Asymmetric encryption for **integrity**

Alice encrypts a message $\mathbf{m}$ with her private key $\mathbf{Ks_A}$

➡ Everybody can decrypt $\mathbf{m}$ using Alice's public key $\mathbf{Kp_A}$

✓ Authentication with non-repudiation (a.k.a Digital Signature)

K_{SA}, K_{pA}

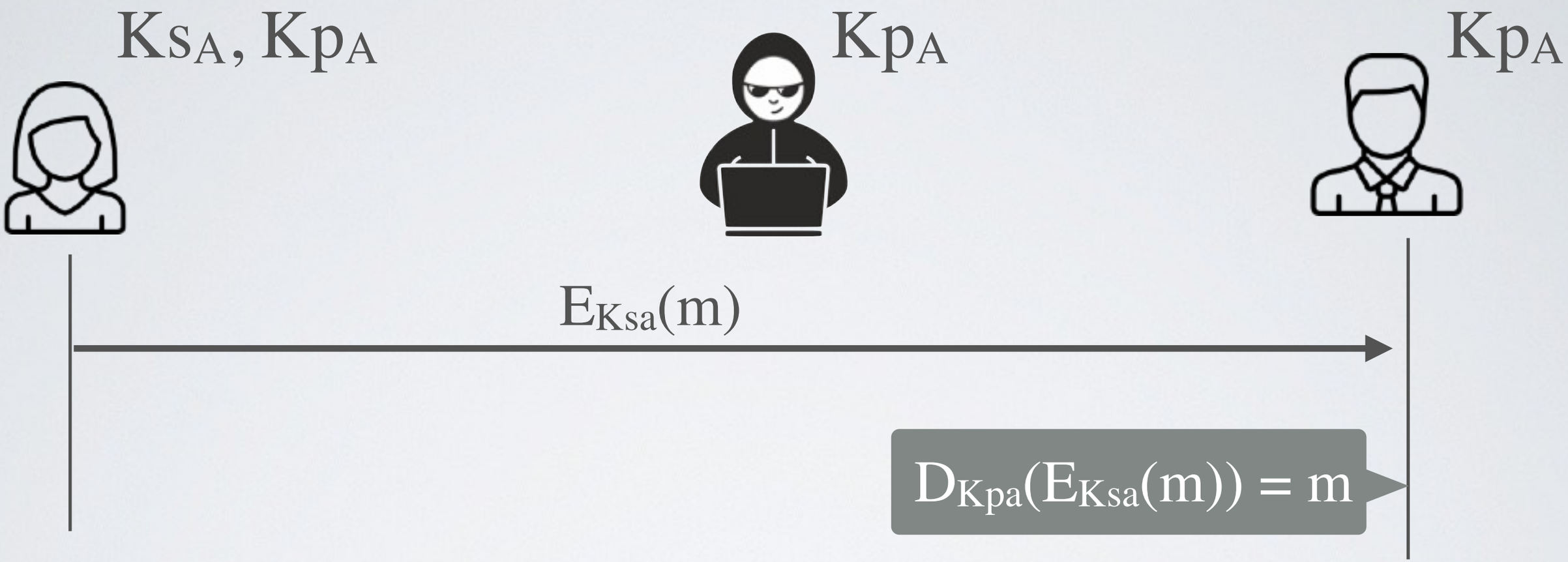

Кра


Кра


$E_{Ksa}(m)$

$$D_{Kpa}(E_{Ksa}(m)) = m$$

Asymmetric encryption for **integrity**



Alice encrypts a message **m** with her private key $\mathbf{Ks_A}$

➡ Everybody can decrypt **m** using Alice's public key $\mathbf{Kp_A}$

✓ Authentication with non-repudiation (a.k.a Digital Signature)

Functional Requirements

$D_{Ks}(E_{Kp}(m)) = m$ and $D_{Kp}(E_{Ks}(m)) = m$ for every pair (Kp, Ks)

- ✓ Generating a pair (Kp, Ks) is easy to compute (polynomial)
- ✓ Encryption is easy to compute (either polynomial or linear)
- ✓ Decryption is easy to compute (either polynomial or linear)
- ◉ Finding a matching key Ks for a given Kp is hard (exponential)
- ◉ Decryption without knowing the corresponding key is hard (exponential)