Exercise

• Open “EncryptionTrivia.docx”
• Go through each scenario and add
  • C for “Confidentiality”
  • I for “Integrity”
  • NR for “non-repudiation”
Confidentiality, integrity, non-repudiation?

- **M + H(M)**
  - No confidentiality bc M is in clear
  - Maybe integrity – if we send M from A to B, we cannot send H(M) bc Mallory can replace M and recalculate H(M), for integrity we have to store H(M) somewhere so B can get it through a separate channel or we have to encrypt it or generate a MIC
  - No non-repudiation bc we don’t have signatures in this scenario

- **M + E(H(M))**
  - No conf, bc M is in clear
  - Integrity and NR depend on how key is shared:
    - **M + E_{sharedK}(H(M))**
      - yes integrity bc only A and B know the sharedK
      - no NR bc A and B both have sharedK so they both could have sent the message
    - **M + E_{privA}(H(M))**
      - yes integrity bc only A knows privA
      - yes NR bc A only A knows privA
    - **M + E_{pubB}(H(M))**
      - no integrity bc pubB and H are both public
      - no NR bc pubB and H are both public
Confidentiality, integrity, non-repudiation?

- \( M + H(E(M)) \)
  - No conf bc \( M \) is in clear
  - For integrity and NR, let’s look at different versions of \( E \):
    - \( M + H(E_{\text{shared}}(M)) \)
      - yes integrity, but this is more expensive bc encrypting a big \( M \) is more expensive than encrypting a small hash
      - no NR
    - \( M + H(E_{\text{priv}}(M)) \)
      - no integrity
      - no NR, no one can check \( H(E_{\text{priv}}(M)) \)
    - \( M + H(E_{\text{pub}}(M)) \)
      - no integrity bc \( \text{pubB} \) and \( H \) are both public
      - no NR
Confidentiality, integrity, non-repudiation?

- $E(M) + H(M)$
  - $E_{sharedK}(M) + H(M)$
    - yes conf. bc only B and A can read the message
    - yes integrity bc Mallory doesn’t have sharedK and cannot generate $H(M)$ to fit $M$ and she cannot retrieve $M$ from encrypted content
    - no NR bc sharedK is used to encrypt
  - $E_{privA}(M) + H(M)$
    - no conf bc everyone can read w pubA
    - yes integrity bc Mallory doesn’t have privA and she cannot change $M$ to $M_1$ without $B$ noticing that two pieces don’t fit together
    - yes NR bc we are encrypting w privA
  - $E_{pubB}(M) + H(M)$
    - yes conf bc only B has privB and can decrypt the message
    - no integrity bc everyone knows pub and $H$
    - no NR
  - $E_{sharedK}(M)$
    - we could have integrity here even without $H(M)$ only if $M$ has a known format, otherwise if $M$ is random we need $H(M)$
    - Yes conf
    - No NR
  - $E_{privA}(M)$
    - No conf bc everyone can read w pubA
    - we could have integrity here even without $H(M)$ only if $M$ has a known format, otherwise if $M$ is random we need $H(M)$, same goes for NR
When/How to Encrypt/Hash?

- Confidentiality, integrity, non-repudiation
  - $E(M) + H(E(M))$
  - $E_{\text{shared}}(M) + H(E_{\text{shared}}(M))$
    - yes conf, yes integrity but the second encryption is wasteful, no NR
  - $E_{\text{shared}}(M) + H(E_{\text{privA}}(M))$
    - yes conf, no integrity bc B cannot check the second chunk, he does not have privA, no NR **
  - $E_{\text{shared}}(M) + H(E_{\text{pubB}}(M))$
    - yes conf, yes integrity but the second encryption is wasteful, no NR bc everyone knows pubB so B can produce $H(E_{\text{pubB}}(M))$ ***
  - $E_{\text{privA}}(M) + H(E_{\text{shared}}(M))$
    - no conf bc everyone has pubA, yes integrity but the second encryption is wasteful, yes NR
  - $E_{\text{privA}}(M) + H(E_{\text{privA}}(M))$
    - no conf bc everyone has pubA, no integrity because B cannot verify the second chunk, no NR **
  - $E_{\text{privA}}(M) + H(E_{\text{pubB}}(M))$
    - no conf bc everyone has pubA, yes integrity for regular, non-random message from just $E_{\text{privA}}(M)$, and we have integrity for random messages with the second chunk, yes NR chunk, yes NR ***