CT255
INTRODUCTION TO CYBERSECURITY
DIFFIE-HELLMAN KEY EXCHANGE

Dr. Michael Schukat

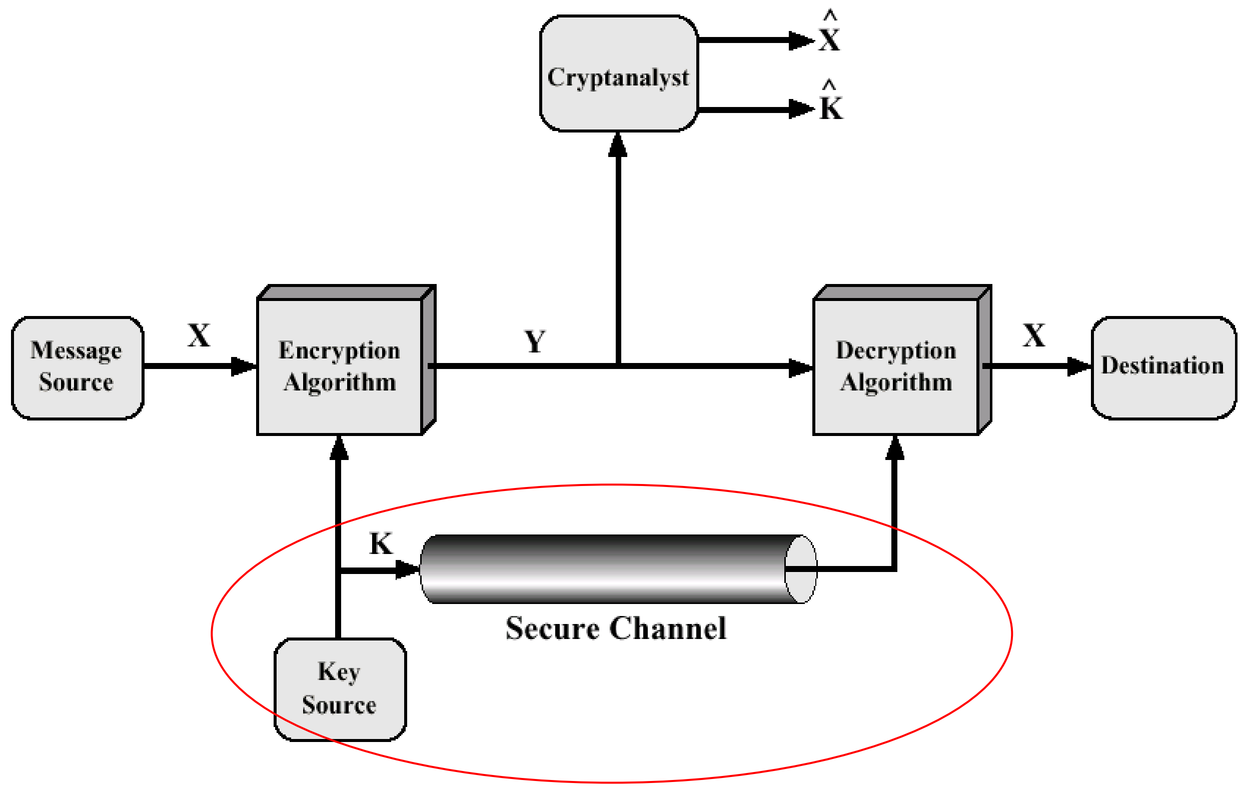
OÉ Gaillimh
NUI Galway

Lecture Content

- Diffie-Hellman Key exchange
- Man-in-the-Middle (MitM) attacks
- Optimisation techniques for public key encryption

Model of Conventional Cryptosystem

Problem: How to securely circulate a secret key?

