
- ▶ Fix a very large prime  $p$ , and  $g$  generator of  $\mathbb{Z}_p^*$

Alice

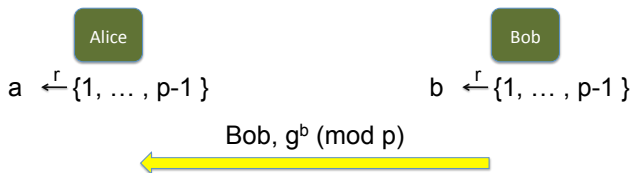
$$a \xleftarrow{r} \{1, \dots, p-1\}$$

Bob

$$b \xleftarrow{r} \{1, \dots, p-1\}$$

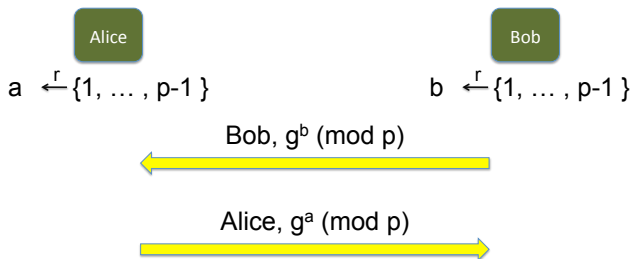
# The Diffie-Hellman (DH) protocol

- ▶ Assumption: the DHP is hard in  $\mathbb{Z}_p^*$
- ▶ Fix a very large prime  $p$ , and  $g \in \{1, \dots, p-1\}$



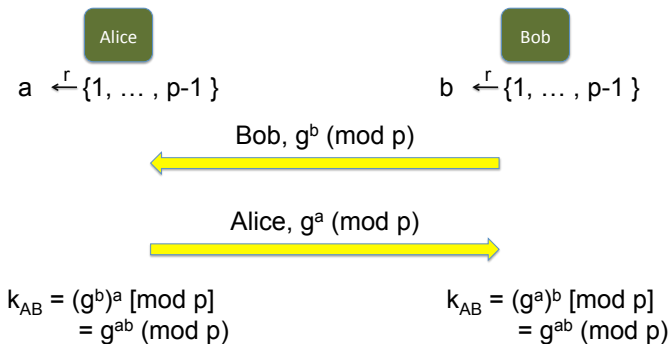
# The Diffie-Hellman (DH) protocol

- ▶ Assumption: the DHP is hard in  $\mathbb{Z}_p^*$
- ▶ Fix a very large prime  $p$ , and  $g \in \{1, \dots, p-1\}$



# The Diffie-Hellman (DH) protocol

- ▶ Assumption: the DHP is hard in  $\mathbb{Z}_p^*$
- ▶ Fix a very large prime  $p$ , and  $g \in \{1, \dots, p-1\}$



