Diffie-Hellman Key Exchange

CS 5158/6058 Data Security and Privacy

Spring 2018

Instructor: Boyang Wang

Symmetric-Key Exchange

- Share a secret key between Alice and Bob
  - Alice and Bob meet in advance at a secure place (e.g., Starbucks)
- If they have a secure place, why not exchange private message at secure place (to avoid sharing keys)?
  - Alice/Bob cannot go to Starbucks all the time
  - Starbucks is not open all the time

Symmetric-Key Exchange

- Each pair of two users should have a different key
  - Messages should be private between two users

- The total number of keys in a system is high
  - 2 users: Alice, Bob
    - 1 key: A <—> B
  - 3 users: Alice, Bob, Charlie
    - 3 keys: A <—> B, A <—> C,
      B <—> C

Symmetric-Key Exchange

- The total number of keys in a system is high
  - 4 users: Alice, Bob, Charlie, David
    - 6 keys: A <—> B, A <—> C, A <—> D,
      B <—> C, B <—> D,
      C <—> D
  - n users: 1st user needs (n-1) keys,
    2nd user needs another (n-2) keys,
    3rd user needs another (n-3) keys, …
  - No. of keys: (n-1) + (n-2) + … + 1 = (n)(n-1)/2

Symmetric-Key Exchange

- The total number of keys in a system is high
  - n users: (n)(n-1)/2 keys
- The cost to establish/share all the keys is high
  - E.g., n = 60 in this class, 60*59/2 = 1770 keys
  - Share each key at Starbucks
    - Each one costs $10, $10*1770=$17700
    - no wonder Starbucks is rich!

Symmetric-Key Exchange

- The total number of keys in a system is high
  - n users: (n)(n-1)/2 keys
  - Each user needs to maintain n-1 keys

- The cost to maintain all the keys is high
  - E.g., n = 60 in this class
  - Each user needs to maintain 59 keys
    - Keep all those keys secret
    - Synchronize among PC, laptop, smartphone

Symmetric-Key Exchange

- The total number of keys in a system is high
  - n users: (n)(n-1)/2 keys
  - Each user needs to maintain n-1 keys

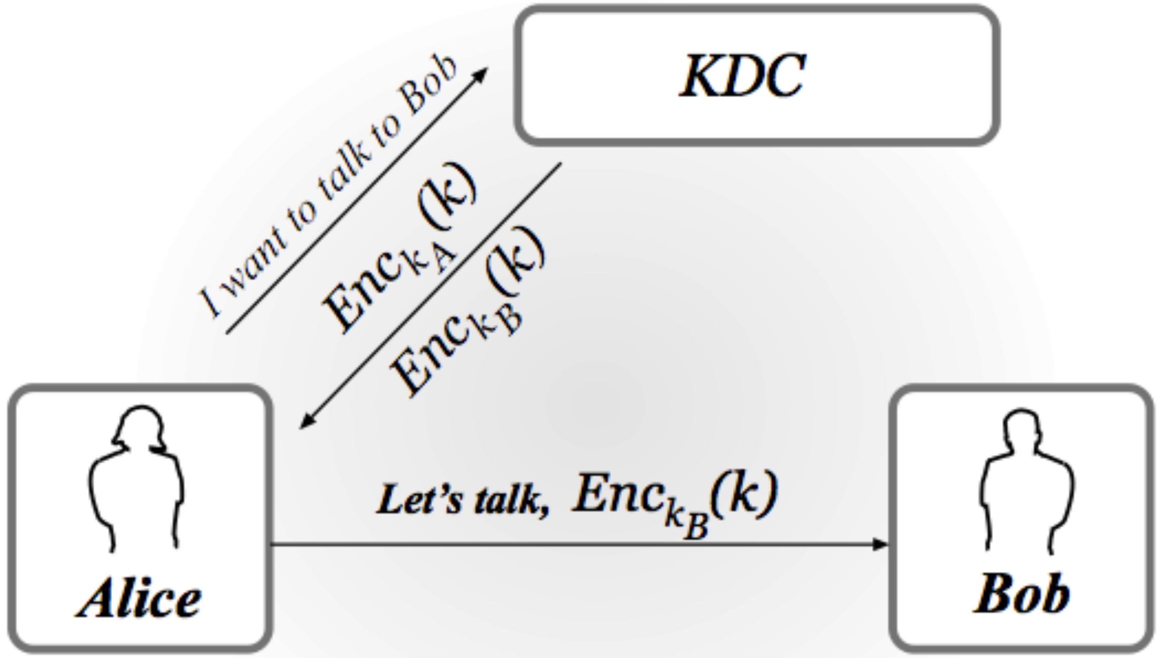
- <u>Example:</u> UC has 40,000 students, the total number of keys? how many keys for each user?
  - 40000*(39999)/2 = 799 million keys
  - Each key costs $10 at Starbucks, 8 billion dollars
  - Each student maintains 39999 keys

