CNS: EXPERIMENT - 5

Aim: To understand how to Encrypt long messages using various modes of operation using AES and DES.

Theory:
DES (Data Encryption Standard) and AES (Advanced Encryption Standard) are two widely used symmetric-key encryption algorithms that serve to protect data confidentiality by converting plaintext data into ciphertext using a secret key. Here's a brief overview of both:

1. DES (Data Encryption Standard):

- Key Length: DES uses a 56-bit encryption key. This relatively short key length is one of the primary reasons why DES is no longer considered secure against modern attacks.
- Block Size: DES operates on 64-bit blocks of plaintext data.
- Encryption Process: DES uses a Feistel network structure. The encryption process involves multiple rounds (typically 16 rounds). During each round, the plaintext block is divided into two halves, and various mathematical operations, including substitution (S-boxes), permutation (P-boxes), and bitwise operations, are applied to each half using a round-specific subkey derived from the main encryption key. The results from each round are mixed and swapped, creating the ciphertext.
- Security Concerns: DES is no longer considered secure against modern cryptographic attacks, primarily due to its short key length. It can be vulnerable to brute-force attacks where an attacker tries all possible 2^{56} keys to decrypt the data.

2. AES (Advanced Encryption Standard):

- Key Length: AES supports multiple key lengths, including 128-bit, 192-bit, and 256-bit keys. Longer key lengths provide higher security.
- Block Size:AES operates on 128-bit blocks of plaintext data.
- Encryption Process: AES uses a substitution-permutation network (SPN) structure. The encryption process involves several rounds, with the number of rounds depending on the key length (10 rounds for 128-bit keys, 12 rounds for 192-bit keys, and 14 rounds for 256-bit keys). Each round consists of several operations, including a substitution step (SubBytes), permutation step (ShiftRows), mixing step (MixColumns), and adding a round key (XOR with a round-specific key derived from the main encryption key).
- Security: AES is widely regarded as highly secure against both brute-force and cryptographic attacks when used with sufficient key lengths. It has withstood extensive scrutiny and is widely adopted in various applications, including data encryption, secure communication protocols, and more.

Review Questions

1. **Define confusion and diffusion. Give specific examples of how DES and AES achieve these properties.**
   Confusion refers to making the relationship between the ciphertext and the encryption key as complex as possible. It prevents attackers from deducing the key, even if they analyze the ciphertext. It's done by using non-linear transformations such as substitution boxes (S-boxes), which replace input bits with output bits in a non-linear, unpredictable way.
   Examples:
   - DES: Uses 8 different S-boxes, each mapping 6 input bits to 4 output bits. This substitution is non-linear, providing confusion by hiding how specific bits of the key affect the ciphertext.
   - AES: Uses the SubBytes step, which applies an invertible S-box to each byte. This makes the ciphertext highly non-linear with respect to both the plaintext and key.

   Diffusion spreads the influence of a single plaintext or key bit across many ciphertext bits. It ensures that changing one bit of plaintext or key affects many bits of the ciphertext, making patterns harder to detect. It's done by using permutation operations or mixing transformations.
   Examples:
   - DES: Diffusion comes from the expansion (E-box) and P-permutation, which spread changes across many bits over rounds.
   - AES: Diffusion comes from ShiftRows (spreads bytes across columns) and MixColumns (mixes bytes within each column so one change affects the whole column).

2. **Describe the key expansion process for both DES and AES. How does the size of the key affect the number of rounds in AES?**
   In DES, the key is 64 bits, but only 56 bits are used (8 bits are for parity). The 56 bits are split into two halves of 28 bits. For each round, the halves are shifted left (by 1 or 2 bits depending on the round), and then 48 bits are selected through a permutation (PC-2). This process generates 16 different 48-bit subkeys, one for each round.
   In AES, the key can be 128, 192, or 256 bits. The key expansion (Rijndael key schedule) generates a separate round key for each round. It uses operations like RotWord (cyclic byte shift), SubWord (S-box substitution), and addition of round constants (Rcon). The expanded keys are then used in each round of encryption. The size of the key determines the number of rounds:
   - AES-128 uses 10 rounds
   - AES-192 uses 12 rounds
   - AES-256 uses 14 rounds

   Thus, larger keys require more rounds to provide sufficient security.

