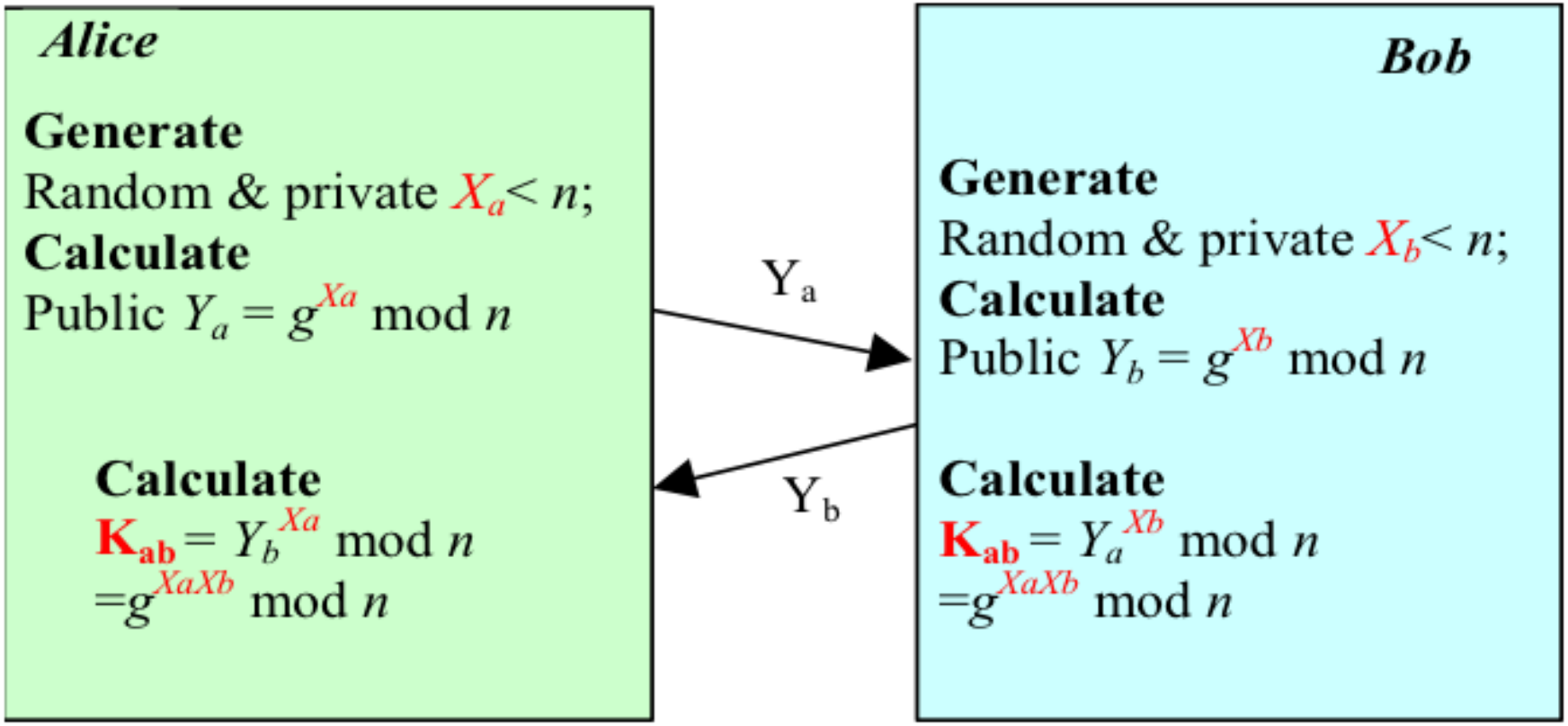
## Diffie-Hellman Algorithm/Protocol

- DH was the 1st public-key algorithm ever invented - back in 1976.
- DH key exchange protocol allows two parties who have never met before to exchange messages in public and collectively generate a key that is private to them, and none of the parties could predetermine the key.
- Its security is based on the difficulty of calculating discrete logarithms in a finite field.
  - Given integers $y$ and $g$ and prime number $n$, compute $x$ such that $y = g^x \bmod n$.
  - This is computationally infeasible if $n$ is sufficiently large.

