CMSC 426
Principles of Computer Security

Lecture 10

Introduction to Cryptography

Last Class We Covered

- Demo of malware analysis

Any Questions from Last Time?

Today's Topics

- Introduction to crypto
  - Definitions
  - Ciphers
- Symmetric Encryption
- Block ciphers
  - DES
  - 3DES
  - AES

Crypto- Definitions

Crypto-

- Cryptography
  - “Hidden writing”
  - Creation and use of secret codes and data-related security measures
- Cryptanalysis
  - Theory and practice of “breaking” cryptographic protocols
  - “Breaking” means recovering protected text/bypassing security
- Cryptology
  - The study of coded messages
  - Scientific study of codes: creating, using, analyzing, “breaking”

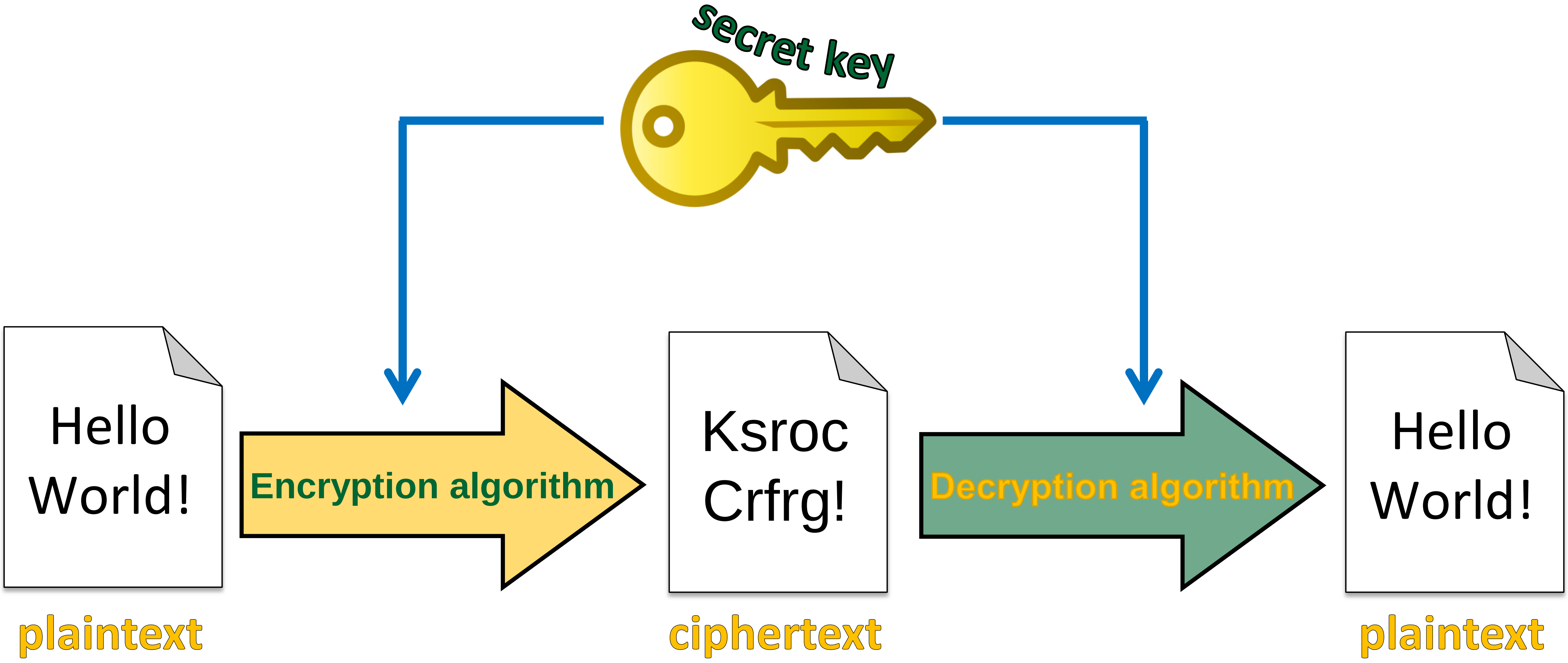
Encryption Types

- Encryption
  - Turning plain text into encrypted, “protected” text
- Decryption
  - Returning encrypted text to a readable, plain text state

- Symmetric Encryption
  - Uses the same key for encryption and decryption
- Asymmetric Encryption
  - Uses different keys for encryption and decryption