# Man in the middle attack on DH

Alice

$$a \xleftarrow{r} \{1, \dots, p-1\}$$

Attacker

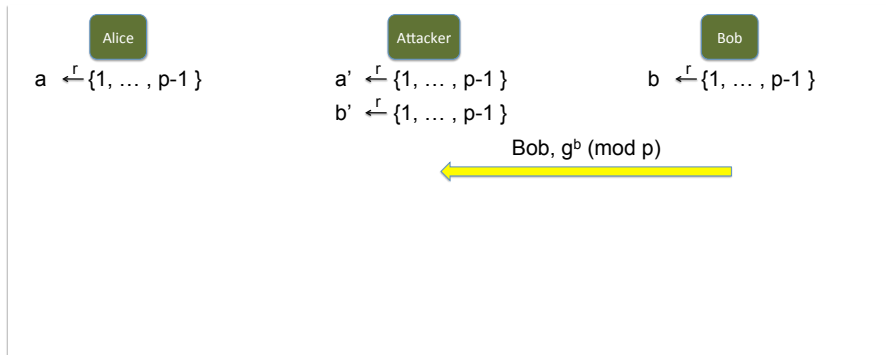
$$a' \xleftarrow{r} \{1, \dots, p-1\}$$

$$b' \xleftarrow{r} \{1, \dots, p-1\}$$

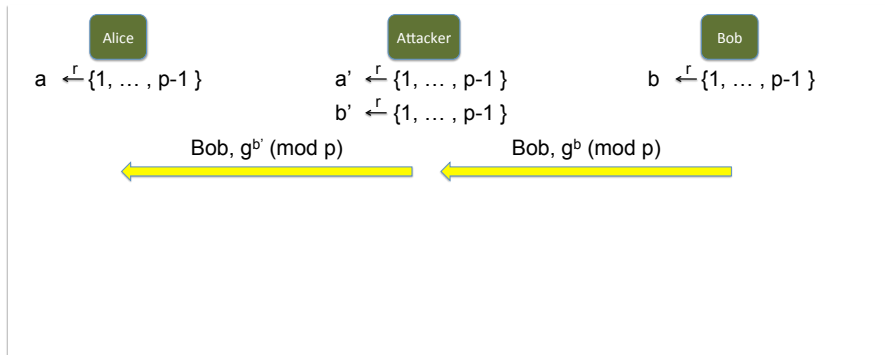
Bob

$$b \xleftarrow{r} \{1, \dots, p-1\}$$

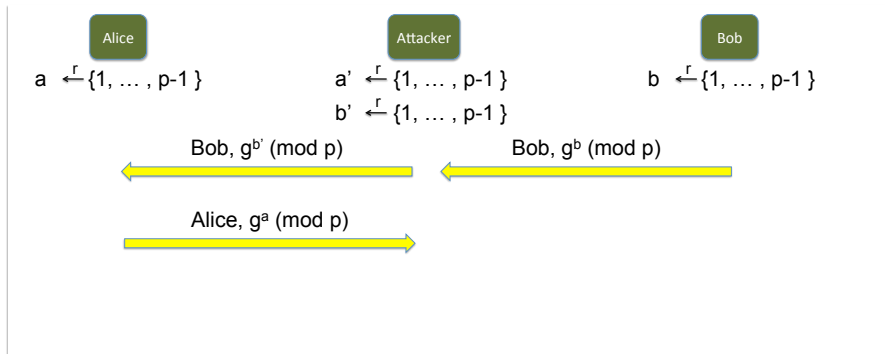
# Man in the middle attack on DH



# Man in the middle attack on DH

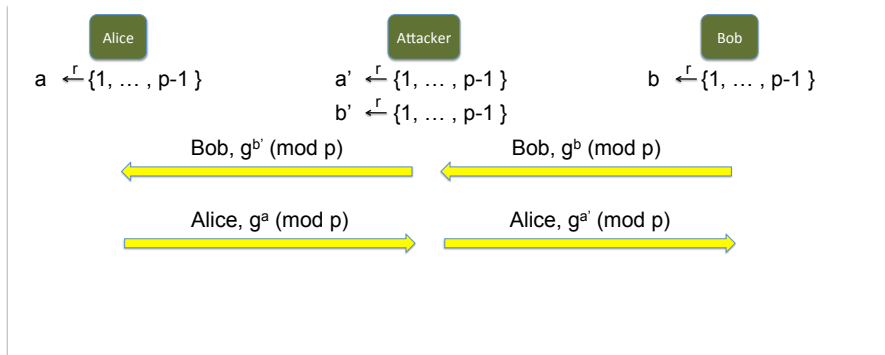


# Man in the middle attack on DH

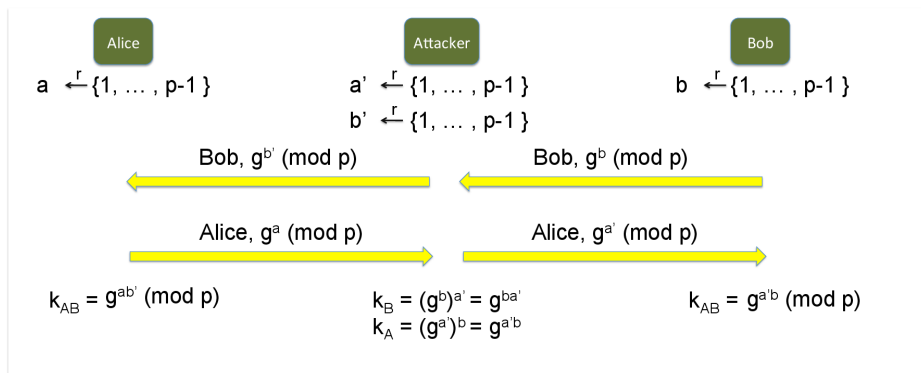




# Man in the middle attack on DH



# Man in the middle attack on DH



## RSA trapdoor permutation

►  $G_{RSA}() = (pk, sk)$

where  $pk = (N, e)$  and  $sk = (N, d)$   
and  $N = p \cdot q$  with  $p, q$  random primes  
and  $e, d \in \mathbb{Z}$  st.  $e \cdot d \equiv 1 \pmod{\phi(N)}$

## RSA trapdoor permutation

►  $G_{RSA}() = (pk, sk)$

where  $pk = (N, e)$  and  $sk = (N, d)$   
and  $N = p \cdot q$  with  $p, q$  random primes  
and  $e, d \in \mathbb{Z}$  st.  $e \cdot d \equiv 1 \pmod{\phi(N)}$

►  $\mathcal{M} = \mathcal{C} = \mathbb{Z}_N$

