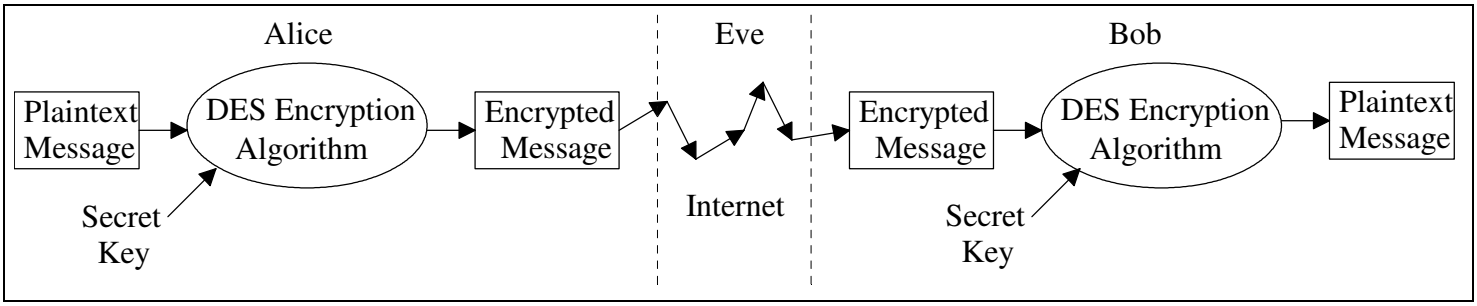
Encryption Summary

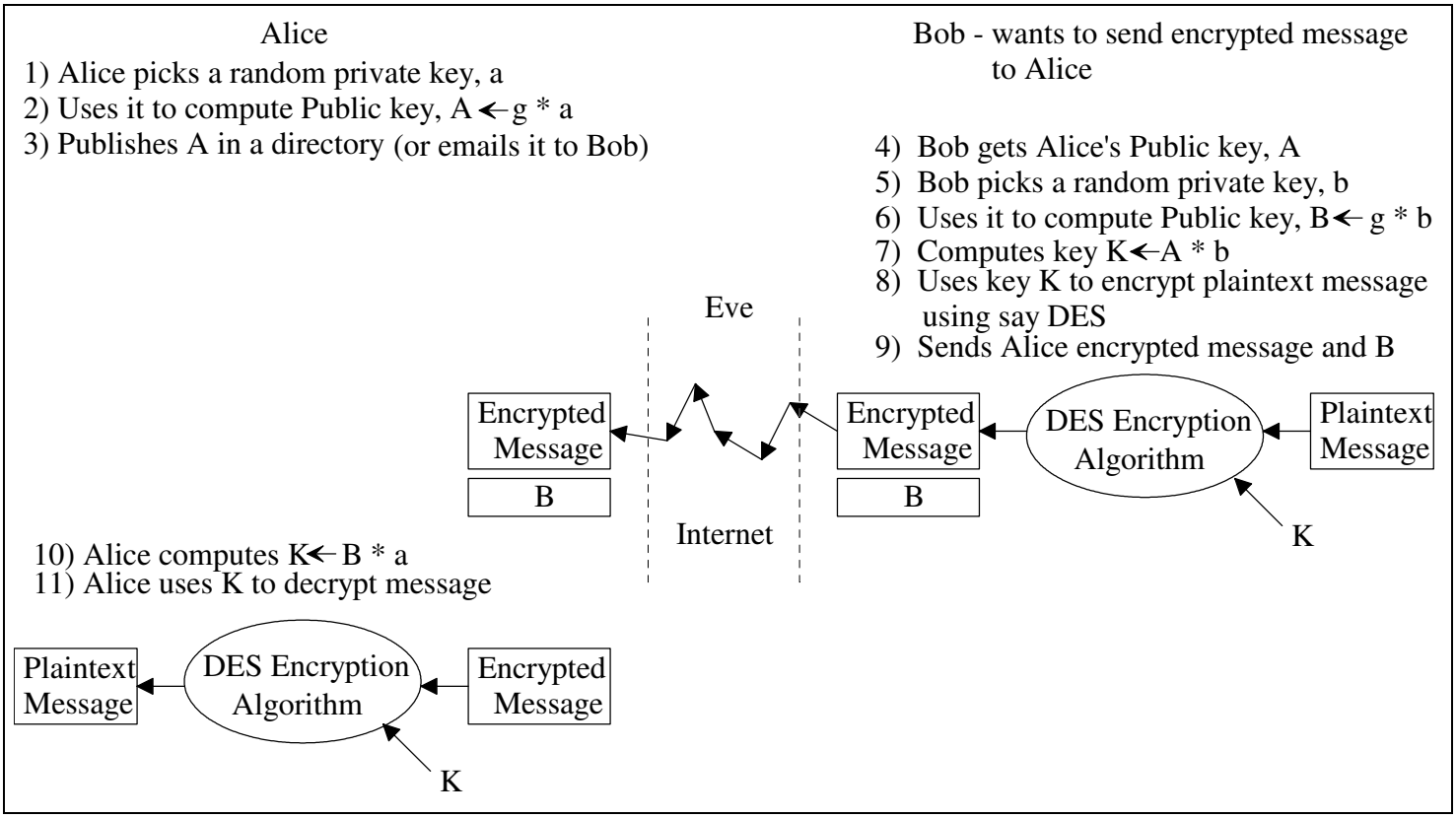
Two main types of modern encryption approaches:

