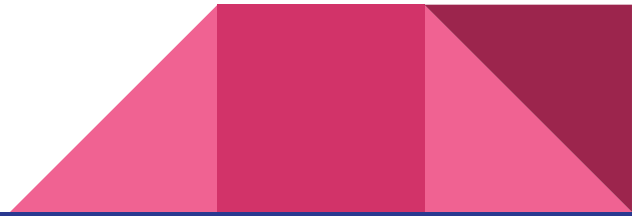
Cryptography Summary

Perfect Secrecy

- Key must be as long as the message
- Key must be random
- All possible messages have the same chance of being the plaintext for a given ciphertext.



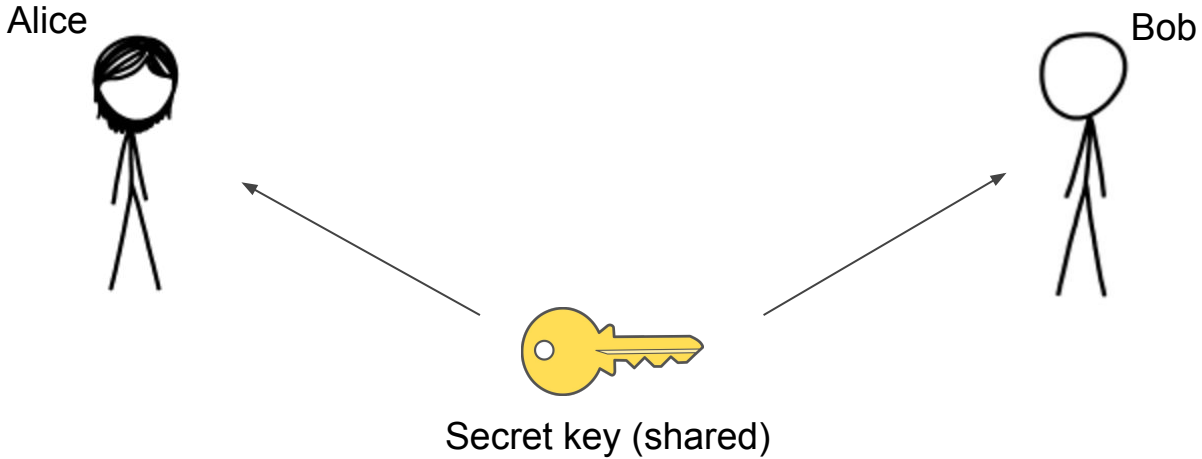
Semantic Secrecy

- Cannot determine any information on Ciphertext given realistic computer power
- Pseudorandom Generator
  - Input a smaller random number, give you a larger pseudorandom random number
- Pseudorandom Function
  - Input a plaintext and key, give you a pseudo random ciphertext

Symmetric Key Cryptography

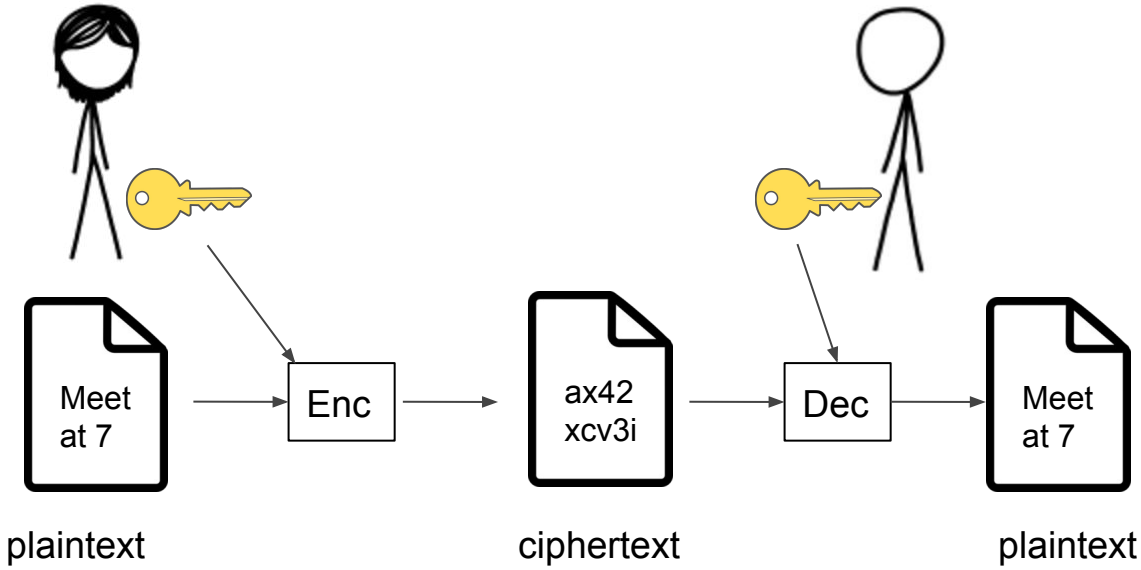
Secret communication using a **shared** secret key.