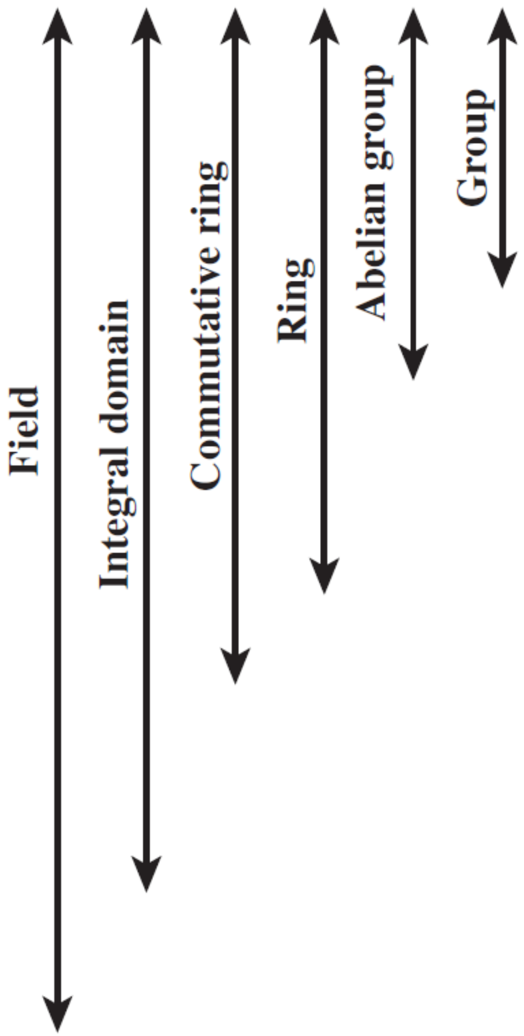


$$Y = E_K(X),\ X = E_K^{-1}(Y)$$

Groups, Rings and Fields (Wikipedia)

- In mathematics,
  - a **group** is a set equipped with a binary operation that is associative, has an identity element, and is such that every element has an inverse, e.g. (Z, +)
  - a **ring** is a set equipped with two binary operations satisfying properties analogous to those of addition and multiplication of integers, e.g. (Z, +, *)
  - a **field** is a set on which addition, subtraction, multiplication, and division are defined and behave as the corresponding operations on rational and real numbers do

Properties of Groups, Rings and Fields (Stallings)



(A1) Closure under addition:	If a and b belong to S, then $a + b$ is also in S
(A2) Associativity of addition:	$a + (b + c) = (a + b) + c$ for all a, b, c in S
(A3) Additive identity:	There is an element 0 in R such that $a + 0 = 0 + a = a$ for all a in S
(A4) Additive inverse:	For each a in S there is an element $-a$ in S such that $a + (-a) = (-a) + a = 0$
(A5) Commutativity of addition:	$a + b = b + a$ for all a, b in S
(M1) Closure under multiplication:	If a and b belong to S, then ab is also in S
(M2) Associativity of multiplication:	$a(bc) = (ab)c$ for all a, b, c in S
(M3) Distributive laws:	$a(b + c) = ab + ac$ for all a, b, c in S $(a + b)c = ac + bc$ for all a, b, c in S
(M4) Commutativity of multiplication:	$ab = ba$ for all a, b in S
(M5) Multiplicative identity:	There is an element 1 in S such that $a1 = 1a = a$ for all a in S
(M6) No zero divisors:	If a, b in S and $ab = 0$, then either $a = 0$ or $b = 0$
(M7) Multiplicative inverse:	If a belongs to S and $a \neq 0$, there is an element a^{-1} in S such that $aa^{-1} = a^{-1}a = 1$

Modular Arithmetic

- In mathematics, modular arithmetic is a system of arithmetic for integers, where numbers wrap around when reaching a certain value n, called the modulus
  - Recall modulus operator "%" in C and other languages, i.e. "division with rest" with rest being the modulus
  - Example: 75 / 6 = 12 remainder 3 ➔ 75 % 6 = 3
- The ring of integers modulo n, denoted Z/nZ or Z/n
- Z/nZ is defined for n > 0 as: $\mathbb{Z}/n\mathbb{Z} = \{\overline{a}_n \mid a \in \mathbb{Z}\} = \left\{\overline{0}_n, \overline{1}_n, \overline{2}_n, \ldots, \overline{n-1}_n\right\}$
- With:
  - $\overline{a}_n + \overline{b}_n = \overline{(a+b)}_n$
  - $\overline{a}_n - \overline{b}_n = \overline{(a-b)}_n$
  - $\overline{a}_n \overline{b}_n = \overline{(ab)}_n.$

Example: Normal Multiplication

*****	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
0	0	0	0	0	0	0	0	0	0
1	0	1	2	3	4	5	6	7	8
2	0	2	4	6	8	10	12	14	16
3	0	3	6	9	12	15	18	21	24
4	0	4	8	12	16	20	24	28	32
5	0	5	10	15	20	25	30	35	40
6	0	6	12	18	24	30	36	42	48
7	0	7	14	21	28	35	42	49	56
8	0	8	16	24	32	40	48	56	64

