Encryption Modes

CIS-4040

Peter Chapin

Why Modes?

- Encryption algorithms can be used several different ways (modes)
- Each mode has relative pros and cons
- Mode behavior independent of the underlying algorithm
  - All algorithms can be used in several modes
  - Strengths and weakness of a mode are specific to the mode, not the algorithm
- When communicating…
  - Partners must use the same algorithm (assumed known to attacker)
  - Partners must use the same mode (assumed known to attacker)
  - Partners must use the same key (**unknown to attacker**)

Mode Issue: Error Propagation

- *If a single bit error occurs in the cipher text, how much plain text is trashed?*
  - **At least one block will be trashed**; a good cipher has high diffusion
  - Ideally only one block will be trashed
  - Some modes might trash everything after the error

Mode Issue: Random Access

- If it is necessary to modify a random bit of plain text, how much cipher text needs to be decrypted and re-encrypted?
  - Ideally only one block should need to be processed
  - Some modes might require the decryption and re-encryption of all data after the modification…
  - … or before the modification…
  - … or both

Mode Issue: Block Cipher Decryption?

- Does the mode require the block cipher to be used as a decryptor?
  - Some modes use the block cipher as an encryptor even when decrypting
  - Advantage: If decryption is slow, you don’t have to worry about that
  - Advantage: Encryption can be one-way without any feasible way to decrypt

Mode Issue: Initialization Vector

- Does the mode need an IV and, if so, how should it be handled?
- Disadvantages…
  - IV is another piece of data that needs to be shared between partners
  - IV increases the size of the ciphertext beyond the size of the plain text
  - IV probably needs to be a *nonce*
    - In some modes the same IV (and key) will lead to the same key stream. If two plaintexts are encrypted with the same key stream, XORing the cipher texts together makes extracting both plain texts easy. Like the One Time Pad where the pad is used more than once!
  - Generating quality nonces can be hard

One Time Pad Approximation

- Stream Ciphers
  - Some modes work by using *E* to generate a “key stream” of pseudo-random bits and then XORing those bits into the plain text.
  - Like the One Time Pad…
    - … except with the OTP the key stream is truly random
    - … except the amount of information needed to generate the key stream is small: (IV, key, knowledge of the algorithm *E*)
  - Note: *Do not use the same key stream twice!*
    - This amounts to not using the same (IV, key) pair more than once
    - **Every message must have a new key or a new IV or both**