Anyone with the key can both **encrypt** and **decrypt** messages.



Symmetric Key Cryptography

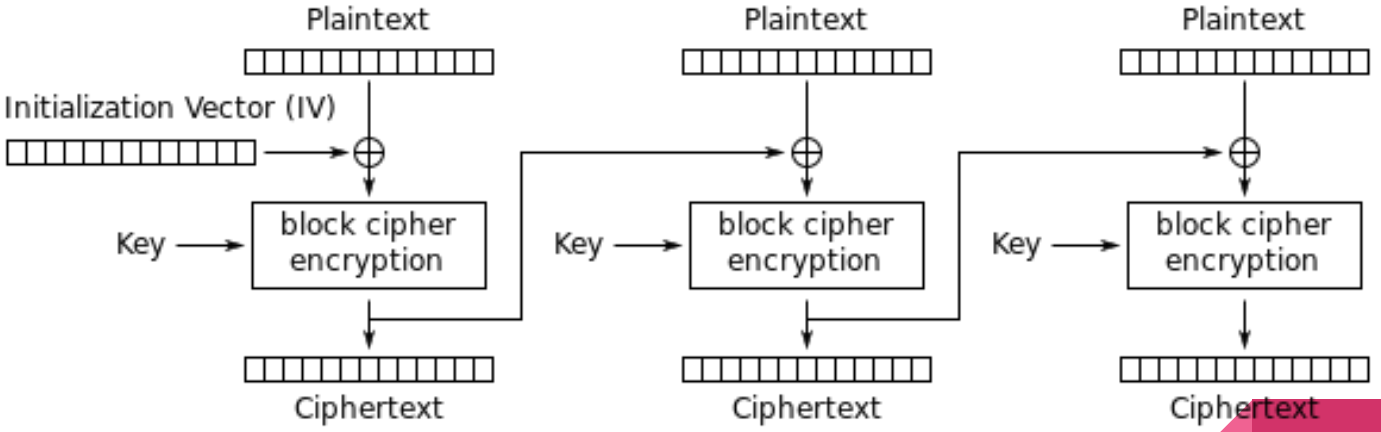
Secret communication using a **shared** secret key.



Note: symmetric-key schemes have a shared key, but Enc and Dec may be completely different operations!

