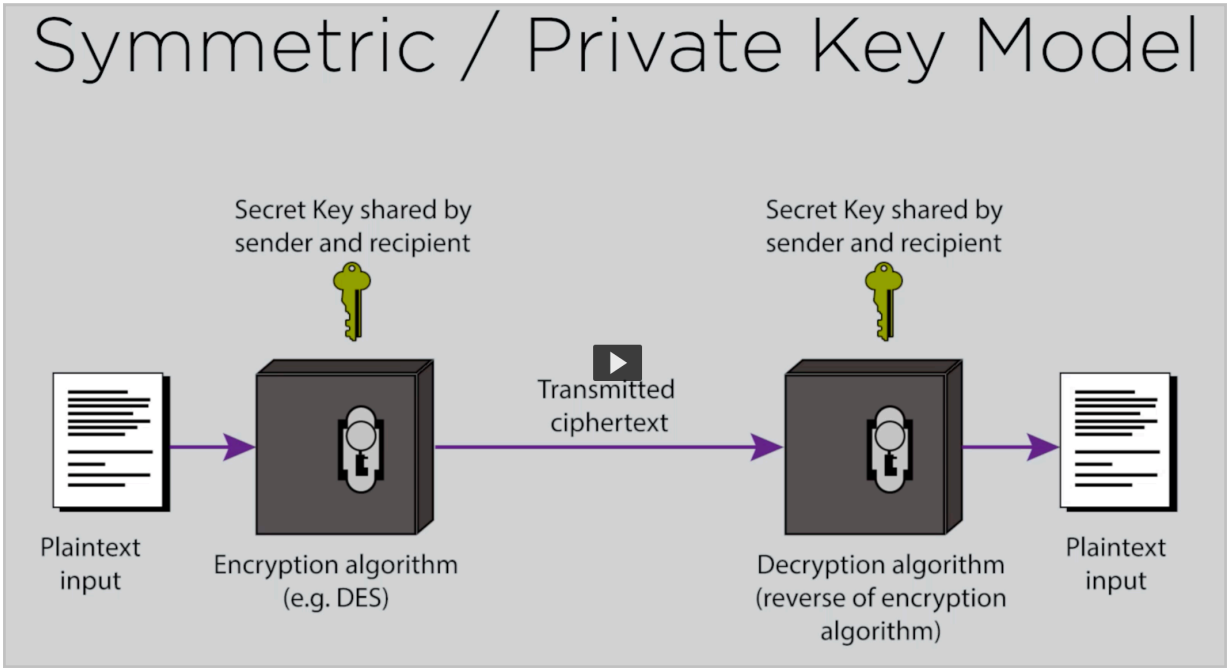
Intro To Cryptography

Symmetric Key Cryptography - the sender and the receiver share a common secret key that is known to only them.
Both Encryption and Decryption are performed using this single key. Also known as private key cryptography.



EXAMPLE OF PRIVATE KEY CRYPTOGRAPHY:

- Caesarean Cipher - taking letters of Alphabet and shifting the characters (N) letters down (D -> A, A -> X ... 3 char shift) to encrypt. The one who decrypts shall upshift 3 characters to see the plaintext. Easy to decrypt by intercepting parties.

- One Time Pad - not widely used in practice but very secure. Instead of shifting all characters the same N, we let each character have it's own shift number thus greatly increasing the variation and making brute force too cumbersome. Transmitter sends list of random shifts as long as the

message. Assuming interceptor of this message does not have key, they cannot use some predetermined pattern to decrypt. Every character has an equal likelihood of being shifted by any N between 1 -> 26 so algorithm is protected by randomness

NOTE:
Encryption is as only as strong as the channel by which the secret keys are shared. If it's send over an insecure channel, the encryption is useless.

3 Functions for Symmetric Encryption:

A Key Generation algorithm KeyGen(L) -> K where L is a security parameter that represents the length of the key.
An Encryption algorithm Enc(K,M) -> C that takes a key K and a message M and returns a ciphertext C.
A Decryption algorithm Dec(K,C) -> M that takes the same key K and a ciphertext C and returns a message M.