Symmetric Encryption

Components of Symmetric Encryption



Historical Ciphers (Substitution)

- Caesar cipher
  - Used by Julius Caesar
- “Rotation” of the alphabet
  - Caesar always used it with a shift of 3
- Only 25 possible encryptions
  - Relatively easy to brute force
- “Dogs are great” → 7 shift → “Kvnz hyl nylha”

Historical Ciphers (Algorithms)

- Keyword cipher
  - Keyword "begins" the alphabet, rest follows in order

- Keyword of "Cryptography" results in the ciphertext alphabet

  CRYPTOGAHBDEFIJKLMNQSUVWXZ
  ABCDEFGHIJKLMNOPQRSTUVWXYZ

- "Computer Security" → encryption → "Yjfksqtm Ntysmhqx"
- For better security and readability, often shown in 5-letter blocks:
  - "YJFKX QTMNT YSMHQ X"

Historical Ciphers (Algorithms)

- Vigenère cipher
  - Keyword is repeated, and is used to shift plaintext into ciphertext

- For example, a keyword of "**DOGS**", to encrypt the following:

```
PRINCIPLESOFCOMPUTERSECURITY
DOGSDOGSDOGSDOGSDOGSDOGSDOGS
↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓
TGPGGXWEIHVYGDTIYILKWTJNVXAR
```

Simple Substitution Cipher Example

- Assume an “alphabet” of 38 characters: A-Z, 0-9, “ ”, and .

- The substitution cipher is random in this case – there is no keyword or simple reversal/shift of the alphabet
  - **`PX2LOB.1MWGSU0V5H6TYNF9K IA7QO3ZJRE4CD8`**

- What is the plaintext, ciphertext, encryption algorithm, secret key, and decryption algorithm in this case?
- How many possible permutations of the cipher are possible?

Cryptanalytic Attacks

- How many possible permutations are there of 38 characters?
  - 38! → 38 possibilities for the first letter, 37 for the second, etc.
- 523,022,617,466,601,111,760,007,224,100,074,291,200,000,000
  - 523 tredecillion, 22 duodecillion, 617 undecillion, 466 decillion, 601 nonillion, 111 octillion, 760 septillion, 7 sextillion, 224 quintillion, 100 quadrillion, 74 trillion, 291 billion, and 200 million
- That's a LOT!!!

Substitution Cipher Example

- Plaintext
- Ciphertext
  - Both are a message written in the 38-character alphabet
- Encryption algorithm
  - Application of the substitution cipher to the original message
- Secret Key
  - The substitution ciphered alphabet
- Decryption algorithm
  - Application of the inverse of the substitution cipher

Block Ciphers
(Symmetric Block Encryption)

Block Ciphers

- Process the plaintext in fixed-size “blocks” (hence the name)
- Ciphertext produced is of blocks of equal size

- Block ciphers are symmetric algorithms
  - Key remains the same for encryption and decryption
  - However, two separate algorithms for en/decryption

- Most commonly-used algorithms are DES, 3DES, and AES

Block Cipher Algorithms

- Sequence of rounds, made of permutations and substitutions
  - Each round has its own unique subkey value, derived from the key
- DES and 3DES both use a Feistel network structure
  - Basic encryption and decryption algorithm are the same
    - Only difference is the order in which subkeys are applied
  - 16 rounds of en/decryption
  - Makes use of XOR and substitution

Components of Block Ciphers

- Block size
  - Size in bits of a plaintext/ciphertext block (commonly 128 bits)
- Key size
  - Size in bits of the key (commonly 128 bits)
- Round function
  - Basic encryption function; iterated to form the encryption algorithm
- Number of rounds
  - The number of iterations of the round function
- Subkey algorithm
  - Algorithm that expands the key into multiple round keys

Feistel Networks

- Iterative structure used in the DES and 3DES algorithms
  - Split 64 bits of input into right and left blocks
  - Apply Feistel function to the right half of the data
  - XOR it with the left half of the data
  - Swap the two blocks for the next round

  - Each of the 16 rounds is identical
    - *(Which is why we swap the data's sides)*
    - Only difference is the subkey used in the Feistel function