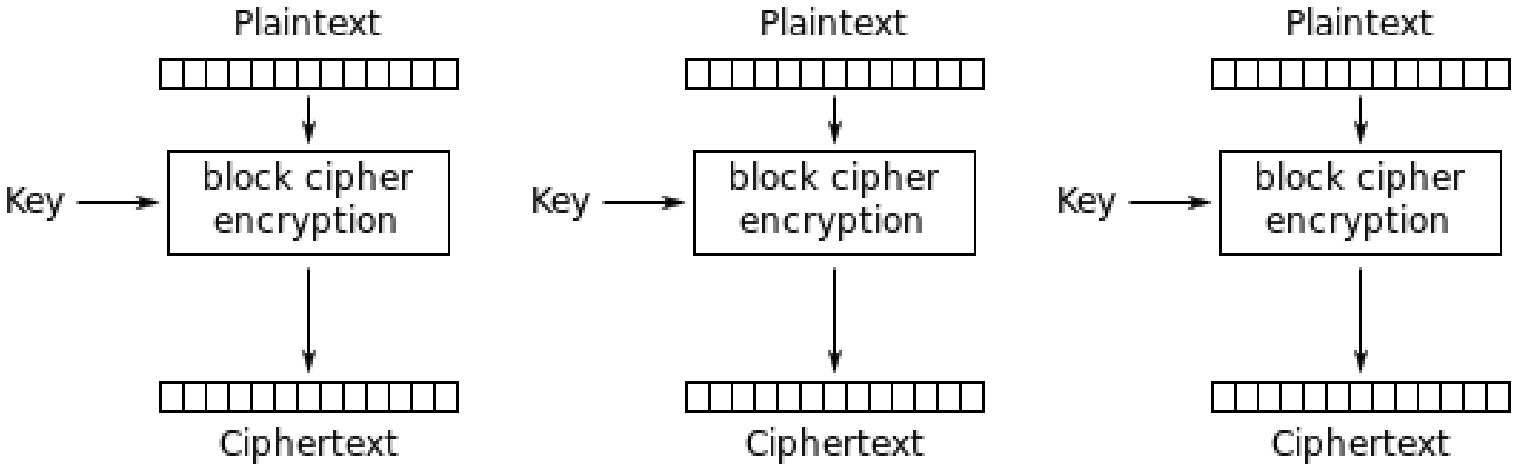
Advanced Encryption Standard (AES)

These days, symmetric-key crypto is **fast**, **efficient**, and **secure**.



