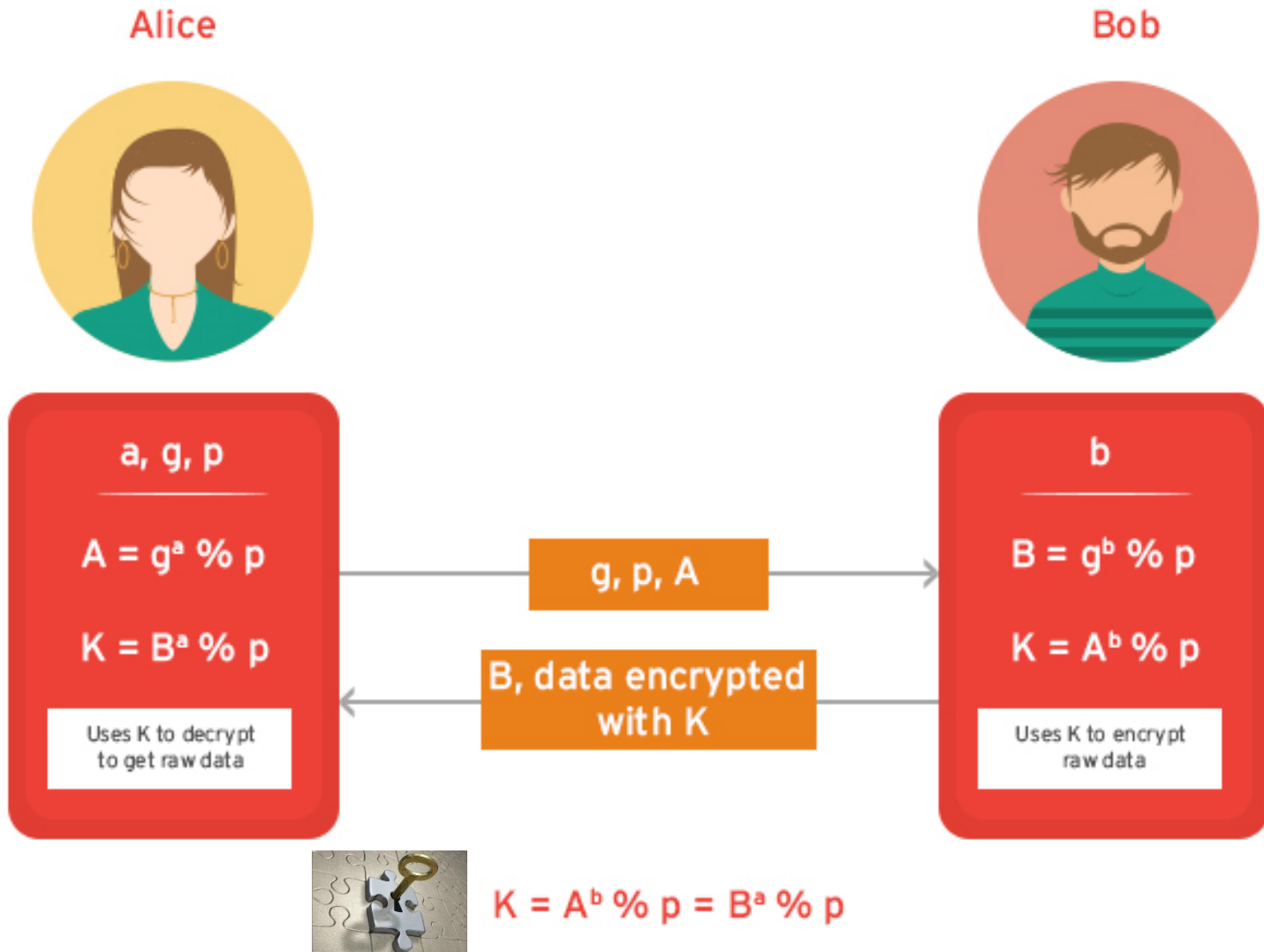


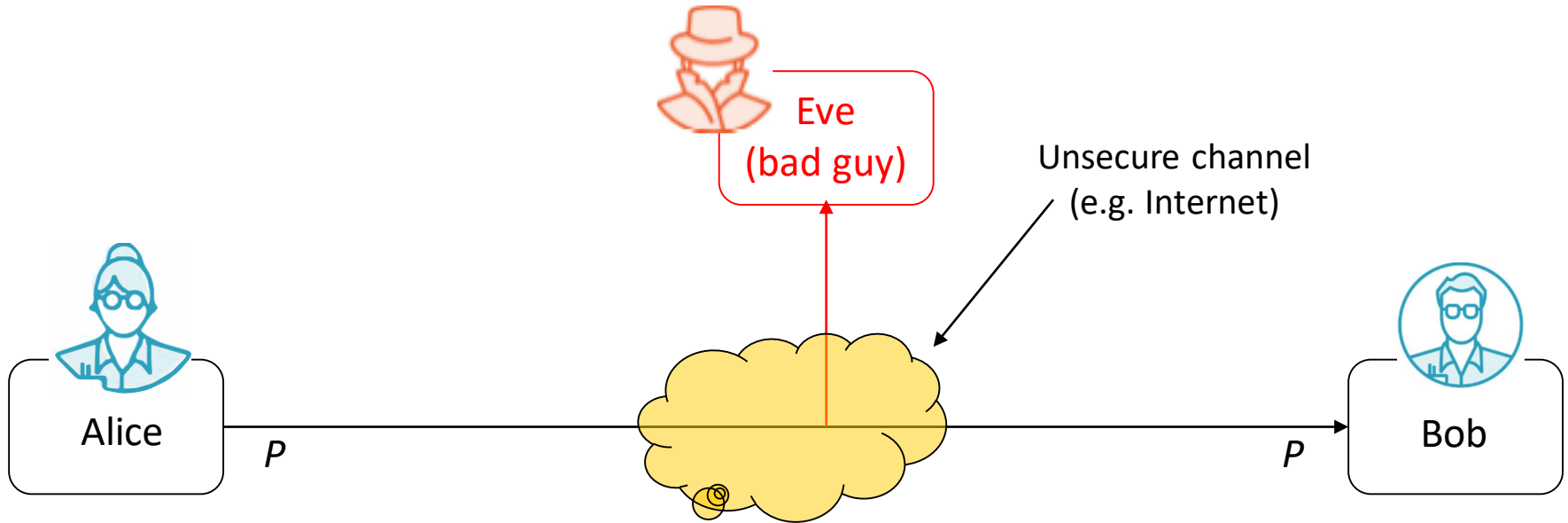
Diffie Hellman Key Exchange



Symmetric Key Crypto Problem

- Symmetric key crypto lets two parties exchange secret messages *as long as they already have a shared key*
- How do you exchange secret messages with someone when you don't already have a shared key?

Symmetric Cryptography



- **Problem Statement:**

- 1) Alice and Bob would like to communicate via an unsecure channel (e.g., Internet)
- 2) A malicious third party Eve (the bad guy) has channel access but should not be able to understand the exchanged messages

=> Alice and Bob can use symmetric encryption to securely exchange messages

But how do they can share a key?

Eve the Eavesdropper

- Eve is an attacker who can see Alice and Bob's messages
- Eve can't modify them
- Eve is a *passive attacker*
- Real-world examples
 - Internet provider
 - Government
 - Anyone nearby if your Wi-Fi is unencrypted
 - Someone else on the same network
 - Lots of potential people...

Ok, so...