## RSA trapdoor permutation

- ▶  $G_{RSA}() = (pk, sk)$  where  $pk = (N, e)$  and  $sk = (N, d)$   
and  $N = p \cdot q$  with  $p, q$  random primes  
and  $e, d \in \mathbb{Z}$  st.  $e \cdot d \equiv 1 \pmod{\phi(N)}$
- ▶  $\mathcal{M} = \mathcal{C} = \mathbb{Z}_N$
- ▶  $RSA(pk, x) = x^e \pmod{N}$  where  $pk = (N, e)$

## RSA trapdoor permutation

- ▶  $G_{RSA}() = (pk, sk)$  where  $pk = (N, e)$  and  $sk = (N, d)$   
and  $N = p \cdot q$  with  $p, q$  random primes  
and  $e, d \in \mathbb{Z}$  st.  $e \cdot d \equiv 1 \pmod{\phi(N)}$
- ▶  $\mathcal{M} = \mathcal{C} = \mathbb{Z}_N$
- ▶  $RSA(pk, x) = x^e \pmod{N}$  where  $pk = (N, e)$
- ▶  $RSA^{-1}(sk, x) = x^d \pmod{N}$  where  $sk = (N, d)$

## RSA trapdoor permutation

- ▶  $G_{RSA}() = (pk, sk)$  where  $pk = (N, e)$  and  $sk = (N, d)$   
and  $N = p \cdot q$  with  $p, q$  random primes  
and  $e, d \in \mathbb{Z}$  st.  $e \cdot d \equiv 1 \pmod{\phi(N)}$
- ▶  $\mathcal{M} = \mathcal{C} = \mathbb{Z}_N$
- ▶  $RSA(pk, x) = x^e \pmod{N}$  where  $pk = (N, e)$
- ▶  $RSA^{-1}(sk, x) = x^d \pmod{N}$  where  $sk = (N, d)$
- ▶ Consistency:  $\forall(pk, sk) = G_{RSA}(), \forall x, RSA^{-1}(sk, RSA(pk, x)) = x$