## Diffie-Hellman Algorithm/Protocol

- Assuming two parties, *Alice* and *Bob,* take part in the exchange.
- Initial condition
  - *Alice* and *Bob* agree on two large integers, ***g*** and ***n***;
  - $n$ - prime number that serves as the modulus.
  - $g$ - random number that serves as the basis, with $1<g<n$.
  - $g$ and $n$ do not have to be secret.
- Definition
  - *Alice* has private key $X_a$, and public key $Y_a$.
  - *Bob* has private key $X_b$, and public key $Y_b$.

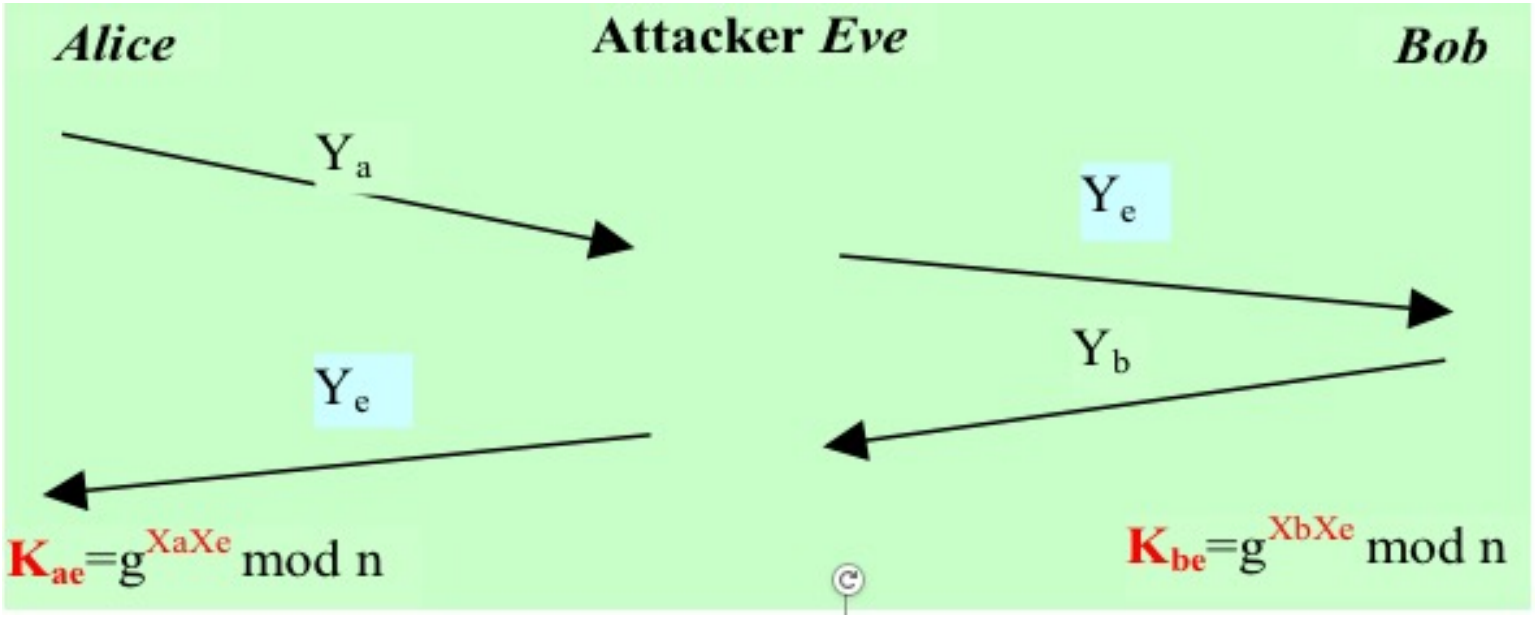
# Diffie-Hellman Algorithm/Protocol



## Diffie-Hellman Protocol

- It resists passive attacks such as eavesdropper, as calculating a discrete logarithm is a computationally hard problem.
- There is one problem - neither party knows who it shares the secret with! So it is vulnerable to active, man-in-the-middle attacks, as to be illustrated shortly.

# Diffie-Hellman Protocol - Man-in-the-middle attack




- *Alice* (*Bob*) thought she shares a key with *Bob* (*Alice*), but actually with *Eve*.
- So the attacker *Eve* can intercept and read any messages encrypted without been detected by *Alice* and *Bob* .