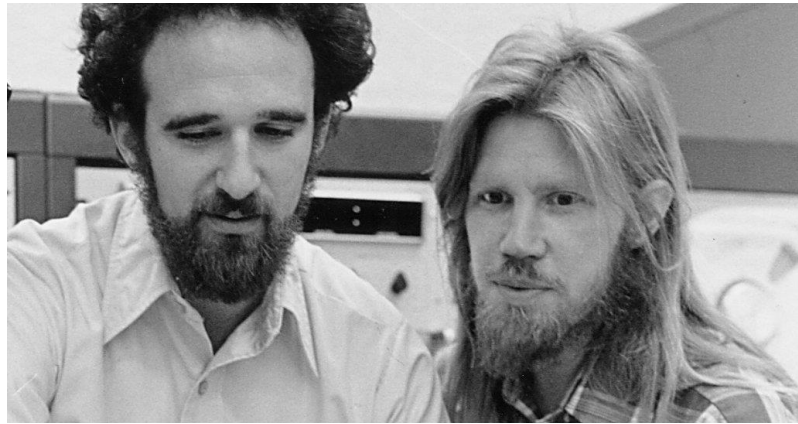
- We can't pick a key and send it => Eve will know it !
- We could pick a key together offline
 - Not feasible in the general case
 - You want to use encrypted communication with a lot of different services on the internet...



Diffie-Hellman Key Exchange

- Invented by Whitfield Diffie and Martin Hellman in 1976
- Allows Alice and Bob to exchange a key without Eve learning it
- Better name for it is Diffie-Hellman key **agreement** protocol
- **Revolutionary Idea:** no need for any prior secret agreement in order to communicate securely
 - Any two users start open negotiations resulting with a shared secret !

1976



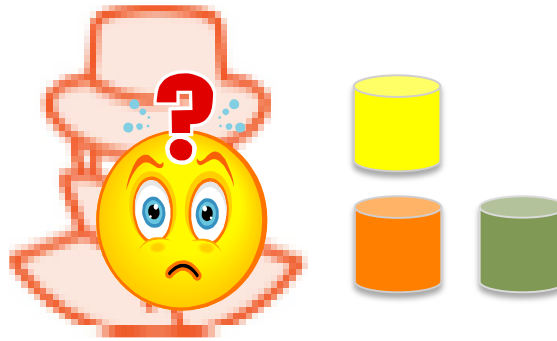
2015



Encryption wizards **Whitfield Diffie** and **Martin Hellman** won \$1m 2015 ACM Turing Award (CS Nobel Prize) for their **Diffie Hellman** Key Exchange protocol

DH in Colors

How can Alice and Bob agree on a **secret color** without Eve finding it out?

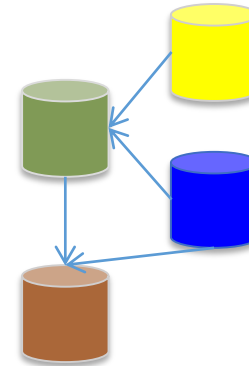
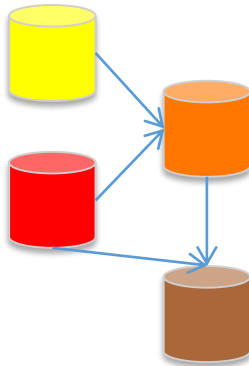


Eve unable to determine the secret color because she doesn't have the right colors to mix together

