We spoke about defense challenges.

Crypto introduction:
- Secret key, public algorithms
- Symmetric, asymmetric crypto, one-way hashes

Attacks on cryptography:
- Cyphertext-only, known plaintext, chosen plaintext, MITM, brute-force

Types of ciphers:
- Mix of substitution and transposition
- Monoalphabetic, homophonic, polygram, polyalphabetic

Can You Decrypt This?
- Bpqa kzgxbwozixpg ammua zmit miag. Em eivb uwzm!
- Answer 1:
  - The call of death is a call of love. Death can be sweet if we answer it in the affirmative, if we accept it as one of the great eternal forms of life and transformation.
  - Broken by using frequency analysis
- Answer 2:
  - This cryptography seems real easy. We want more!
  - Ceasar cipher with offset 18, symmetric
  - Q: If we use offset 18 to encrypt and offset 8 to decrypt how is this symmetric crypto?

Substitution
- Polyalphabetic – many monoalphabetic ciphers are used sequentially
  - First mapping is used for the first letter, second mapping for the second letter and so on
  - XOR is a polyalphabetic cipher in binary domain

One-Time Pad
- Polyalphabetic cipher with infinite key
  - Combine letters from the message with the letters from an infinite key, randomly generated
  - Never reuse the key
  - Key needs to be generated using a very good RNG (to avoid any patterns)
  - This cipher cannot be broken
  - Sender and receiver must be perfectly synchronized

Symmetric Crypto Algorithms
- Stream ciphers: polyalphabetic
  - Work on message a bit or a byte at a time
  - Same bit/byte will encrypt differently, depending on the position of the key
- Block ciphers: polygram
  - Work on message block by block
  - Block size is usually the same as key size
  - Same plaintext block may encrypt into the same ciphertext block, depending on the cipher mode
  - Assume XOR with the key
Stream Cipher Example

plaintext: SANTA CLAUS SANTA CLAUS
key: SUPERCALIFRAGILISTIC
Ciphertext: LVDYSTMDYKBUCMLEUDV

Bonus question: What was the encryption algorithm I used here?

Stream Ciphers
- If Eve can get hold of plaintext/cyphertext pair she can retrieve the key
- Keystream is generated continuously and is the function of the secret stored inside the RNG
- Key should be pseudorandom – hard to break but easily reproduced for decryption
- Security depends entirely on RNG generating the key

Keystream Generators
- Random Number Generators
- Block Ciphers
- Chaining

Keystream Generator

Synchronous Stream Cipher
- Keystream is generated from the key K
- Sender and receiver must be synchronized
- One-bit error in ciphertext produces one-bit error in plaintext
- Upon loss of synchronization both sides start afresh with a new key
- Any deletions and insertions will cause loss of synchronization
- Mallory can toggle/change bits

Self-Synchronizing Stream Cipher
Self-Synchronizing Stream Cipher
- Internal state is the function only of the previous \( n \) ciphertext bits and depends on the key \( K \)
- Decryption keystream generator will completely synchronize with encryption generator after receiving \( n \) bits
- Advantage:
  - Recovery from loss of bits after \( n \) bits
- Drawback:
  - Error extension – one-bit error in ciphertext produces \( n \) errors in plaintext
  - Mallory can replay messages

Generating Random Numbers
- We need to generate a sequence that looks random but is reproducible
- There shouldn’t be any obvious regularities, otherwise Eve can learn the pattern after seeing several numbers, and guess the next ones
- We would like to cover the whole range of numbers (e.g. \( 2^n \) if the number has \( n \) bits)

Linear Congruential Generators
- Generators of the form
  \[ X_n = (aX_{n-1} + b) \mod m \]
  - A period of a generator is number of steps before it repeats the sequence
  - If \( a, b \) and \( m \) are properly chosen, this generator will be maximal period generator and have period of \( m \)
  - It has been proven that any polynomial congruential generator can be broken

Linear Feedback Shift Registers
- Used for cryptography today
- A shift register is transformed in every step through feedback function
  - Contents are shifted one bit to the right, the bit that ‘falls out’ is the output
  - New leftmost bit is XOR of some bits in the shift register – tap sequence
  - If we choose a proper tap sequence period will be \( 2^n - 1 \)

Linear Feedback Shift Registers
- \( X^n = X^4 \oplus X^1 \)