Key Distribution Center

- All the users trust a same entity (KDC):

- Each user only needs one secret key with KDC
  - Alice only has 1 key with KDC
  - Bob only has 1 key with KDC

- Alice & Bob need to exchange private messages?
  - KDC helps two users establish a session key
    - Session key: short-term, easy to replace



- k_A: Alice <—> KDC; k_B: Bob <—> KDC
    - Alice to KDC: I want to talk to Bob
    - KDC generates a session key k
    - KDC to Alice: $Enc_{kA}(k)$, $Enc_{kB}(k)$
    - Alice decrypts $Enc_{kA}(k)$, obtains session key k
    - Alice to Bob: Let's talk, $Enc_{kB}(k)$
    - Bob decrypts $Enc_{kB}(k)$, obtains session key k

Key Distribution Center

- Alice and Bob use session key k to talk
  - KDC deletes session key after sending it to Alice
  - Alice & Bob delete session key after they talk
  - Want talk again tomorrow? Get a new session key from KDC

- Total: n (long-term) secret keys in the system
  - Each user: 1 secret key with KDC
  - KDC: n secret keys

Key Distribution Center

- The total number of keys in a system is lower
  - n users: n keys (v.s. n(n-1)/2)

- The cost to establish/share all the keys is lower
  - E.g., n = 60 in this class, 60 keys
  - Share each key at Starbucks
    - Each one costs $10, total $600 (v.s.$17700)
- The cost to maintain all the keys is lower
  - Each user only maintains 1 key (v.s. 59 keys)

Limitation of KDC

- Everything depends on KDC
  - Security: if KDC is compromised, all comm. are not secure; KDC becomes a popular target
  - Performance: all the requests for session keys need to go though KDC; if KDC is down, the entire system is down