Eve



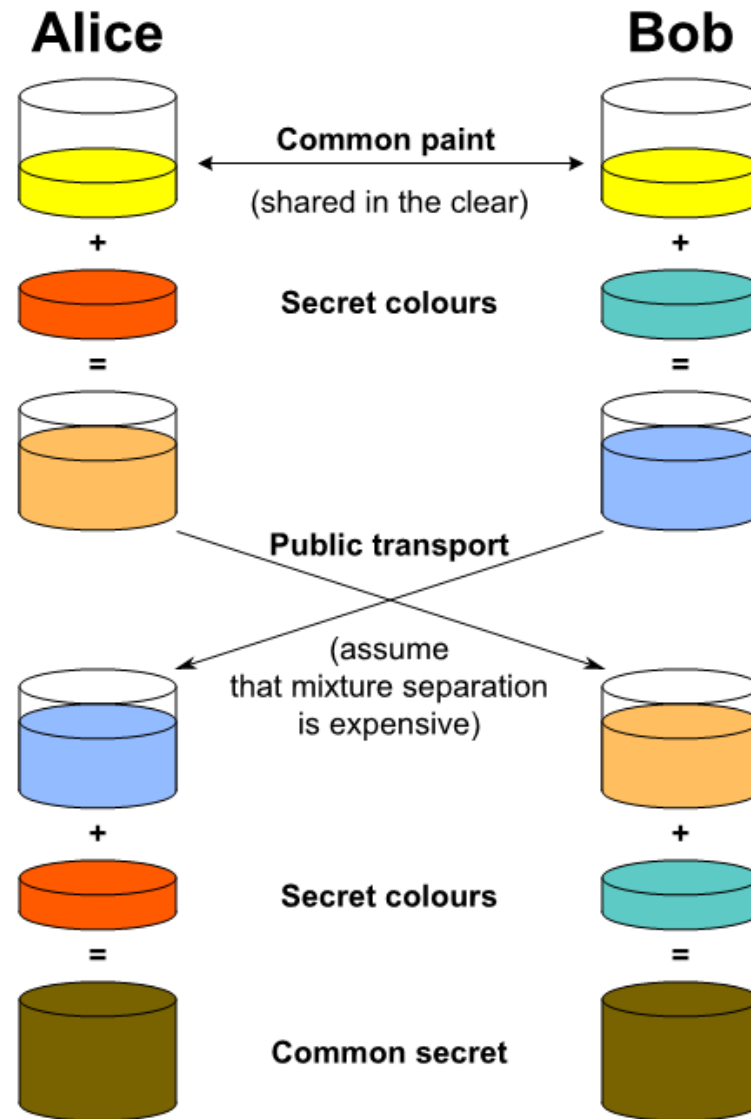
Alice



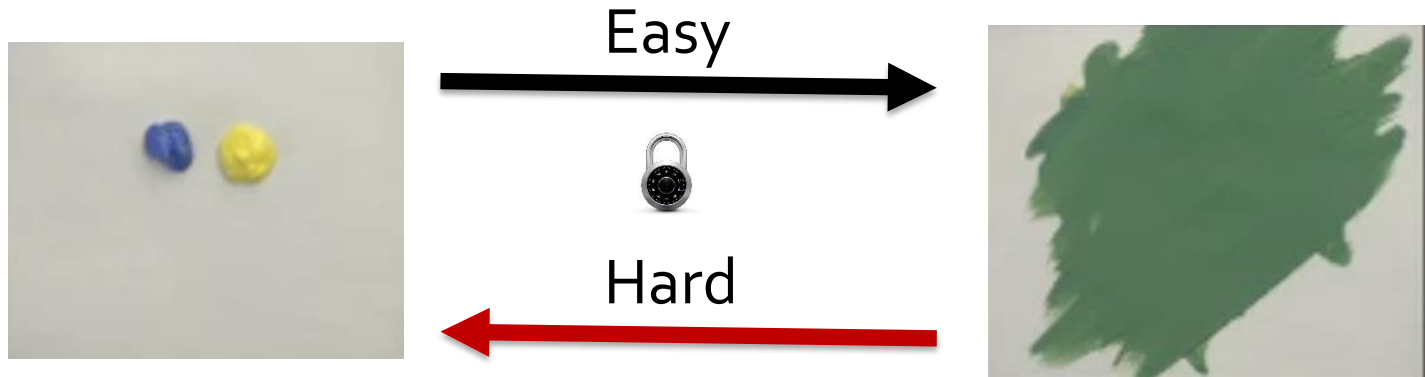
Bob

This is just an analogy, the actual algorithm uses mathematics

DH Colors - Summary



One Way Function



- Easy to mix 2 colors to produce a 3rd one
- Given a mixed color, it is hard to unmix it to get the exact original colors