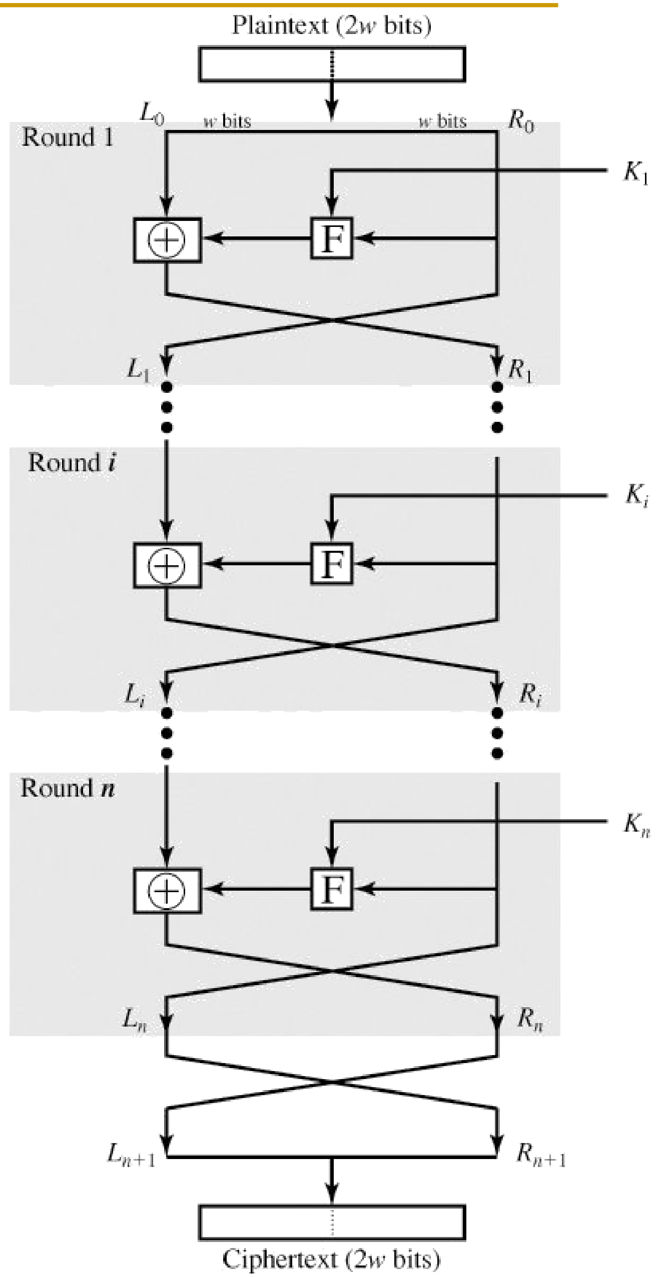
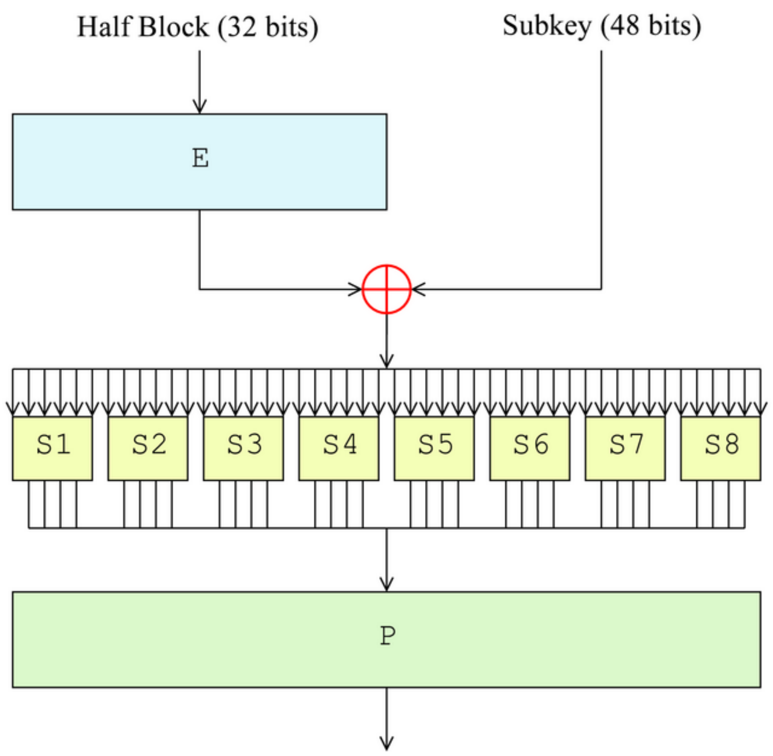


