HA NOI UNIVERSITY OF SCIENCE AND TECHNOLOGY
SCHOOL OF INFORMATION AND COMMUNICATION TECHNOLOGY

Introduction to Cryptography and Security

Public key encryption from Diffie-Hellman

Outline

1. Diffie-Hellman Protocol

2. The ElGamal Public-key System

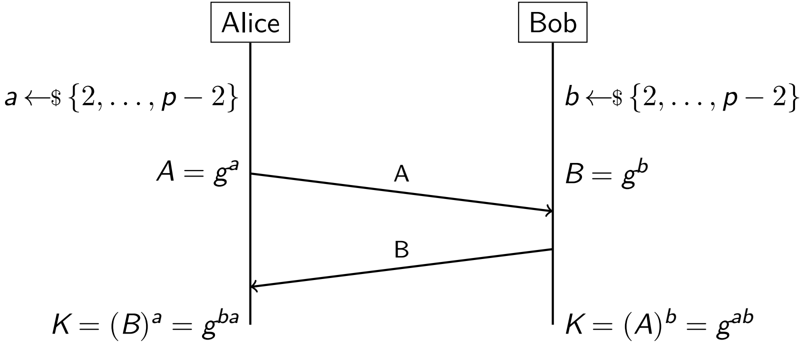
Question

You are given a prime $p = 4969$ and the corresponding multiplicative group $\mathbb{Z}^*_{4969}$.

1. Determinine how many generators exist in $\mathbb{Z}^*_{4969}$.
2. What is the probability of a randomly chosen element $g \in Z^*_{4969}$ being a generator?
3. Determine the smallest generator $g \in Z^*_{4969}$ with $a > 1000$.

The Diffie-Hellman protocol

Public parameter: g, p



Exercise

Compute the two public keys and the shared key K for Diffie-Hellman protocol with the parameters $p = 467$ and $g = 2$, and

1. $a = 3, b = 5$
2. $a = 400, b = 134$
3. $a = 228, b = 57$

Diffie-Hellman protocol over Galois field

- We consider a Diffie-Hellman protocol over Galois field $GF(2^m)$.
- All arithmetic is done in $GF(2^5)$ with $P(x) = x^5 + x^2 + 1$ as an irreducible field polynomial.
- The generator for Diffie-Hellman protocol is $g = x^2$. The private key are $a = 3$ and $b = 12$.
- What is the shared key K?

Security

- Eavesdropper sees:

$$p,\ \ g,\ \ A = g^a \bmod p,\ \text{ and } B = g^b \bmod p$$

- Can she compute $g^{ab} \bmod p$?
- More generally, we define

$$\mathrm{DH}_g(g^a,\ g^b) = g^{ab} \bmod p$$

- How hard is the DH function $\bmod p$?

Exercise

Compute the following values in $\mathbb{Z}_{13}^*$:

- $\mathsf{DH}_7(10,\ 5)$
- $\mathsf{DH}_2(12,\ 9)$

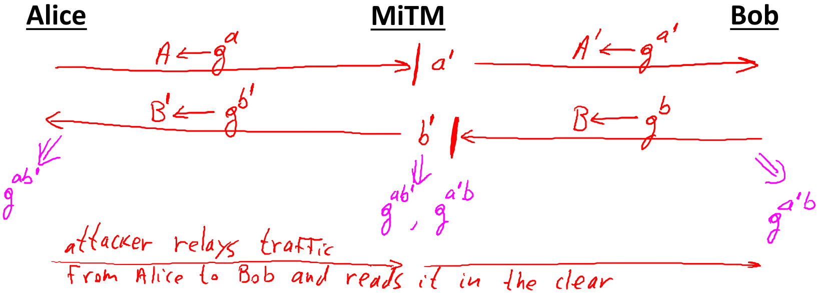
How hard is the DH function $\pmod{p}$?

- Suppose prime p is n bits long.
- Best known algorithm (GNFS): run time $\exp(O(\sqrt[3]{n}))$

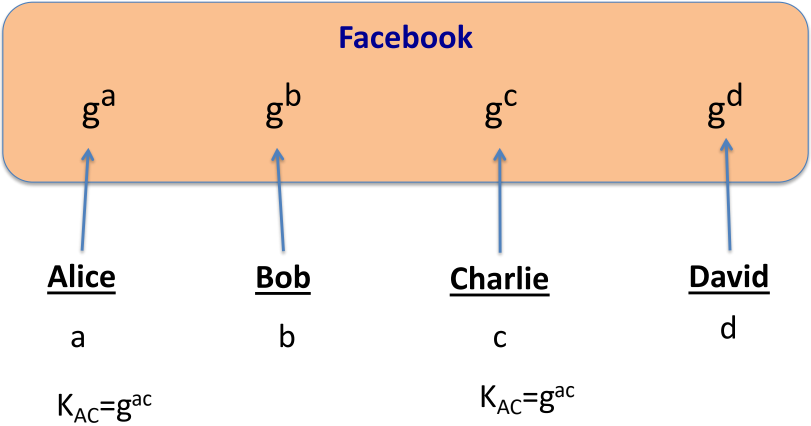
Cipher key size	Modulus size	Elliptic Curve size
80 bits	1024 bits	160 bits
128 bits	3072 bits	256 bits
256 bits (AES)	**15360** bits	512 bits