Cipher Block Chaining (CBC) mode encryption

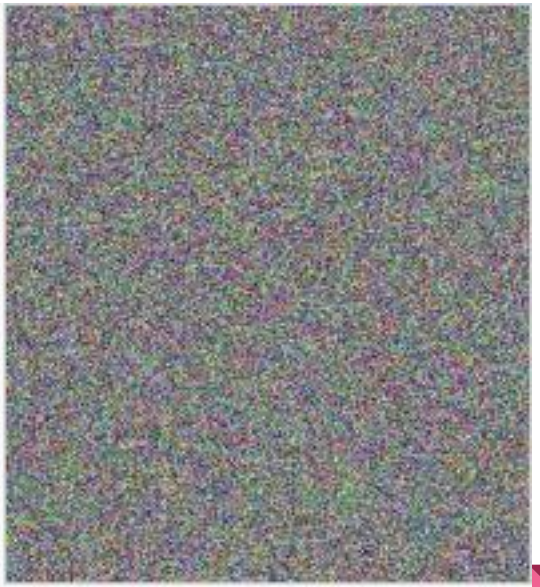
AES ECB mode



Electronic Codebook (ECB) mode encryption

Original image

Encrypted using ECB mode

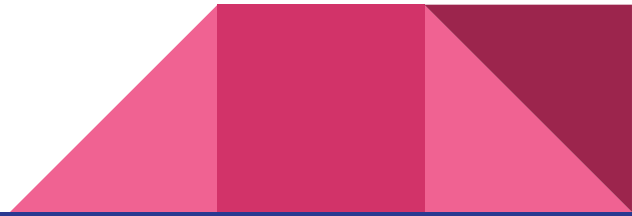


Modes other than ECB result in pseudo-randomness

Conclusion

Plaintext + Key + Encryption =

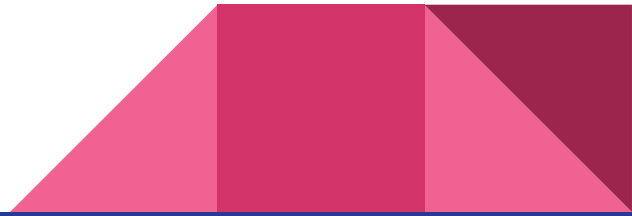
Ciphertext + Key + Decryption =



Conclusion

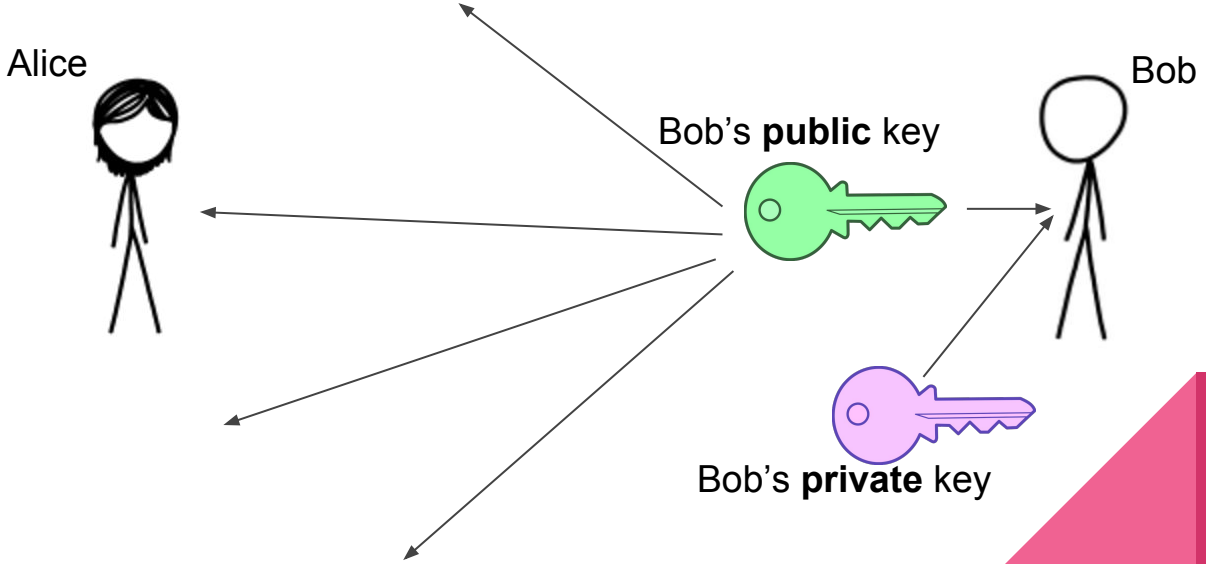
Plaintext + Key + Encryption = Ciphertext

Ciphertext + Key + Decryption = Plaintext



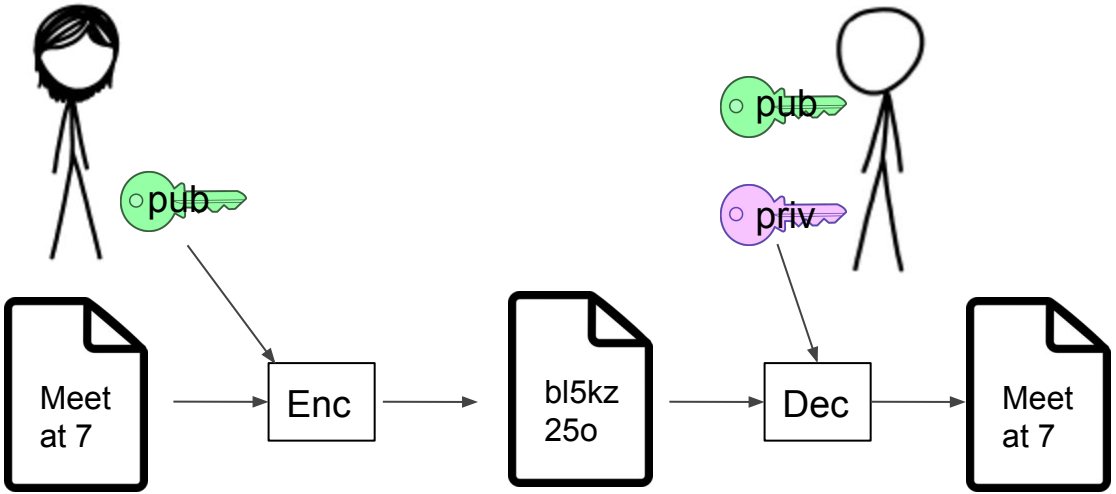
Public Key Cryptography

Secret communication using **public** and **private** keys. **Encrypting** (to a specific person) can be done by anyone, but only the owner of the private key can **decrypt**.