Image taken from Computer Security (Stallings & Brown)

Feistel Function



- Consists of four stages, done on 32 bits of data
- Expansion: 32 bits is expanded to 48 bits (eight 6 bit pieces, which each contain a copy of the adjacent bit on each side)
- Key mixing: XOR'd with 48-bit subkey
- Substitution: divided into eight 6 bit pieces again, which are processed by the substitution boxes (S-box)
  - Turns 6 bits in 4 bits according to a non-linear transformation (provided by a lookup table)
  - **Core component of the security of DES; without these, it would be trivial to break**
- Permutation: outputs are rearranged according to a fixed permutation, so that the same bits don't go through the same substitution box again together

Image and information taken from https://en.wikipedia.org/wiki/Data_Encryption_Standard

DES (Data Encryption Standard)

- Blocks are 64 bits
- Key is 56 bits
  - Actually 64 bits, but every 8th bit is a parity bit
- Algorithm itself is very secure
  - Very well-studied, and no reported fatal weaknesses
- Key size is woefully small
  - Only 72,000,000,000,000,000 possible keys (72 quadrillion)
  - Can be brute-forced by a powerful machine in about an hour
- Adopted in 1977, but not used widely since the 90s

Triple DES (or 3DES)

- Uses 3 keys, for a total key size of 168 bits
  - Either two or three independent keys, depending on implementation

- To encrypt, it applies the original DES algorithm as follows:
  - Encrypt with key 1
  - Decrypt with key 2
  - Encrypt with key 3
  - (If only two keys used, duplicate is used for key 1 and key 3)

- Three times as slow as DES… not good for large encryption jobs

AES (Advanced Encryption Standard)

Advanced Encryption Standard

- AES is also a block cipher, but does not use Feistel networks
  - Instead of splitting data in half and using one half to modify the other, AES processes the entire data block at once

- Block length is 128 bits, and key can be 128, 192, or 256 bits
  - For purposes of this class, we'll assume the key is always 128 bits
    - With 128 bits, this means that AES performs 10 rounds

- Decryption is still performed with keys applied in reverse
  - But encryption and decryption algorithms are not identical as in DES

AES Algorithm Overview

- 128 bits of input are represented as a 4 by 4 array of bytes

$a_{0,0}$	$a_{0,1}$	$a_{0,2}$	$a_{0,3}$
$a_{1,0}$	$a_{1,1}$	$a_{1,2}$	$a_{1,3}$
$a_{2,0}$	$a_{2,1}$	$a_{2,2}$	$a_{2,3}$
$a_{3,0}$	$a_{3,1}$	$a_{3,2}$	$a_{3,3}$

- Four different stages are performed in each round
  - Substitute Bytes
  - Shift Rows
  - Mix Columns
  - Add Round Key

substitution steps

Skipped in last round

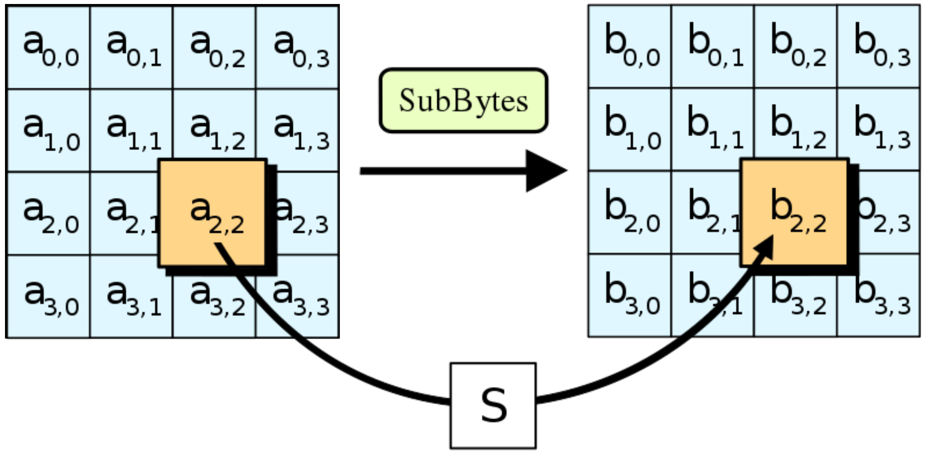
permutation step

Also occurs before the rounds begin

Image and information taken from https://en.wikipedia.org/wiki/Advanced_Encryption_Standard