- Using multiple KDCs is better
  - Synchronization requires more costs

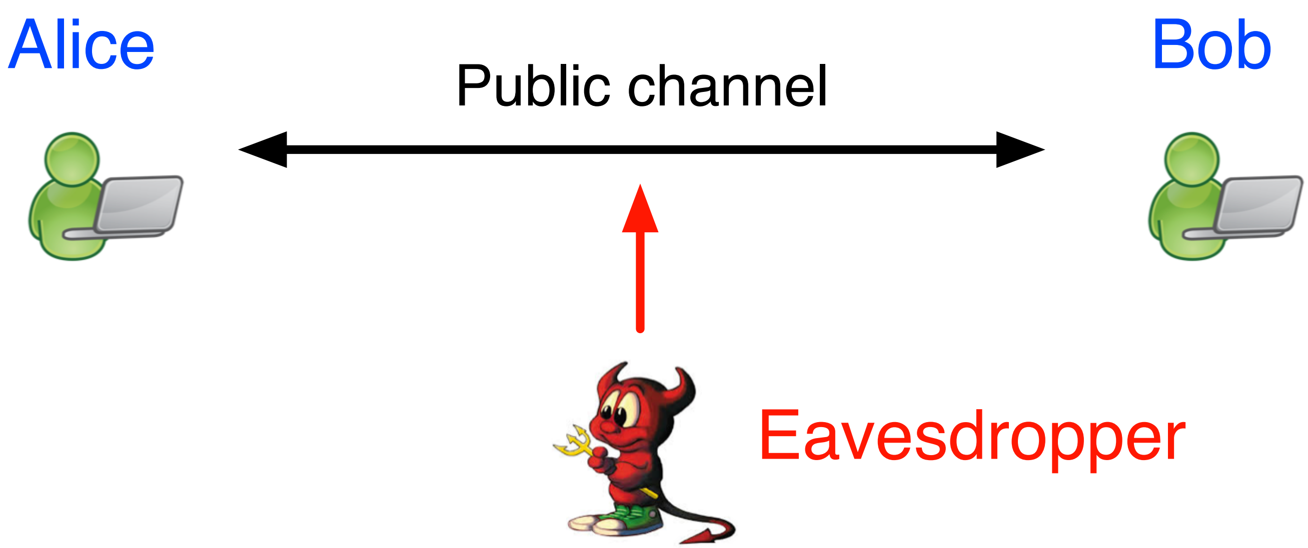
Public-Key Revolution

- Public-Key Revolution (Diffie and Hellman, 1976)
  - "*New Direction in Cryptography*"
  - Idea: No need to share a private key

  - Did not propose a detailed encryption scheme
  - Proposed a key-exchange protocol on public channel
    - Diffie-Hellman Key Exchange

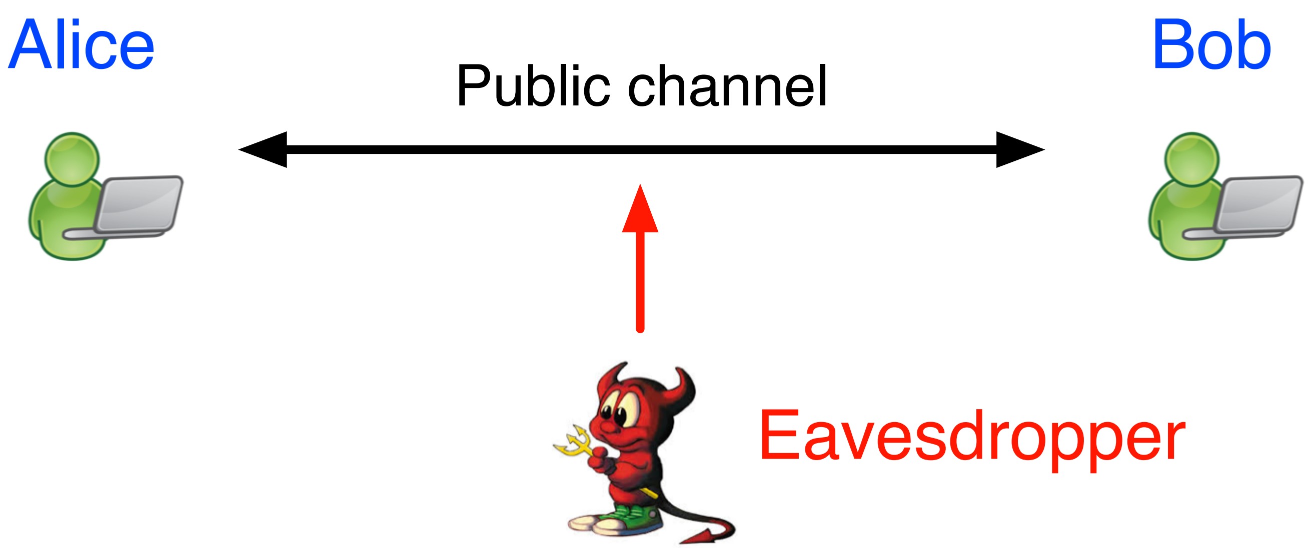
Diffie-Hellman Key Exchange

- Alice & Bob do not have a private channel
- Alice & Bob establish a secret key using a public channel
    - Will use this secret key for later encryption



Diffie-Hellman Key Exchange

- No secret keys need to be shared at secure location
  - If we have 60 students in a system
  - Total cost at Starbucks $0 (v.s.$17700, v.s. $600)