Example: Multiplication Z/9Z

Mx3

*	0	1	2	3	4	5	6	7	8
0	0	0	0	0	0	0	0	0	0
1	0	1	2	3	4	5	6	7	8
2	0	2	4	6	8	1	3	5	7
3	0	3	6	0	3	6	0	3	6
4	0	4	8	3	7	2	6	1	5
5	0	5	1	6	2	7	3	8	4
6	0	6	3	0	6	3	0	6	3
7	0	7	5	3	1	8	6	4	2
8	0	8	7	6	5	4	3	2	1

Diffie-Hellman Key Exchange

- **Diffie-Hellman provides secure key exchange between two partners**
  - The negotiated key is subsequently used for private key encryption / authentication
- It uses the **multiplicative group of integers modulo n $(Z/nZ)^x$**
- It is based on the difficulty of computing discrete logarithms over such groups, e.g.

$$6^3 \bmod 17 = 216 \bmod 17 = 12 \qquad \text{(easy)}$$

$$12 = 6^y \bmod 17? \qquad \text{(difficult)}$$

- It uses modulo n ("division with rest") operation.
- The core equation for the key exchange is

$$K = (A)^B \bmod q$$

Diffie-Hellman: Global Public Elements

- Select prime number q and positive integer a, whereby $a < q$ and a is a **primitive root** of q.

- Definition: a is a primitive root of q, if numbers
  $a \bmod q,\ a^2 \bmod q,\ \cdots\ a^{(q-1)} \bmod q$
  are distinct integer values between 1 and $(q - 1)$ in some permutation, i.e. elements of $(Z/qZ)^x$

M

- Example: $a = 3$ is a primitive root of $(Z/5Z)^x$, $a = 4$ is not:

$3^1 = 3 = 0 * 5 + \mathbf{3}$	$4^1 = 4 = 0 * 5 + \mathbf{4}$
$3^2 = 9 = 1 * 5 + \mathbf{4}$	$4^2 = 16 = 3 * 5 + \mathbf{1}$
$3^3 = 27 = 5 * 5 + \mathbf{2}$	$4^3 = 64 = 12 * 5 + \mathbf{4}$
$3^4 = 81 = 16 * 5 + \mathbf{1}$	$4^4 = 256 = 51 * 5 + \mathbf{1}$

Generation of Secret-Key: Part 1

- Both users share a (public) prime number q and primitive root a
- User A:
  - Select secret number XA with XA < q
  - Calculate public value $YA = a^{XA} \bmod q$ (← difficult to reverse)
  - YA is send to user B
- User B:
  - Select secret number XB with XB < q
  - Calculate public value $YB = a^{XB} \bmod q$ (← difficult to reverse)
  - YB is send to user A

Generation of Secret-Key: Part 2

- User A:
  - User A owns X_A and receives Y_B
  - Generate secret key: $K = (Y_B)^{X_A} \bmod q$
- User B:
  - User B owns X_B and receives Y_A
  - Generate secret key: $K = (Y_A)^{X_B} \bmod q$
- **Both keys are identical!**