Substitute Bytes

- Uses an S-box to perform a table lookup that allows for a byte-by-byte substitution of the block

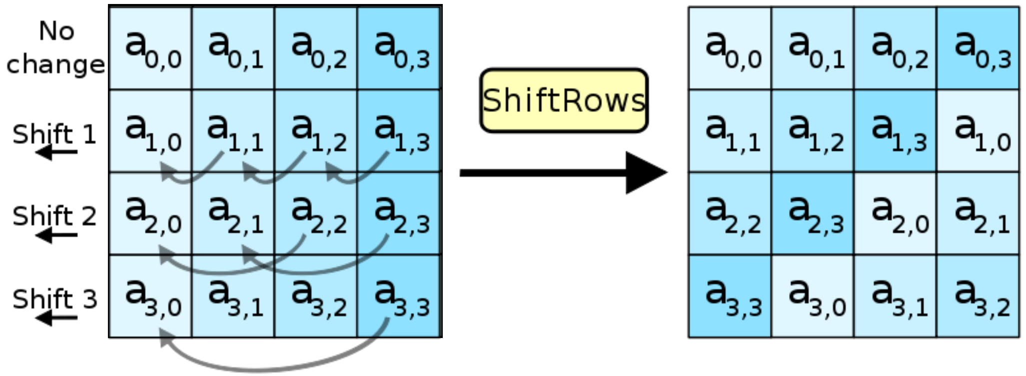


- Provides the non-linearity in the cipher
  - S-box is derived based on information from the key, using complex math we won’t cover in this class
    - (Multiplicative inverse, affine transformation, etc.)
  - When decrypting, this is the step that differs, using an “inverse” S-box

Image and information taken from https://en.wikipedia.org/wiki/Advanced_Encryption_Standard

Shift Rows

- Each row is shifted by an offset
  - This means that each column now contains information from each row

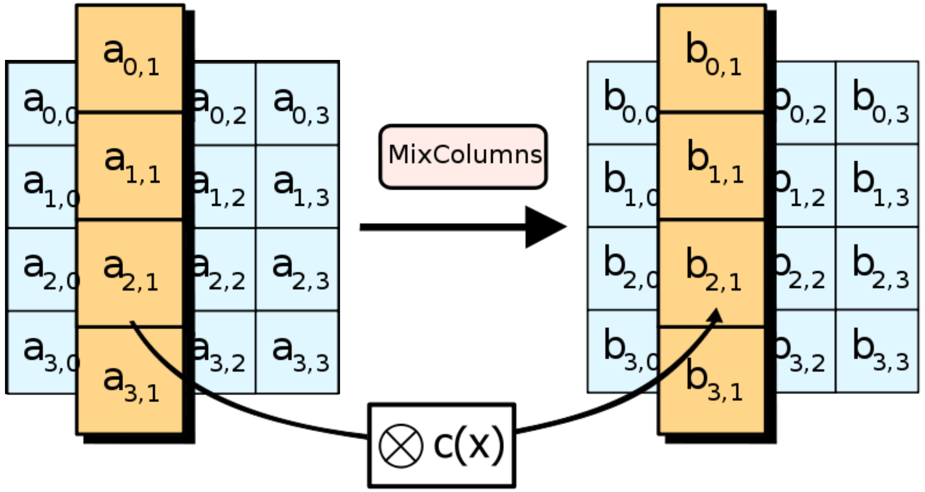


- This prevents the columns in the 4 by 4 array from being encrypted together throughout all the rounds

Image and information taken from https://en.wikipedia.org/wiki/Advanced_Encryption_Standard

Mix Columns

- Each column is altered, taking in the four bytes of the column, and outputting four bytes
- Each input byte affects all four output bytes (more math)
- This step does not occur in the final round of the algorithm



$$\begin{bmatrix} b_{0,j} \\ b_{1,j} \\ b_{2,j} \\ b_{3,j} \end{bmatrix} = \begin{bmatrix} 2 & 3 & 1 & 1 \\ 1 & 2 & 3 & 1 \\ 1 & 1 & 2 & 3 \\ 3 & 1 & 1 & 2 \end{bmatrix} \begin{bmatrix} a_{0,j} \\ a_{1,j} \\ a_{2,j} \\ a_{3,j} \end{bmatrix} \qquad 0 \le j \le 3$$