1. Secret-key - same secret key used to encrypt and decrypt the message
   - Examples: Data Encryption Standard (DES) and Advanced Encryption Standard (AES)
   - Not proven to be unbreakable, but best known method is to brute force try all possible keys takes exponential amount of time
   - Both Alice and Bob must know the same secret key



2. Public-key - everybody has their own private key that only they know and everybody has their own public key that they share freely (in a directory or send with the encrypted message).
   - Examples: Diffie-Hellman-Merkle and RSA (Rivest-Shamir-Adleman) with RSA public and private keys are inverses of each other
   - Neither proven to be unbreakable, but best known method is to brute force try all possible keys takes exponential amount of time
   - RSA is typically used to distribute a secret key between sender and receiver, so the less computationally intensive ("faster") DES (or AES) algorithm can be used to encrypt/decrypt the plaintext message

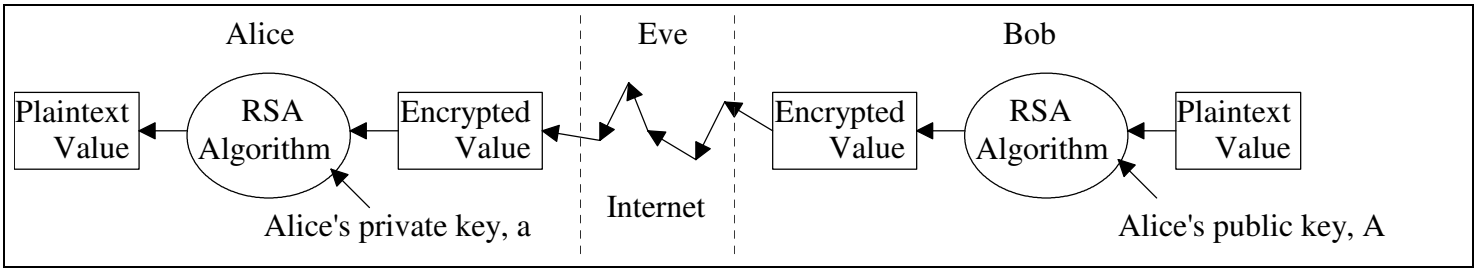
Example using Diffie-Hellman-Merkle algorithm: Bob want to send Alice an encrypted message



Notes:

- K is only used once (like a one-time pad) so it is harder to break.
- Bob computed K ← A * b = (g * a) * b, and Alice computed K ← B * a = (g * b) * a. These K's are equal by the one-way computation properties of the Diffie-Hellman-Merkle algorithm

Recall that in the RSA algorithm each Public - Private key pair (e. g. Alice's A and a) are inverses of each other. For example, Bob can use Alice's public key, A, for encryption and Alice can use her private key, a, for decryption



Typically, the "Plaintext Value" being sent via RSA is the secret key used to encrypt/decrypt the message via a "faster" secret-key algorithm like DES or AES.

For example, assume Bob wants to send Alice an encrypted message: (Both publish their public keys, but each keep their private keys to themselves)