Generation of Secret-Key: Part 2

$K = (YB)^{XA} \bmod q$

$= (a^{XB} \bmod q)^{XA} \bmod q$

$= (a^{XB})^{XA} \bmod q$

$= a^{XB\ XA} \bmod q = a^{XA\ XB} \bmod q$

$= (a^{XA})^{XB} \bmod q$

$= (a^{XA} \bmod q)^{XB} \bmod q$

$= (YA)^{XB} \bmod q$

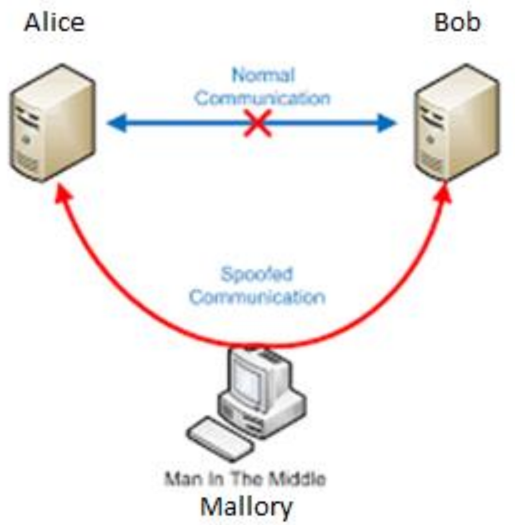
Example for Diffie-Hellman

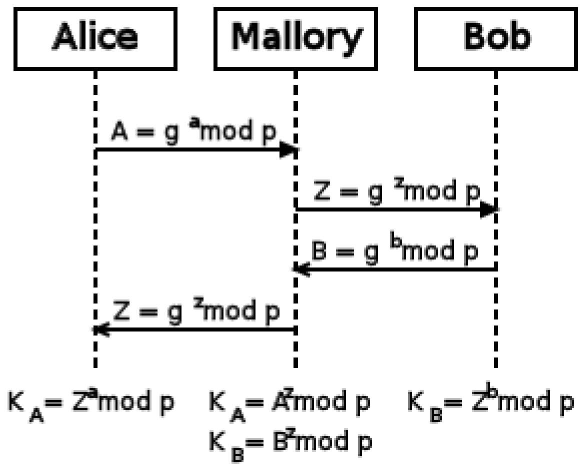
- Let $q = 5$ and $a = 3$;
- $XA = 2$, therefore $YA = a^{XA} \bmod 5 = 4$
- $XB = 3$, therefore $YB = a^{XB} \bmod 5 = 2$
- User A: $K = (YB)^{XA} \bmod q = 2^2 \bmod 5 = 4$
- User B: $K = (YA)^{XB} \bmod q = 4^3 \bmod 5 = 4$

Diffie-Hellman in Practice

- The algorithm is used in tandem with a variety of secure network protocols
  - Provision of secure end-to-end connection
  - No endpoint authentication though!
    - You can't validate who you are talking to
  - Modulus p typically has a minimum length of 1024 bits

DH and Man-in-the-Middle (MitM) Attacks





- Mallory is a MitM attacker and performs message interception and message fabrication
- Mallory establishes two individual (secure) connections with Alice and Bob
- Both Alice and Bob are unaware of Mallory's existence (as there is no authentication)

In-Class Activity: Diffie-Hellman MitM Attack

- Let q = 5 and a = 3;
- $X_{Alice} = 2$, therefore $Y_{Alice} = a^{XAlice} \bmod 5 = 4$
- $X_{Bob} = 3$, therefore $Y_{Bob} = a^{XBob} \bmod 5 = 2$
- $X_{Malory} = 1$, therefore $Y_{Malory} = a^{XMalory} \bmod 5 = 3$
- What session keys between
  - Alice and Malory
  - Malory and Bob

  are generated?
- Note: User A's key $K = (YB)^{XA} \bmod q$
- Note: User B's key $K = (YA)^{XB} \bmod q$

Mx2

Solution

- Alice sends "4" to Bob, but this message is intercepted by Malory
- Bob sends "2" to Alice, but this message is intercepted by Malory
- Malory sends "3" to both parties, claiming to be either Bob or Alice
- Alice receives "3" and calculates K as follow: $K = 3^2 \bmod 5 = 4$
  - Malory calculates $4^1 \bmod 5 = 4$
- Bob receives "3" and calculates K as follow: $K = 3^3 \bmod 5 = 2$
  - Malory calculates $2^1 \bmod 5 = 2$
- Alice and Bob think they just mutually agreed on a shared secret key
- They have no idea that Malory is a MitM and can read, manipulate and fabricate messages between both sides

Computational Aspects of Diffie-Hellman

- Assume you have to evaluate the expression $C = 503^{23} \bmod 899$ as part of the DH algorithm
- $503^{23} = 1.3679293137954084232504397101 06 \times 10^{62}$ cannot be properly represented using an ordinary integer or floating point variable!
- In order to solve this problem the exponentiation must be broken down into smaller steps, e.g.
  - $503^{23} \bmod 899 = ((503^{6} \bmod 899) \times (503^{6} \bmod 899) \times (503^{6} \bmod 899) \times (503^{5} \bmod 899)) \bmod 899$
  - $503^{6} \bmod 899 = ((503^{3} \bmod 899) \times (503^{3} \bmod 899)) \bmod 899$
  - $503^{5} \bmod 899 = ((503^{3} \bmod 899) \times (503^{2} \bmod 899)) \bmod 899$
  - $503^{3} \bmod 899 = ((503^{2} \bmod 899) \times 503) \bmod 899$

Computational Aspects of Diffie-Hellman

- or even iteratively:
  $503^{23} \bmod 899 =$
  $((((((503^2 \bmod 899) \times 503) \bmod 899) \times 503) \bmod 899) \times \cdots \times 503) \bmod 899$

- This expression consists of 22 nested multiplications and 22 nested modulus operations and can be easily calculated by using a loop