ALGOS FOR PRIVATE KEY ENCRYPTION:

Data Encryption Standard (DES) - too insecure for applications due to short key length which can effortlessly be trumped by a machine (back in 1999, it took 22 hrs to due so)

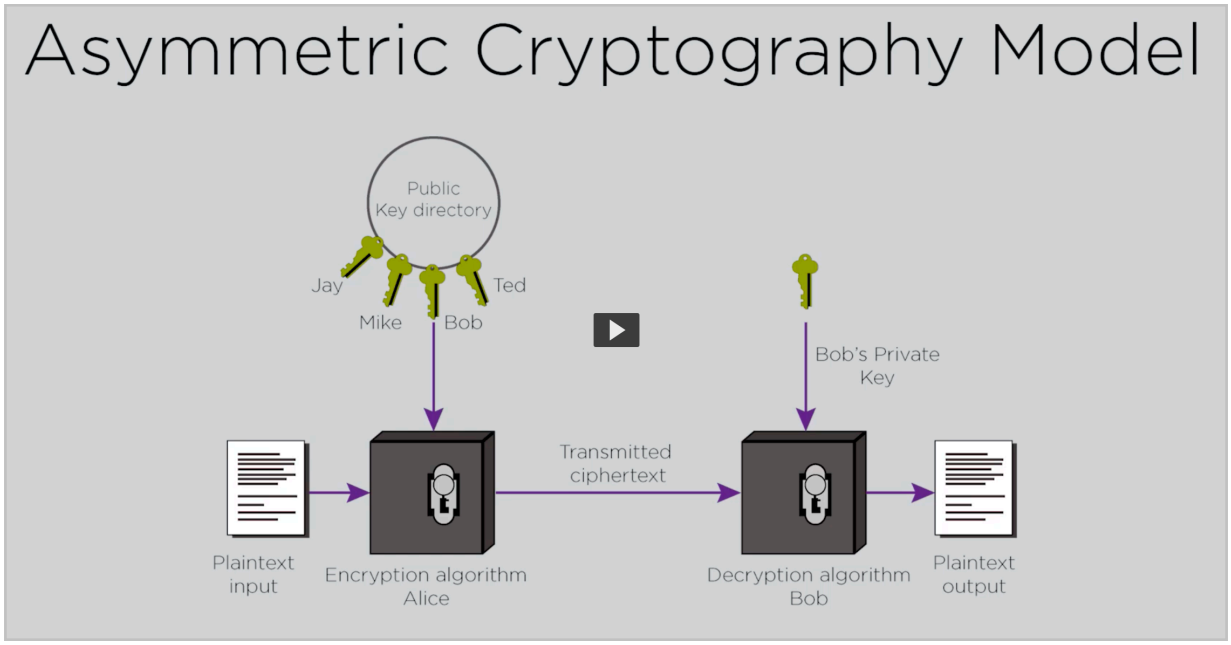
Advanced Encryption Standard (AES) - recommended algorithm for the current time due to it's key size of 128, 192, or 256 bits which makes it too tiresome to crack (it would take over 1 billion years to break 128 bit AES key with a supercomputer as computer will have to perform exhaustive permutation trials to get a valid break)

Triple-DES - forms a 168-bit long key makes it difficult to brute force against traditional DES. It basically worked by encrypting the cipher text 3 times with 3 unique keys then decrypting them with same keys in reverse order. AES should still be preferred as it may be excessive to transmit 3 unique keys.

Ciphertext = Enc(K3, Enc(K2, Enc(K1, plaintext)))
plaintext = Dec(K1, Dec(K2, Dec(K3, ciphertext)))

Asymmetric Cryptography - (also known as public key crytography) - this form of cryptography alleviates the need to share private keys on a secure channel before communication. The receiver and transmitter has 2 keys (public and private). The public key is meant to be released to anyone who wants to send messages to the user. Encryption is done using the public key and decryption is accomplished using private key.

https://www.giac.org/paper/gsec/2171/idiots-guide-public-key-infrastructure/103692



Function Definitions:

Generate Keys (This generation is done through a PKI which performs key management) :
Keygen(L) -> K_{pub}, K_{priv}

Encryption:
Enc(K_{pub}, Message) -> Ciphertext

Decryption:
Dec(K_{priv}, Ciphertext) -> Message

Remember, it is assumed that it is extremely difficult to compute K_{priv} given K_{pub}!

Hash Functions and Digital Signatures:
Imagine that we have a large amount of data and we want to efficiently calculate a non-reversible fixed-length value that represents that data. However, as an added challenge, we do not want that value to give away any hint about the data it represents. We want this value to be fast to compute, but infeasible to reverse, among other properties. This is the idea of a cryptographic hash.

Hash Properties:

- Can be applied to data of any length.
- Output is fixed length.
- Relatively easy to compute h(x), given x.
- Is deterministic, meaning given the same x we get the same h(x).
- Infeasible to get x, given h(x). This is called the one-wayness property.
- Given x, infeasible to find y such that h(x) = h(y). This is called the weak-collision resistance property.
- Infeasible to find any pair x and y such that h(x) = h(y). This is called the strong-collision resistance property.