Cyclic Group

- Group G (defined under an operation)
  - Group order p: there are p elements in G
  - Set Z_6={0, 1, 2, 3, 4, 5} is an additive group
  - Set Z_6^*={1, 5} is a multiplicative group

- For any g in group G, <u>group order</u> is p
  - Identity is 1 (i.e., a multiplicative group)
  - Order of element g
    - is the smallest <u>positive</u> integer a, s.t., $g^a = 1$

Cyclic Group

- Order of element g
  - is the smallest <u>positive</u> integer s.t., $g^a = 1$

- Example: $Z_6^* = \{1, 5\}$, group order is 2
  - element 1:
    - $1^1 = 1 \bmod 6$, order of element 1 is 1
  - element 5:
    - $5^1 = 5 \,!= 1 \bmod 6$
    - $5^2 = 1 \bmod 6$, order of element 5 is 2

Cyclic Group

- Order of element g
  - is the smallest positive integer s.t., $g^a = 1$
- Practice: $Z_{10}^* = \{1,3,7,9\}$, group order is 4
  - What is the order of element 1?
  - What is the order of element 3?
    - $1^1 = 1 \bmod 10$, order of element 1 is 1
    - $3^1 = 3 \,!= 1 \bmod 10$
    - $3^2 = 9 \,!= 1 \bmod 10$
    - $3^3 = 27 = 7 \,!= 1 \bmod 10$
    - $3^4 = 81 = 1 \bmod 10$, order of element 3 is 4

Cyclic Group

- If the order of element g is equal to group order p
  - This element g is a generator of group G

- Example: $Z_6^*=\{1, 5\}$, group order is 2
  - element 1:
    - $1^1 = 1 \bmod 6$, order of element 1 is 1
    - element 1 is not a generator of Z_6^*
  - element 5:
    - $5^2 = 25 = 1 \bmod 6$, order of element 5 is 2
    - element 5 is a generator of Z_6^*

Cyclic Group

- If the order of element g is equal to group order p
  - This element g is a generator of group G

- Practice: Z_{10}^* = {1,3,7,9}, group order is 4
  - $1^1 = 1$ mod 10, order of element 1 is 1
  - $3^4 = 81 = 1$ mod 10, order of element 3 is 4
  - Is element 1 a generator?
  - Is element 3 a generator?

Cyclic Group

- If the order of element g is equal to group order p
  - This element g is a generator of group G
    - $<g> = \{g^0, g^1, \ldots, g^{p-1}\}$ has all elements in G

- Example: $Z_6^*=\{1, 5\}$, group order is 2
  - element 5: order of element 5 is 2
    - 5 is a generator of Z_6^*
    - $<5> = \{5^0, 5^{p-1}\} = \{5^0, 5^{2-1}\} = \{1, 5\}$
    - $<5>$ has all the elements in Z_6^*

Cyclic Group

- If the order of element g is equal to group order p
  - This element g is a generator of group G
    - $<g> = \{g^0, g^1, \ldots, g^{p-1}\}$ has all elements in G

- Example: $Z_{10}^* = \{1,3,7,9\}$, group order is 4
  - order of element 3 is 4
    - 3 is a generator of Z_{10}^*
    - $<3> = \{3^0, 3^1, 3^2, 3^{p-1}\} = \{1, 3, 9, 7\}$
    - $<3>$ has all the elements in Z_{10}^*

Cyclic Group

- G is a cyclic group if there is a generator in G
- Example: $Z_6^*=\{1, 5\}$, group order is 2
  - order of element 5 is 2, 5 is a generator of Z_6^*
- Example: $Z_{10}^* = \{1,3,7,9\}$, group order is 4
  - order of element 3 is 4, 3 is a generator of Z_{10}^*
- Z_6^* and Z_{10}^* are both cyclic groups

Cyclic Group

Thm: If p+1 is a prime, Z_{p+1}^* is a cyclic group & group order is p

- $Z_5^* = \{1,2,3,4\}$ is a (multiplicative) group
  - 5 is a prime, Z_5^* is cyclic group, order p=4
- Example: Find the generator(s) of Z_5^*
  - Order of element 1 is 1: $1^1 = 1 \bmod 5$
  - Order of element 2 is 4: $2^4 = 1 \bmod 5$
  - Order of element 3 is 4: $3^4 = 1 \bmod 5$
  - Order of element 4 is 2: $4^2 = 1 \bmod 5$
  - Generators: 2, 3