Public Key Cryptography

Secret communication using **public** and **private** keys. **Encrypting** (to a specific person) can be done by anyone, but only the owner of the private key can **decrypt**.



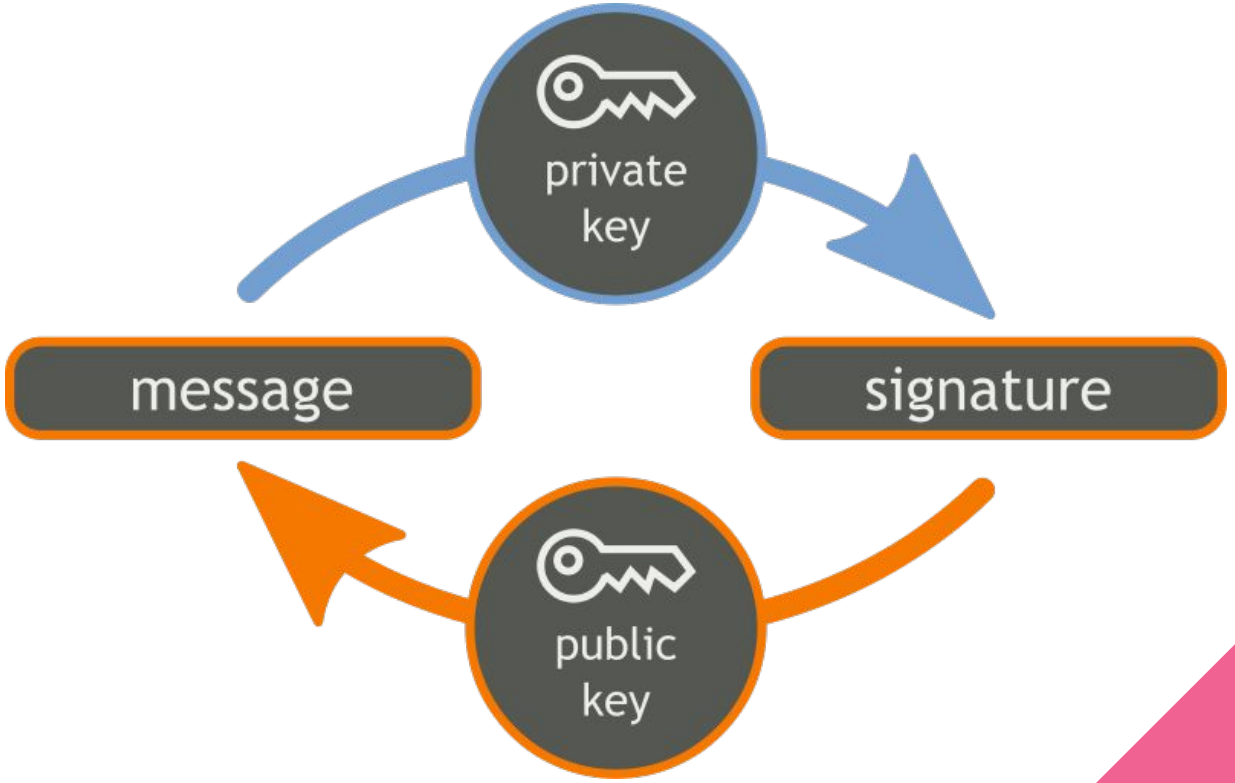
Lock analogy: If Bob has 1 key and 100 locks for the key, he can give the locks to everyone. People can put things in a box, lock it with his lock, and then only he can open it.