- As a result: slow transition away from $\pmod{p}$ to elliptic curves

Insecure against man-in-the-middle

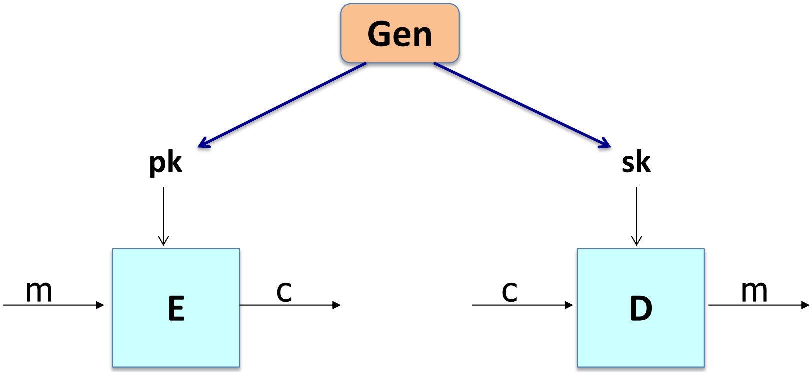


Another look at DH



Outline

Recap: public key encryption



Constructions

Previous lecture: based on trapdoor functions (such as RSA)

- Schemes: ISO standard, OAEP+, ...

This lecture: based on the Diffie-Hellman protocol

- Schemes: ElGamal encryption and variants (e.g. used in GPG)

The Diffie-Hellman protocol

Public parameter: g, p

Alice		Bob
$a \xleftarrow{\$} \{2, \ldots, p-2\}$		$b \xleftarrow{\$} \{2, \ldots, p-2\}$
$A = g^a$	$\xrightarrow{\text{A}}$	$B = g^b$
	$\xleftarrow{\text{B}}$	
$K = (B)^a = g^{ba}$		$K = (A)^b = g^{ab}$

ElGamal: converting to pub-key encyption

Public parameter: g, p

Alice		Bob
$a \xleftarrow{\$} \{2, \dots, p-2\}$		$b \xleftarrow{\$} \{2, \dots, p-2\}$
$A = g^a$		
	$\xrightarrow{A}$	
		$k = g^{ab} = A^b$
		$B = g^b$
		$c = m \cdot k$
	$\xleftarrow{(B,\ c)}$	
$K = B^a = g^{ba}$		
$m = c \cdot k^{-1}$		

The ElGamal Scheme: Idea

Let Alice have public key g^a and secret key a.

If Bob wants to encrypt m for Alice, he

- Picks b and computes $k = g^{ab} = (g^a)^b$
- Sends ciphertext $c = (g^b,\ m \cdot k)$ to Alice.

Alice can recompute $k = g^{ab} = (g^b)^a$ because

- g^b is in the received ciphertext
- a is her secret key

and she can decrypt $m = c \cdot k^{-1}$.

Exercise

Encrypt the following messages with the Elgamal scheme ($p = 467$ and $g = 2$):

1. secrete key $a = 105$ and $b = 213$ and message $m = 33$
2. secrete key $a = 105$ and $b = 123$ and message $m = 33$
3. secrete key $a = 300$ and $b = 45$ and message $m = 248$
4. secrete key $a = 300$ and $b = 47$ and message $m = 248$

ElGamal system (a modern view)

- G: finite cyclic group of order n
- $(\mathrm{E}_s, \mathrm{D}_s)$: symmetric encryption defined over (K, M, C);
- $\mathrm{H} : G^2 \to K$: a hash function.

We construct a pub-key encryption system $(\mathrm{G}, \mathrm{E}, \mathrm{D})$.

ElGamal system (a modern view)

Key generation G() :

- choose random generator g in G and random a in $\mathbb{Z}_n$
- output $\mathrm{sk} = a, \quad \mathrm{pk} = (g,\ h = g^a)$

Encryption E(pk = (g, h), m) :

$b \xleftarrow{\$} \mathbb{Z}_n,\ u = g^b,\ v = h^b,\ k = \mathrm{H}(u, v),\ c \leftarrow \mathrm{E}_s(k, m)$

return (u, c)

Decryption D(sk = a, (u, c)) :

$v = u^a,\ k = \mathrm{H}(u, v),\ m = \mathrm{D}_s(k, c)$

return m

ElGamal performance

Encryption & Decryption

$\mathsf{E}(pk = (g, h),\ m):$ $b \leftarrow\$\ \mathbb{Z}_n,\ u = g^b,\ v = h^b$	$\mathsf{D}(sk = a,\ (u, c)):$ $v = u^a$

Encryption: 2 exp. (fixed basis)

- Can pre-compute $\left\{g^{(2^i)},\ h^{(2^i)} \mid \text{for } i = 1, \ldots, \log_2 n\right\}$
- 3x speed-up (or more)

Decryption: 1 exp. (variable basis)

Computational Diffie-Hellman Assumption

- G : finite cyclic group of order n
- Computational Diffie-Hellman assumption holds in G if:

$$g,\ g^a,\ g^b\ \not\Rightarrow\ g^{ab}.$$

For all efficient algorithms A:

$$\Pr\Big[\ A(g,\ g^a,\ g^b) = g^{ab}\Big]\ <\ \text{negligible}$$

where $g \xleftarrow{\$} \{\text{ generators of } G\ \}, \quad a, b \xleftarrow{\$} \mathbb{Z}_n.$

VIỆN CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG
SCHOOL OF INFORMATION AND COMMUNICATION TECHNOLOGY

Thank you!

soict.hust.edu.vn/ fb.com/groups/soict