Key Exchange

- CEG 6430/4430 Cyber Network Security

- Junjie Zhang
- junjie.zhang@wright.edu
- Wright State University

Key Exchange

To allow two parties to establish a shared secret/key over an unsecured channel.

Wait...

Didn't you say

Use RSA to Share A Symmetric-Encryption Key

- The sender randomly generates a symmetric secret key.
- The sender encrypts this secret key using the reciever public key.
- The receiever decrypts the ciphertext using its private key.
- Bulk data can not be encrypted using the symmetric secret key (i.e., using a mode of operation).

Why bother sovling a solved probem?

Well...Not Fully Solved

- Yes and we indeed use use RSA to exchange symmetric keys in practice (e.g., as an option of SSL/TLS).
- But it does not have **forward secrecy**, which protects past sessions against future compromises of keys or passwords.
  - An attacker keeps all encrypted traffic sent to the receiver.
  - She cannot break the private key now. But she might be able to do it in the future.
  - If she breaks the private key, she can recover all symmetric keys and then use them to further decrypt all exchanged ciphertexts.

Diffie-Hellman

DH alows two parties to establish a shared secret/key over an unsecured channel.

- The shared key is for each session (e.g., for a short period).
- Clarifications of the unsecured channel: an attacker who can intercept the communication is indeed able to break DH via MITM attacks.

Diffie-Hellman

- Alice and Bob agree upon two non-secret numbers, P, the prime number and G, the generator of P.
  - P and G can be exchanged in plaintext.
- Alice and Bob indepdently and randomly generated their private keys, s_A and s_B, respectively.

Diffie-Hellman

- Each generates his/her public key
  - $p_A = G^{s_A} \ mod \ P.$
  - $p_B = G^{s_B} \ mod \ P.$
- Exchange public keys.
- Calculate the shared secret
  - Alice: $p_B^{s_A} \ mod \ P.$
  - Bob: $p_A^{s_B} \ mod \ P.$

Diffie-Hellman

An example:

- $P = 13, G = 6$
- $s_A = 5, b_A = 4$
- $p_A = 6^5 \ mod \ 13$ (i.e., 2), $p_B = 6^4 \ mod \ 13$ (i.e., 9)
- Shared Secret:
    - Alice: $9^5 mod 13$, which is 3
    - Bob: $2^4 mod 13$, which is 3

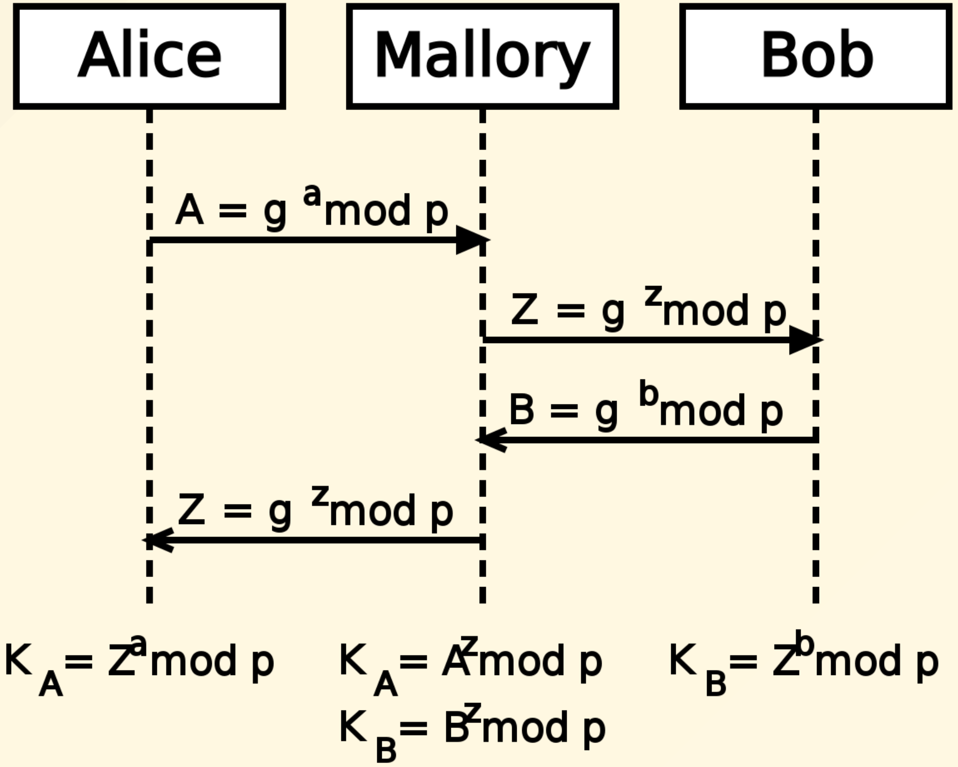
Diffie-Hellman

DH's security is rooted in **Discrete Logarithm**. $G^X \ mod \ P = N$.

- Given G, P, and X, it is easy to find N.
- Given G, P, and N, it is hard to find X.

DH is vulnerable to MITM

DH is vulnerable to man-in-the-middle (MITM) attack, where an attacker can intercept the communication between Alice and Bob and simultaneously impersonate them.



Then we have a problem

Q: An attacker can indeed intercept the communication. Correct?
A: Yes.

Q: Then DH is useless. Correct?
A: No.

Then we have a problem (cont.)

Q: How come?
A: Use RSA.

- If Bob knows Alice's RSA public key, Alice can use digital signature to protect the integrity of $p_A = G^{s_A} \ mod \ P$.
  - Alice $signature = sign(p_A, s_{RSA,Alice})$
  - Bob $verify(p_A, signature, p_{RSA,Alice})$

RSA + DH

Advantages:

- It counteracts the MITM attack since Mallory does not have Alice's RSA private key.
- Even if Alice's RSA private key is broken/stolen, the attacker cannot recover the session key exchanged by DH.