<p>| | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1111</td>
<td>0110</td>
<td>1000</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>0111</td>
<td>0011</td>
<td>0100</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>1011</td>
<td>1001</td>
<td>1110</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>0101</td>
<td>0100</td>
<td>1111</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>1010</td>
<td>0010</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1101</td>
<td>0001</td>
<td>0</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Linear Feedback Shift Registers
- Proper tap sequences are those where a polynomial from a tap sequence + 1 is a primitive polynomial in GF(2)
- There are tables of primitive polynomials
- LFSR is fast in hardware but slow in software
- LFSR are not themselves secure but they are used as building blocks in encryption algorithms
Block Cipher Example

plaintext: SANTA CLAUS SANTA CLAUS
key: SUPER

Block Cipher Example

plaintext: SANTA CLAUS SANTA CLAUS
key: SUPER

Block Cipher Example

plaintext: SANTA CLAUS SANTA CLAUS
key: SUPER

Block Cipher Example

plaintext: SANTA CLAUS SANTA CLAUS
key: SUPER

Block Cipher Example

plaintext: SANTA CLAUS SANTA CLAUS
key: SUPER

Block Cipher Example

plaintext: SANTA CLAUS SANTA CLAUS
key: SUPER

Block Encryption In Rounds

round: S S S S S S S
substitution
permutation
Encrypting A Large Message

- Electronic Code Book (ECB)
- Cipher Block Chaining (CBC)
- $k$-bit Cipher Feedback Mode (CFB)
- $k$-bit Output Feedback Mode (OFB)

Things to consider:
- Can we encrypt/decrypt efficiently (as soon as bits arrive)
- How hard it is to break encryption
- What if a bit is flipped on the channel
- What if we lose a bit on the channel

Electronic Code Book (ECB)

- Store mapping for every possible block
  - Fast encryption/decryption – just a table lookup
  - Ability to process text in any order and in parallel
  - Table size could be enormous so we need to make the mapping depend on the key
- Eve can detect which blocks map to other blocks, by seeing several plaintext and corresponding ciphertext messages
- Due to language redundancy even partial decryption might provide enough information
- Bit error invalidates one block
- Bit loss/addition is not recoverable

Block Replay

Mallory does this couple of times, looks for similar block sequences.
She can now replay 12B7 7783 38AC CDC7 at will

Cipher Block Chaining (CBC)

- Problem with ECB is that Mallory can replace, add or drop blocks at will
- Chaining prevents this by adding feedback
  - Each ciphertext block depends on all previous blocks
- Also, with CBC, same plaintext blocks will encrypt to different ciphertext blocks thus obscuring patterns in plaintext

Cipher Block Chaining (CBC)

- Initialization vector (IV) is just a block of random numbers, to ensure that no messages have the same beginning.
  - Both the sender and the receiver must use the same IV.
Cipher Block Chaining (CBC)

- An error in plaintext affects the rest of the message but is easily spotted and removed after decryption.
- An error in ciphertext affects one block and several bits of plaintext.

Error Recovery

- IV must be unique, otherwise it opens a vulnerability.
- If a \( k \)-bit unit is lost or added, next \( n/k-1 \) units will be garbled but then the algorithm will recover from error.
- One-bit error in ciphertext produces one-bit error in plaintext and \( n/k-1 \) subsequent plaintext units are garbled (\( n \) is the block size, \( k \) is the unit size).

Potential Problems With CBC

- Mallory can:
  - Add blocks
  - Drop blocks
  - Introduce bit errors
- Bit loss/addition is not recoverable.

Cipher-Feedback Mode (CFB)

- Similar to CFB but unit is taken from the output queue, not from the ciphertext.
- IV is placed in input queue and encrypted, leftmost unit is XOR-ed with one plaintext unit and sent.

Output-Feedback Mode (OFB)

- IV is placed in input queue and encrypted, leftmost unit is XOR-ed with one plaintext unit and sent.
Output-Feedback Mode (OFB)

- Leftmost unit from the output block is added to the right to input queue, and next plaintext unit is processed
- Output block generation can be done offline, plaintext is then just XOR-ed when it arrives
- One-bit error in ciphertext produces one-bit error in plaintext
- Bit loss/addition is not recoverable

Which Cipher Is The Best?
- Stream ciphers can be analysed mathematically and can be efficiently implemented in hardware
- Block ciphers are more general and can be efficiently implemented in software
- ECB is easiest and fastest but also weakest. Can be used for encrypting random data, such as other keys.
- CBC is good for encrypting files, no danger of lack of synchronization
- CFB is good for encrypting streams of characters
- OFB is good if error propagation cannot be tolerated