DH in Math Example

1. Alice and Bob agree on a prime number **p** and a base value **g**. Here, **p=23** and **g=5**

2. Alice chooses a secret number, **a**, and sends Bob **$A = g^a \bmod p$** . Here, **a=6**

$$A = 5^6 \bmod 23 = 15625 \bmod 23 = 8$$

3. Bob chooses a secret number, **b**, and sends Alice **$B = g^b \bmod p$** . Here, **b=15**

$$B = 5^{15} \bmod 23 = 30,517,578,125 \bmod 23 = 19$$

DH in Math Example

4. Alice computes $s = B^a \bmod p$

$$s = 19^6 \bmod 23$$

$$s = 47,045,881 \bmod 23$$

$$s = 2$$

5. Bob computes $s = A^b \bmod p$

$$s = 8^{15} \bmod 23$$

$$s = 35,184,372,088,832 \bmod 23$$

$$s = 2$$

=> Alice and Bob now share a secret, $s=2$, that can't be derived from the public information

Diffie-Hellman key exchange protocol

1. One-time Setup



Alice

$(g = 5, p = 23)$



Bob

2. Random a and b

$a = 6$



$$A = 5^6 \bmod 23$$

$$A = 8$$



$b = 15$



$$B = 5^{15} \bmod 23$$

$$B = 19$$

3. Compute A and B

4. Send A and B

5. Shared key

$$(g^b)^a \bmod p = (g^a)^b \bmod p$$

$$K = 19^6 \bmod 23$$

$$K = 2$$

$$K = 8^{15} \bmod 23$$

$$K = 2$$

DH in Practice

- **a**, **b**, and **p** would need to be MUCH larger in practice
 - 100s of digits long
- This works because Eve can't use A and B to figure out the secret numbers **a** and **b** chosen by Alice and Bob
- DH doesn't prove *who* you share the key with, just that the key isn't known by anyone else