Cyclic Group

- Cyclic group G, generator g, group order p,
  - $<g> = \{g^0, g^1, \ldots, g^{p-1}\}$ is a permutation of G
  - For any h in G, $h=g^x$ for a unique x in $\{0, \ldots, p-1\}$

- Example: $Z_{10}^* = \{1,3,7,9\}$, group order is 4
  - order of element 3 is 4,
    - 3 is a generator of Z_{10}^*
    - Z_{10}^* is cyclic group
    - $<3> = \{3^0, 3^1, 3^2, 3^{p-1}\} = \{1, 3, 9, 7\}$
    - $<3>$ is a permutation of Z_{10}^*

Cyclic Group

- Cyclic group G, generator g, group order p,
  - <g> = $\{g^0, g^1, \ldots, g^{p-1}\}$ is a <u>permutation</u> of G
  - For any h in G, $h=g^x$ for a <u>unique</u> x in {0, …, p-1}

- <u>Example:</u> Z_5^* = {1,2,3,4}, group order is p=4
  - Z_5^* is cyclic, since 5 is a prime
  - order of element 2 is 4,
    - 2 is a generator
    - <2> = $\{2^0, 2^1, 2^2, 2^3\}$ = {1, 2, 4, 3}
    - <2> is a permutation of Z_5^*

Discrete-Logarithm Problem

- Cyclic group G, generator g, group order p
  - For any h in G, $h=g^x$ for a unique x in {0, …, p-1}

- Given g and x, compute h is easy
- Discrete-Logarithm Problem (DL)
  - If **p is a large integer**, given h and g, compute $x = \log_g(h)$ is hard
  - DL is a one-way function

Discrete-Logarithm Problem

- Given g and x, compute h is easy
- Discrete-Logarithm Problem (DL)
    - If **p is a large integer**, given h and g, compute $x = \log_g(h)$ is hard

- Example: Z_{131}^*, 131 is a prime,
    - Given g=100 and $h=g^x=44$, what is x???
    - Given g=100 and x=2, compute $h = g^x = 100^2 = 44 \bmod 131$

Discrete-Logarithm Problem

Discrete-logarithm experiment $\mathsf{DLog}_{\mathcal{A},\mathbb{G}}(n)$

1. Given 1^n, obtain $(\mathbb{G}, p, g)$, where $\mathbb{G}$ is a cyclic group with order p (p is n-bit), and g is a generator of $\mathbb{G}$.
2. Choose a uniform $h \in \mathbb{G}$
3. Adversary $\mathcal{A}$ is given $\mathbb{G}$, p, g, h, and outputs $x \in \mathbb{Z}_p$
4. Experiment outputs 1 iff $g^x = h$

For any PPT $\mathcal{A}$, $\Pr[\mathsf{DLog}_{\mathcal{A},\mathbb{G}}(n) = 1] \leq \mathtt{negl}(n)$

- Diffie-Hellman Key Exchange
- Alice outputs public parameters (G, p, g)
  - G is cyclic, generator g, group order p

Alice		Bob
choose a uniform $x \in \mathbb{Z}_p$, compute $h_A = g^x$	$\xrightarrow{h_A}$	choose a uniform $y \in \mathbb{Z}_p$, compute $h_B = g^y$
	$\xleftarrow{h_B}$	
compute $k_A = h_B^x = g^{xy}$		compute $k_B = h_A^y = g^{xy}$