## RSA trapdoor permutation

- ▶  $G_{RSA}() = (pk, sk)$  where  $pk = (N, e)$  and  $sk = (N, d)$   
and  $N = p \cdot q$  with  $p, q$  random primes  
and  $e, d \in \mathbb{Z}$  st.  $e \cdot d \equiv 1 \pmod{\phi(N)}$
- ▶  $\mathcal{M} = \mathcal{C} = \mathbb{Z}_N$
- ▶  $RSA(pk, x) = x^e \pmod{N}$  where  $pk = (N, e)$
- ▶  $RSA^{-1}(sk, x) = x^d \pmod{N}$  where  $sk = (N, d)$
- ▶ Consistency:  $\forall(pk, sk) = G_{RSA}(), \forall x, RSA^{-1}(sk, RSA(pk, x)) = x$   
Proof: Let  $pk = (N, e)$ ,  $sk = (N, d)$ . and  $x \in \mathbb{Z}_N$ . Easy case  
where  $x$  and  $N$  are relatively prime



# RSA trapdoor permutation

- ▶  $G_{RSA}() = (pk, sk)$  where  $pk = (N, e)$  and  $sk = (N, d)$   
and  $N = p \cdot q$  with  $p, q$  random primes  
and  $e, d \in \mathbb{Z}$  st.  $e \cdot d \equiv 1 \pmod{\phi(N)}$
- ▶  $\mathcal{M} = \mathcal{C} = \mathbb{Z}_N$
- ▶  $RSA(pk, x) = x^e \pmod{N}$  where  $pk = (N, e)$
- ▶  $RSA^{-1}(sk, x) = x^d \pmod{N}$  where  $sk = (N, d)$
- ▶ Consistency:  $\forall(pk, sk) = G_{RSA}(), \forall x, RSA^{-1}(sk, RSA(pk, x)) = x$   
Proof: Let  $pk = (N, e)$ ,  $sk = (N, d)$ . and  $x \in \mathbb{Z}_N$ . Easy case  
where  $x$  and  $N$  are relatively prime

$$\begin{aligned} RSA^{-1}(sk, RSA(pk, x)) &= (x^e)^d \pmod{N} \\ &= x^{e \cdot d} \pmod{N} \\ &= x^{1+k\phi(N)} \pmod{N} \\ &= x \cdot x^{k\phi(N)} \pmod{N} \\ &= x \cdot (x^{\phi(N)})^k \pmod{N} \\ &\stackrel{\text{Euler}}{=} x \pmod{N} \end{aligned}$$

# How **NOT** to use *RSA*

$(G_{RSA}, RSA, RSA^{-1})$  is called raw *RSA*. Do not use raw *RSA* directly as an asymmetric cipher!

*RSA* is deterministic  $\Rightarrow$  not secure against chosen plaintext attacks

# ISO standard

Goal: build a CPA secure asymmetric cipher using  $(G_{RSA}, RSA, RSA^{-1})$

Let  $(E_s, D_s)$  be a symmetric encryption scheme over  $(\mathcal{M}, \mathcal{C}, \mathcal{K})$

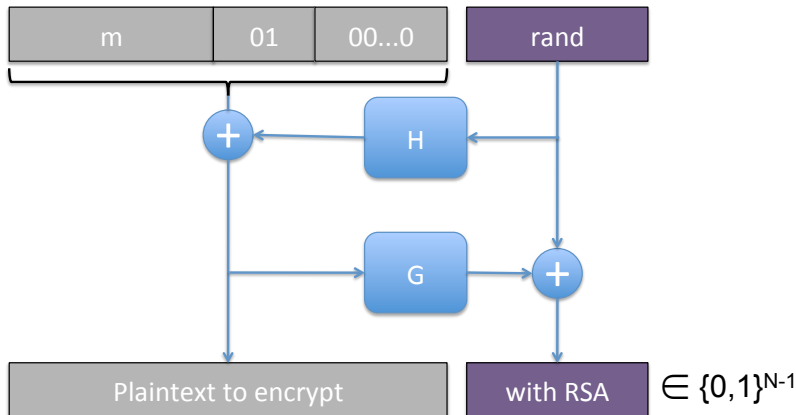
Let  $H : \mathbb{Z}_N^* \rightarrow \mathcal{K}$