DH in Math

Modulo
exponentiation

$$19^6 \bmod 23 \xrightarrow{\text{Easy}} 2$$



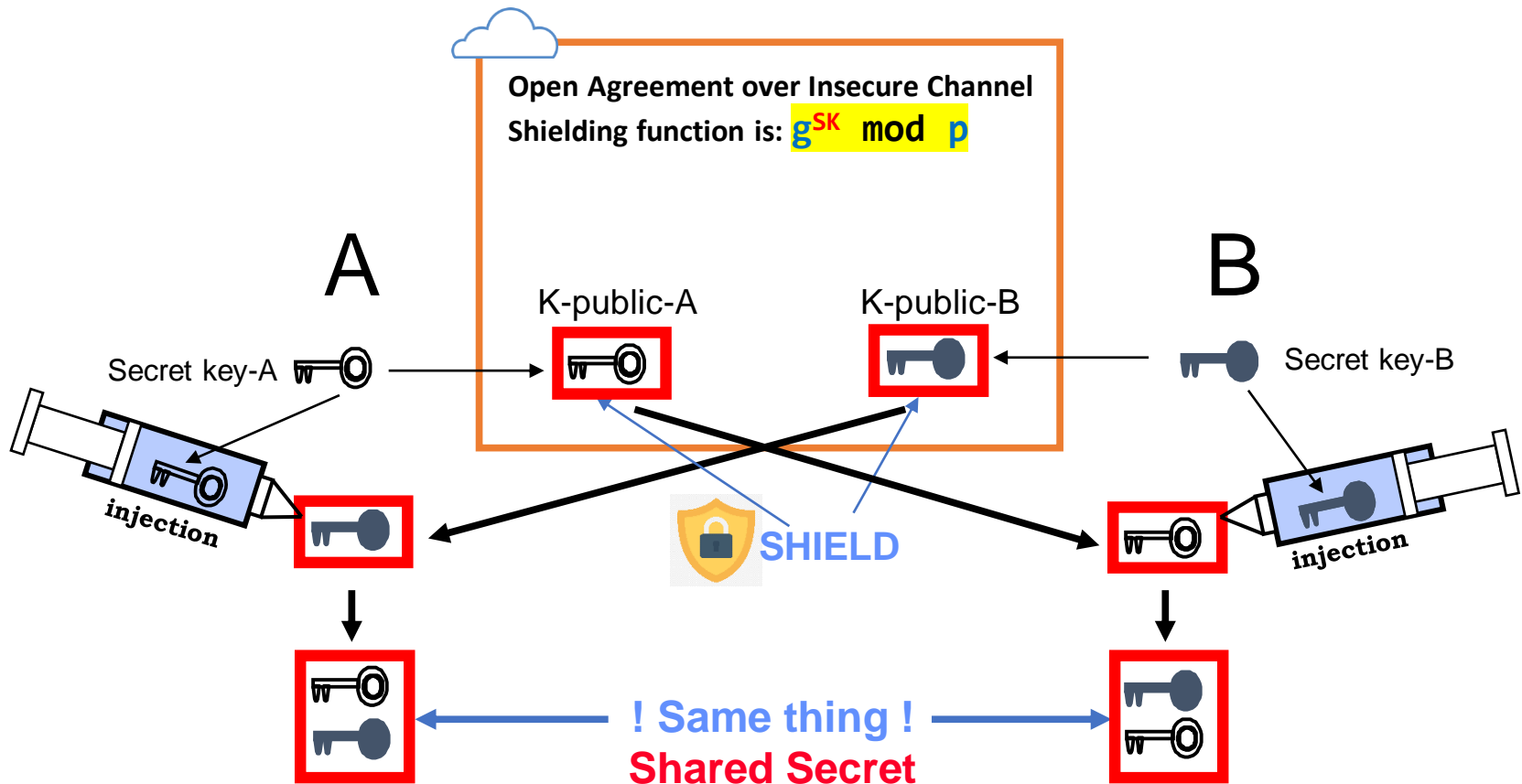
Discrete logarithm
problem

$$19^? \bmod 23 \xleftarrow{\text{Hard}} 2$$

- The mathematics of DH is based on **Modulo exponentiation** + makes use of **big prime numbers** (100s of digits long)
- Security relies on the difficulty of computing **Discrete logarithm problem**. No efficient algorithm is known to solve it.
=> can only be solved through trial and error to find matching exponent but it will take a very long time (thousands of years with the world's computing power to run through all possibilities!)