Image and information taken from https://en.wikipedia.org/wiki/Advanced_Encryption_Standard

Add Round Key

- Before the rounds began, the 128 bit key was expanded into an array of subkeys for each round
- Simple bitwise XOR of the current block with that round's subkey
- This stage also occurs at the beginning, before the rounds have properly begun

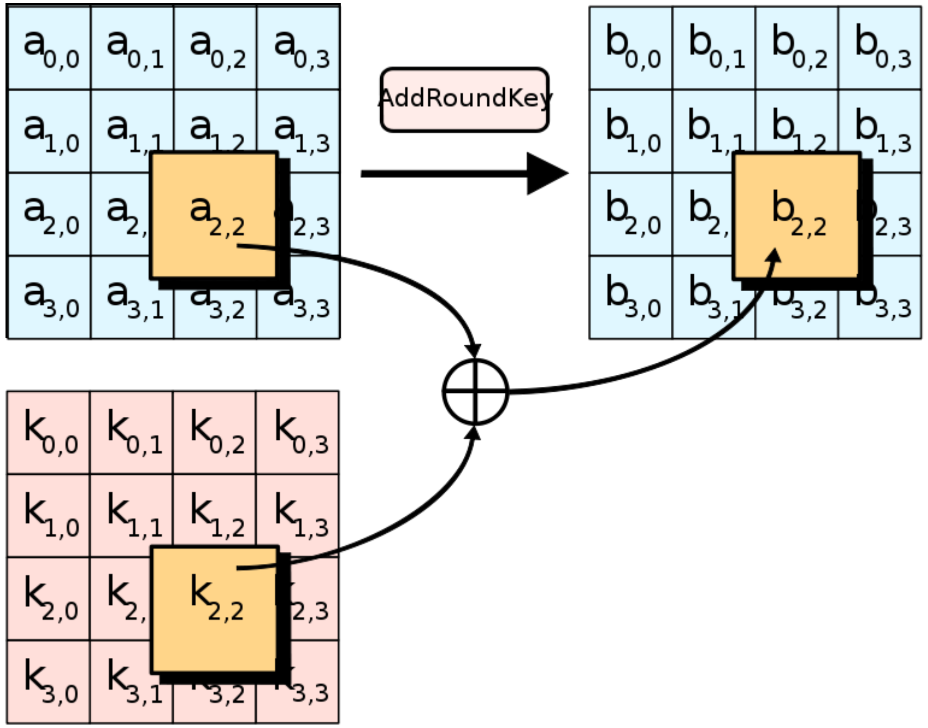


Image and information taken from https://en.wikipedia.org/wiki/Advanced_Encryption_Standard

Important (Crypto) Terms

Confusion and Diffusion

- Important concepts in cryptography and evaluating effectiveness

- Confusion
  - Each bit of the ciphertext should depend on several parts of the key
  - Obscures the connection between key and outcome

- Diffusion
  - If a single bit of the plaintext changes, (statistically) about half of the bits in the ciphertext should change

Cipher Block Modes of Operation

- Block ciphers themselves are only good for encrypting a block
  - Repeatedly applying a block cipher to larger amounts of data requires selecting a mode of operation
  - Some modes require an Initialization Vector (IV) to get started
- Different modes of operation exist for different purposes
  - Efficiency
  - Encrypting a stream of data
  - Parallelizing encrypt and/or decrypt
    - Can heavily affect speed

Parallelization

- Completing components of a single task in parallel, using multiple processing elements (*e.g.*, multi-core CPUs)

- Not always possible to achieve
  - Task must be able to be broken into discrete parts
  - Discrete parts can<u>not</u> be dependent on each another

- Parallelization can result in a <u>massive</u> speedup for a task
  - Highly desirable when, for example, encrypting large amounts of data

Announcements

- Homework 1 is due Wednesday night

- Lab 2 coming out later today
  - Will be due the Thursday after Spring Break
  - Much more of a walkthrough than Lab 1, but pay attention!

- Midterm 1 is happening next class

Image Sources

- Golden key:
  - https://commons.wikimedia.org/wiki/File:Golden_key_icon.svg
- Caesar cipher disk (adapted from):
  - https://en.wikipedia.org/wiki/File:CipherDisk2000.jpg
- Vigenère:
  - https://commons.wikimedia.org/wiki/File:Vigenere.jpg