Build  $(G_{RSA}, E_{RSA}, D_{RSA})$  as follows

- ▶  $G_{RSA}()$  as described above
- ▶  $E_{RSA}(pk, m)$ :
  - ▶ pick random  $x \in \mathbb{Z}_N^*$
  - ▶  $y \leftarrow RSA(pk, x) (= x^e \bmod N)$
  - ▶  $k \leftarrow H(x)$
  - ▶  $E_{RSA}(pk, m) = y || E_s(k, m)$
- ▶  $D_{RSA}(sk, y || c) = D_s(H(RSA^{-1}(sk, y)), c)$

## PKCS1 v2.0: RSA-OAEP

Goal: build a CPA secure asymmetric cipher using  $(G_{RSA}, RSA, RSA^{-1})$



## ElGamal (EG)

- ▶ Fix prime  $p$ , and generator  $g \in \mathbb{Z}_p^*$

# ElGamal (EG)

- ▶ Fix prime  $p$ , and generator  $g \in \mathbb{Z}_p^*$
- ▶  $\mathcal{M} = \{0, \dots, p-1\}$  and  $\mathcal{C} = \mathcal{M} \times \mathcal{M}$

# ElGamal (EG)

- ▶ Fix prime  $p$ , and generator  $g \in \mathbb{Z}_p^*$
- ▶  $\mathcal{M} = \{0, \dots, p-1\}$  and  $\mathcal{C} = \mathcal{M} \times \mathcal{M}$
- ▶  $G_{EG}() = (pk, sk)$  where  $pk = g^d \pmod{p}$  and  $sk = d$   
and  $d \xleftarrow{r} \{1, \dots, p-2\}$

# ElGamal (EG)

- ▶ Fix prime  $p$ , and generator  $g \in \mathbb{Z}_p^*$
- ▶  $\mathcal{M} = \{0, \dots, p-1\}$  and  $\mathcal{C} = \mathcal{M} \times \mathcal{M}$
- ▶  $G_{EG}() = (pk, sk)$  where  $pk = g^d \pmod{p}$  and  $sk = d$   
and  $d \xleftarrow{r} \{1, \dots, p-2\}$
- ▶  $E_{EG}(pk, x) = (g^r \pmod{p}, m \cdot (g^d)^r \pmod{p})$   
where  $pk = g^d \pmod{p}$   
and  $r \xleftarrow{r} \mathbb{Z}$



# ElGamal (EG)

- ▶ Fix prime  $p$ , and generator  $g \in \mathbb{Z}_p^*$
- ▶  $\mathcal{M} = \{0, \dots, p-1\}$  and  $\mathcal{C} = \mathcal{M} \times \mathcal{M}$
- ▶  $G_{EG}() = (pk, sk)$  where  $pk = g^d \pmod{p}$  and  $sk = d$   
and  $d \xleftarrow{r} \{1, \dots, p-2\}$
- ▶  $E_{EG}(pk, x) = (g^r \pmod{p}, m \cdot (g^d)^r \pmod{p})$   
where  $pk = g^d \pmod{p}$   
and  $r \xleftarrow{r} \mathbb{Z}$
- ▶  $D_{EG}(sk, x) = e^{-d} \cdot c \pmod{p}$  where  $x = (e, c)$

# ElGamal (EG)

- ▶ Fix prime  $p$ , and generator  $g \in \mathbb{Z}_p^*$
- ▶  $\mathcal{M} = \{0, \dots, p-1\}$  and  $\mathcal{C} = \mathcal{M} \times \mathcal{M}$
- ▶  $G_{EG}() = (pk, sk)$  where  $pk = g^d \pmod{p}$  and  $sk = d$   
and  $d \xleftarrow{r} \{1, \dots, p-2\}$
- ▶  $E_{EG}(pk, x) = (g^r \pmod{p}, m \cdot (g^d)^r \pmod{p})$   
where  $pk = g^d \pmod{p}$   
and  $r \xleftarrow{r} \mathbb{Z}$
- ▶  $D_{EG}(sk, x) = e^{-d} \cdot c \pmod{p}$  where  $x = (e, c)$
- ▶ Consistency:  $\forall(pk, sk) = G_{EG}(), \forall x, D_{EG}(sk, E_{EG}(pk, x)) = x$