DH Breakthrough

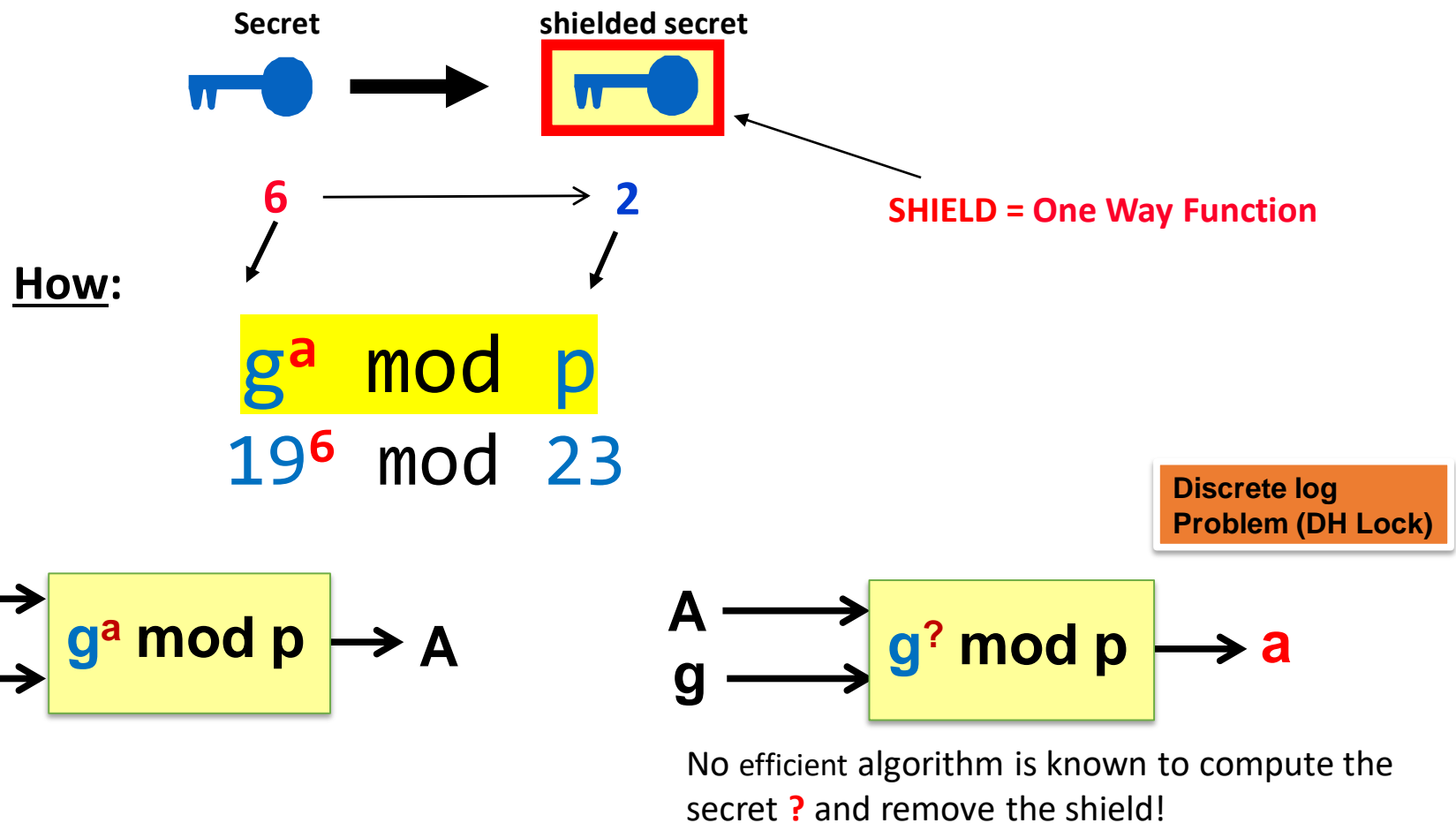
Shared Secret without exchange of secrets