Public Key Cryptography

https://www.youtube.com/watch?v=dleUxfghd5I

How do I know a message is legitimate?

How do I know a message is legitimate?



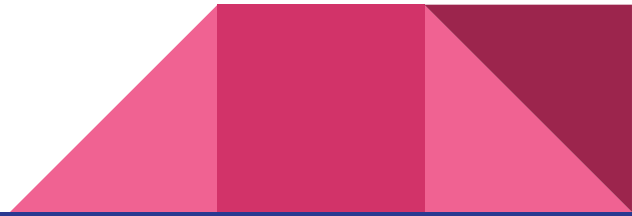
Conclusion

Public Key + Plaintext + Encryption =

Private Key + Ciphertext + Decryption =

Private Key + Message =

Public Key + Message + Signature =



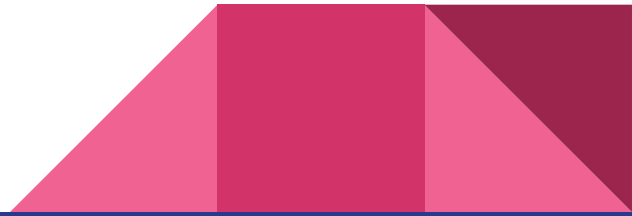
Conclusion

Public Key + Plaintext + Encryption = Ciphertext

Private Key + Ciphertext + Decryption = Plaintext

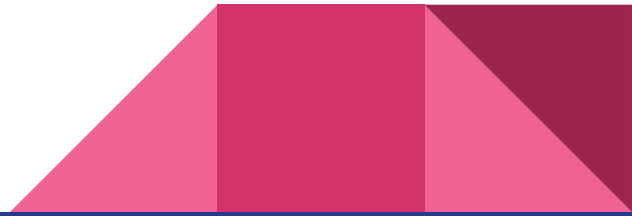
Private Key + Message = Signature

Public Key + Message + Signature = Yes/No

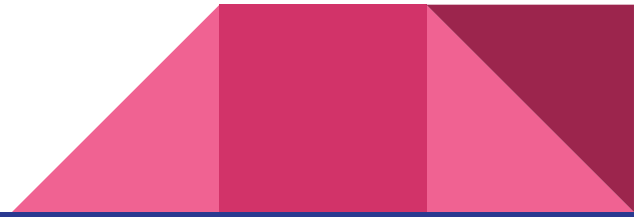


Why not just always use public-key crypto?

Mostly, because it’s much slower.