# ElGamal (EG)

- ▶ Fix prime  $p$ , and generator  $g \in \mathbb{Z}_p^*$
- ▶  $\mathcal{M} = \{0, \dots, p-1\}$  and  $\mathcal{C} = \mathcal{M} \times \mathcal{M}$
- ▶  $G_{EG}() = (pk, sk)$  where  $pk = g^d \pmod{p}$  and  $sk = d$   
and  $d \xleftarrow{r} \{1, \dots, p-2\}$
- ▶  $E_{EG}(pk, x) = (g^r \pmod{p}, m \cdot (g^d)^r \pmod{p})$   
where  $pk = g^d \pmod{p}$   
and  $r \xleftarrow{r} \mathbb{Z}$
- ▶  $D_{EG}(sk, x) = e^{-d} \cdot c \pmod{p}$  where  $x = (e, c)$
- ▶ Consistency:  $\forall(pk, sk) = G_{EG}(), \forall x, D_{EG}(sk, E_{EG}(pk, x)) = x$   
Proof: Let  $pk = g^d \pmod{p}$  and  $sk = d$

# ElGamal (EG)

- ▶ Fix prime  $p$ , and generator  $g \in \mathbb{Z}_p^*$
- ▶  $\mathcal{M} = \{0, \dots, p-1\}$  and  $\mathcal{C} = \mathcal{M} \times \mathcal{M}$
- ▶  $G_{EG}() = (pk, sk)$  where  $pk = g^d \pmod{p}$  and  $sk = d$   
and  $d \xleftarrow{r} \{1, \dots, p-2\}$
- ▶  $E_{EG}(pk, x) = (g^r \pmod{p}, m \cdot (g^d)^r \pmod{p})$   
where  $pk = g^d \pmod{p}$   
and  $r \xleftarrow{r} \mathbb{Z}$
- ▶  $D_{EG}(sk, x) = e^{-d} \cdot c \pmod{p}$  where  $x = (e, c)$
- ▶ Consistency:  $\forall(pk, sk) = G_{EG}(), \forall x, D_{EG}(sk, E_{EG}(pk, x)) = x$   
Proof: Let  $pk = g^d \pmod{p}$  and  $sk = d$

$$D_{EG}(sk, E_{EG}(pk, x)) = (g^r)^{-d} \cdot m \cdot (g^d)^r \pmod{p}$$

# ElGamal (EG)

- ▶ Fix prime  $p$ , and generator  $g \in \mathbb{Z}_p^*$
- ▶  $\mathcal{M} = \{0, \dots, p-1\}$  and  $\mathcal{C} = \mathcal{M} \times \mathcal{M}$
- ▶  $G_{EG}() = (pk, sk)$  where  $pk = g^d \pmod{p}$  and  $sk = d$   
and  $d \xleftarrow{r} \{1, \dots, p-2\}$
- ▶  $E_{EG}(pk, x) = (g^r \pmod{p}, m \cdot (g^d)^r \pmod{p})$   
where  $pk = g^d \pmod{p}$   
and  $r \xleftarrow{r} \mathbb{Z}$
- ▶  $D_{EG}(sk, x) = e^{-d} \cdot c \pmod{p}$  where  $x = (e, c)$
- ▶ Consistency:  $\forall(pk, sk) = G_{EG}(), \forall x, D_{EG}(sk, E_{EG}(pk, x)) = x$   
Proof: Let  $pk = g^d \pmod{p}$  and  $sk = d$

$$\begin{aligned} D_{EG}(sk, E_{EG}(pk, x)) &= (g^r)^{-d} \cdot m \cdot (g^d)^r \pmod{p} \\ &= m \pmod{p} \end{aligned}$$