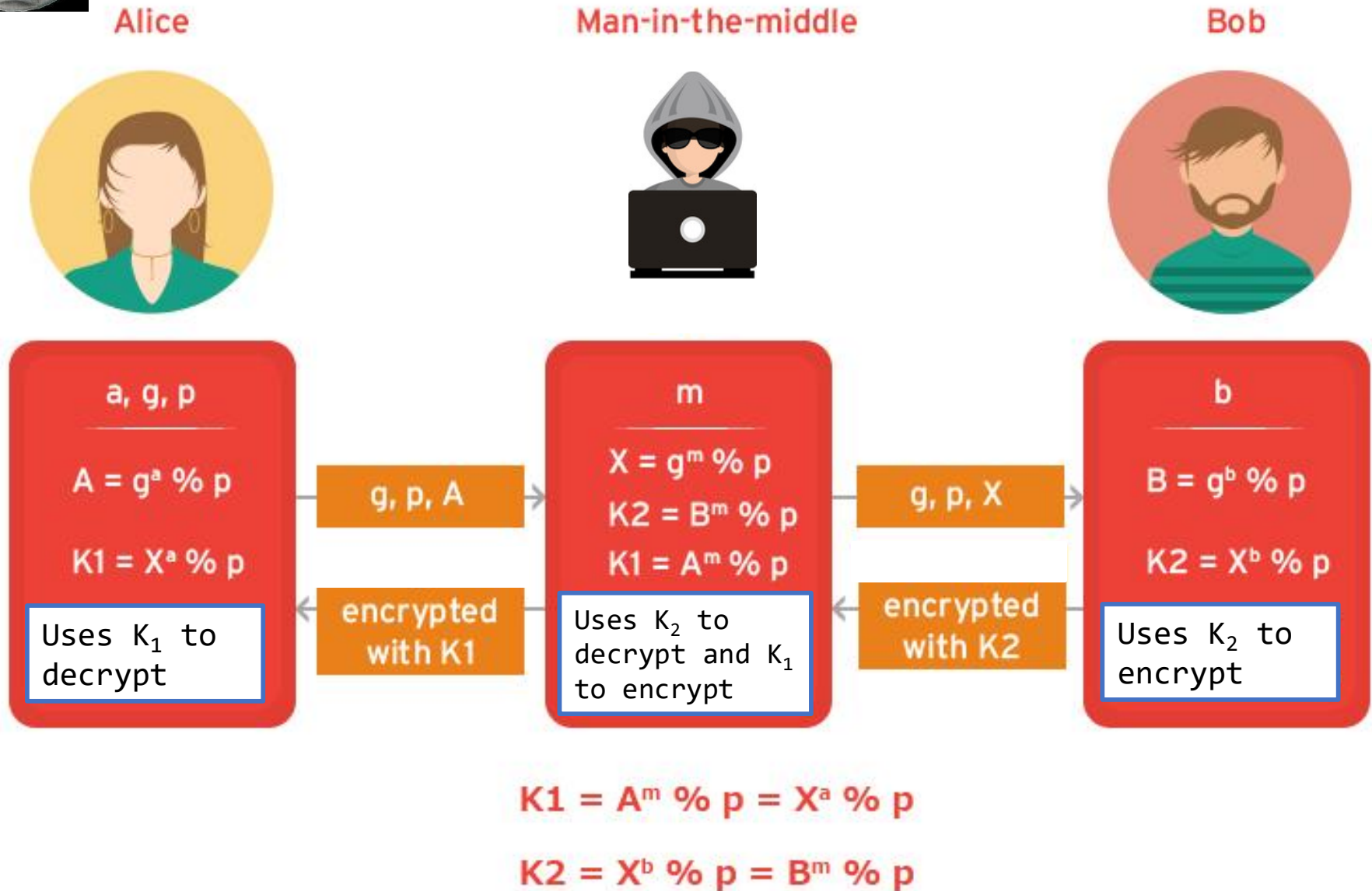
How to “publicly” hide (shield) a secret ?

Key idea: by using the so called *One-way Function*

Diffie-Hellmann used a One-Way function:



Man-in-the-middle (MITM) attack



Man-in-the-middle (MITM) attack

- Mallory (active attacker) intercepts the DH exchanged parameters in both directions
- Mallory and Alice uses DH algorithm to calculate a shared key K_1

$$K_1 = A^m \% p = X^a \% p$$

- Mallory and Bob uses DH algorithm to calculate a shared key K_2

$$K_2 = B^m \% p = X^b \% p$$

- Now Alice and Bob correspond through Mallory who can read all their messages
 - Mallory will use K_2 to decrypt messages from Bob then Encrypt them with K_1 before forwarding them to Alice

Summary

- Symmetric Key crypto has a major problem: How do two people who don't know each other share a key?
- A Diffie-Hellman key exchange lets them compute a shared key even in the presence of an eavesdropper, Eve.
- DH is vulnerable to Man-in-the-middle (MITM) attack

Resources

- DH original paper

<https://ee.stanford.edu/~hellman/publications/24.pdf>

- DH Wikipedia page

[http://en.wikipedia.org/wiki/Diffie-Hellman key exchange](http://en.wikipedia.org/wiki/Diffie-Hellman_key_exchange)

- Play with color mixing

<https://trycolors.com/>