  - Alice and Bob share a same key g^{xy}

Alice | **Bob**

choose a uniform $x \in \mathbb{Z}_p$, compute $h_A = g^x$

$$\xrightarrow{\quad h_A \quad}$$

choose a uniform $y \in \mathbb{Z}_p$, compute $h_B = g^y$

$$\xleftarrow[\quad h_B \quad]{}$$

compute $k_A = h_B^x = g^{xy}$

compute $k_B = h_A^y = g^{xy}$

- Example: $Z_7^* = \{1, 2, 3, 4, 5, 6\}$
  - Group order p = 6, generator g = 3
  - Alice chooses x = 3 and Bob chooses y = 5
  - what is h_A = ?, h_B = ?, $k_A = k_B$ =? in DH protocol

Alice		Bob
choose a uniform $x \in \mathbb{Z}_p$, compute $h_A = g^x$	$\xrightarrow{\quad h_A \quad}$	choose a uniform $y \in \mathbb{Z}_p$, compute $h_B = g^y$
	$\xleftarrow[\quad h_B \quad]{}$	
compute $k_A = h_B^x = g^{xy}$		compute $k_B = h_A^y = g^{xy}$

- <u>Example</u>: $Z_7^* = \{1, 2, 3, 4, 5, 6\}$
    - p = 6, g = 3, choose x = 3 and y = 4
    - $h_A = g^x = 3^3 = 6 \bmod 7$
    - $h_B = g^y = 3^4 = 4 \bmod 7$
    - $k_A = h_B^x = 4^6 = 1 \bmod 7$

Alice		Bob
choose a uniform $x \in \mathbb{Z}_p$, compute $h_A = g^x$	$\xrightarrow{h_A}$	choose a uniform $y \in \mathbb{Z}_p$, compute $h_B = g^y$
	$\xleftarrow[h_B]{}$	
compute $k_A = h_B^x = g^{xy}$		compute $k_B = h_A^y = g^{xy}$

- Practice: $Z_{11}^* = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$
  - Group order p = 10, generator g = 2
  - Alice chooses x = 3 and Bob chooses y = 9
  - what is h_A = ?, h_B = ?, $k_A = k_B$ =? in DH protocol

Alice | **Bob**

choose a uniform $x \in \mathbb{Z}_p$, compute $h_A = g^x$

$$\xrightarrow{\quad h_A \quad}$$

choose a uniform $y \in \mathbb{Z}_p$, compute $h_B = g^y$

$$\xleftarrow{\quad h_B \quad}$$

compute $k_A = h_B^x = g^{xy}$

compute $k_B = h_A^y = g^{xy}$

- <u>Practice</u>: $Z_{11}^* = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$
  - p = 10, g = 2, choose x = 3 and y = 9
  - $h_A = g^x = 2^3 = 8 \bmod 11$
  - $h_B = g^y = 2^9 = 6 \bmod 11$
  - $k_A = h_B^x = 6^3 = 7 \bmod 11$

Security of DH Protocol

- If Discrete-Logarithm problem is easy, DH is not secure
  - Eavesdropper has $h_A = g^x$ and $h_B = g^y$
  - Computes $x = \log_g(h_A)$ and $y = \log_g(h_B)$
  - Obtains key $k = g^{xy}$
  - DL is hard is necessary, but not sufficient

- Computational Diffie-Hellman Problem (CDH)
  - Given g^x and g^y, compute g^{xy} is hard

Security of DH Protocol

- If CDH problem is easy, DH is not secure
  - Given $h_A = g^x$ and $h_B = g^y$
  - Adversary computes key $k = g^{xy}$
  - CDH is hard is necessary, but still not sufficient

- Decisional Diffie-Hellman Problem (DDH)
  - Given g^x, g^y and a random element h in G, decide whether $h ?= g^{xy}$ is hard
- DH protocol is secure if DDH problem is hard

Security of DH Protocol

- Computational Diffie-Hellman Problem (CDH)
  - Given g^x and g^y, compute g^{xy} is hard
- Decisional Diffie-Hellman Problem (DDH)
  - Given g^x, g^y and a random element h in G, decide whether $h ?= g^{xy}$ is hard

- True: DH protocol is secure if DDH problem is hard
- False: DH protocol is secure if CDH problem is hard
- False: DH protocol is secure if DL problem is hard

Additional Reading

Chapter 10, *Introduction to Modern Cryptography, Drs. J. Katz and Y. Lindell, 2nd edition*