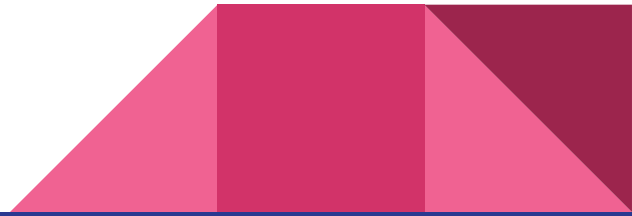


How to share keys for symmetric cryptography

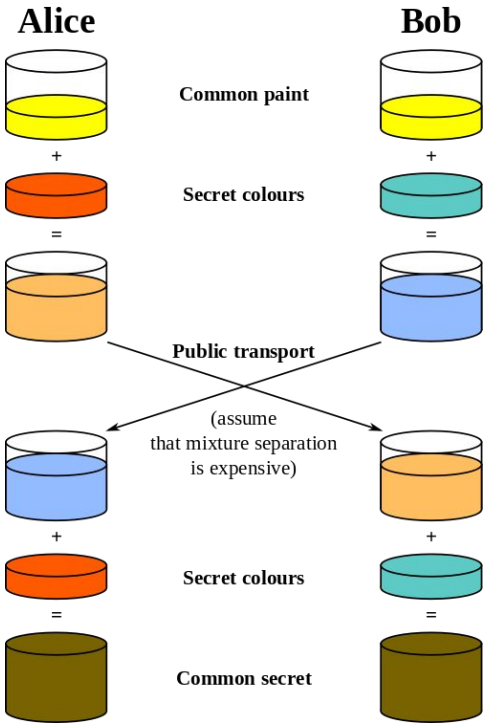


Diffie-Hellman Key Exchange

https://www.youtube.com/watch?v=3QnD2c4Xovk

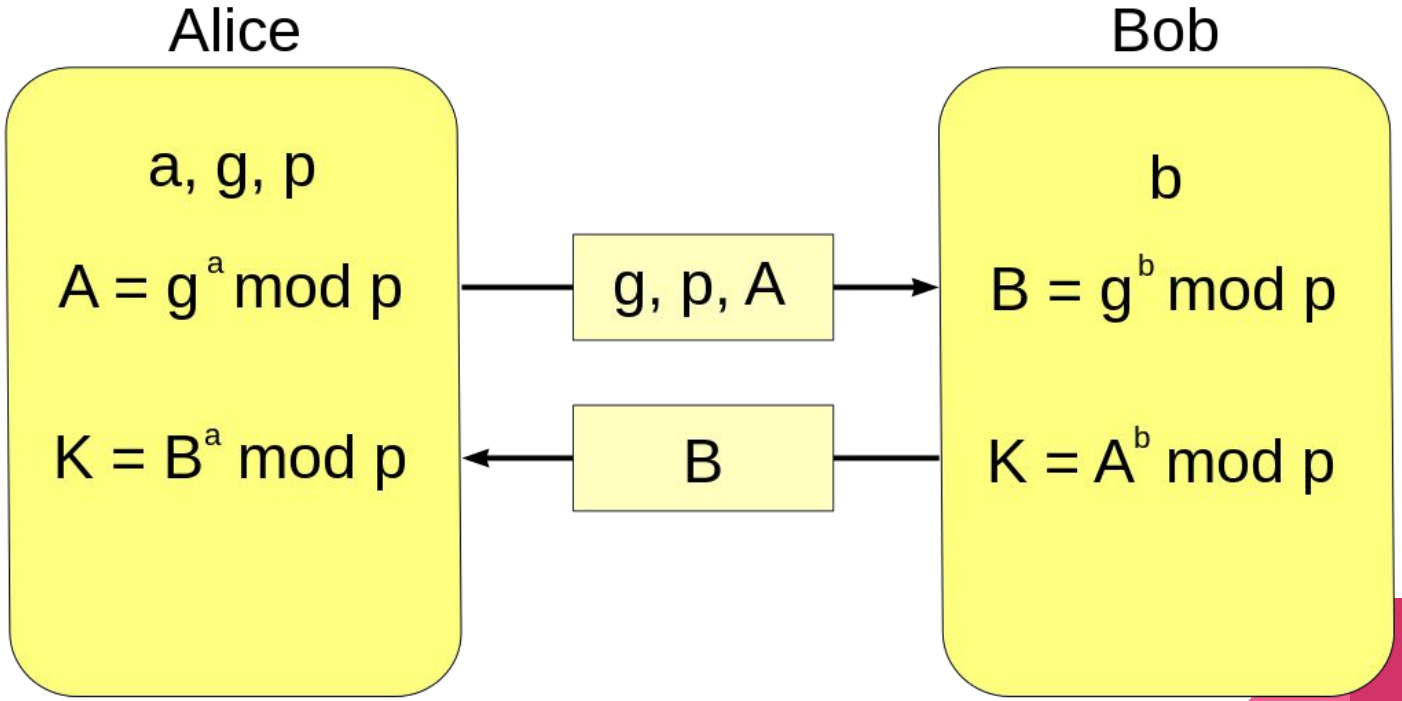


Diffie-Hellman Key Exchange



Use public key crypto to share a secret key!

Activity Diffie-Hellman key exchange



$K = A^b \bmod p = (g^a \bmod p)^b \bmod p = g^{ab} \bmod p = (g^b \bmod p)^a \bmod p = B^a \bmod p$