3. **What is the importance of the Initialisation Vector(IV) and CTR?**
   The Initialization Vector (IV) is a random or unique value used in block cipher modes (like CBC, CFB, or OFB). Its importance is that it ensures the same plaintext encrypted with the same key will produce different ciphertexts, preventing attackers from noticing patterns. The IV itself is not secret but must be unpredictable or unique for each encryption session.

   The Counter (CTR) mode uses a counter value (combined with a nonce/IV) that is encrypted with the key, and the result is XORed with the plaintext. The counter is incremented for each block. Its importance is that it allows parallel encryption and decryption, increases efficiency, and turns a block cipher into a stream cipher. It also ensures that even if the same plaintext is encrypted twice with the same key, the output will differ (due to different IV/counter values).

4. **Compare the computational complexity of implementing DES versus AES based on your practical experience. Which algorithm is more resource-intensive and why?**
   In practice, AES is more computationally efficient than DES, even though AES is newer and uses larger block sizes.

   DES works on 64-bit blocks with a 56-bit key and requires 16 rounds of operations. Its design uses many bit-level permutations (P-boxes) and S-box substitutions, which are not very efficient on modern processors. Because of this, DES tends to be slower and less optimized for software.

   AES, on the other hand, works on 128-bit blocks and supports 128/192/256-bit keys. It requires 10, 12, or 14 rounds depending on key size. Although AES involves more complex math, its operations are byte-oriented and well-suited for modern hardware. Many CPUs even have built-in AES instructions, making it much faster.

   Therefore, AES is less resource-intensive in practice compared to DES, despite having more rounds, because it is designed for efficiency on both hardware and software platforms. DES is considered heavier due to its outdated bit-level operations and weaker key size.

OUTPUT

Triple DES Encryption

Enter Plain Text to Encrypt

i love cns

Select Cipher Mode of Encryption

ECB

Select Padding

PKCS5Padding

Enter Secret Key

shaktishaktishaktishakti

Output Text Format Base64 Hex

Encrypt

DES Encrypted Output

HIN44qNzWm1iv3aLJSC6uQ==

Triple DES Online Decryption

DES Encrypted Text

HIN44qNzWm1iv3aLJSC6uQ==

Select Cipher Mode of Decryption

ECB

Select Padding

PKCS5Padding

Enter Secret Key

shaktishaktishaktishakti

Output Text Format Base64 Plain-Text

Decrypt

Triple DES Decrypted Output

i love cns

Triple DES Encryption

Enter Plain Text to Encrypt

cns is my fav

Select Cipher Mode of Encryption

CBC

Select Padding

PKCS5Padding

Enter IV (Optional)

hello yo

Enter Secret Key

shaktishaktishaktishakti

Output Text Format Base64 Hex

Encrypt

DES Encrypted Output

KCG9gAnA7kUZl6Vvsg/5UQ==

Triple DES Online Decryption

DES Encrypted Text

KCG9gAnA7kUZl6Vvsg/5UQ==

Select Cipher Mode of Decryption

CBC

Select Padding

PKCS5Padding

Enter IV Used During Encryption(Optional)

hello yo

Enter Secret Key

shaktishaktishaktishakti

Output Text Format Base64 Plain-Text

Decrypt

Triple DES Decrypted Output

cns is my fav

AES Encryption

Enter Plain Text to Encrypt

i am shakti

Select Cipher Mode of Encryption

ECB

Select Padding

PKCS5Padding

Key Size in Bits

128

Enter Secret Key

shaktishaktishak

Output Text Format Base64 Hex

Encrypt

AES Encrypted Output

m86S+wnTDDNBoBH7xeDBqQ==

AES Decryption

AES Encrypted Text

m86S+wnTDDNBoBH7xeDBqQ==

Select Cipher Mode of Decryption

ECB

Select Padding

PKCS5Padding

Key Size in Bits

128

Enter Secret Key used for Encryption

shaktishaktishak

Output Text Format Plain-Text Base64

Decrypt

AES Decrypted Output

i am shakti

AES Encryption

Enter Plain Text to Encrypt

i am not shakti

Select Cipher Mode of Encryption

ECB

Select Padding

PKCS5Padding

Key Size in Bits

192

Enter Secret Key

shaktishaktishaktishakti

Output Text Format Base64 Hex

Encrypt

AES Encrypted Output

z9PpYqOldPJBBJ+KEodkRA==

AES Decryption

AES Encrypted Text

z9PpYqOldPJBBJ+KEodkRA==

Select Cipher Mode of Decryption

ECB

Select Padding

PKCS5Padding

Key Size in Bits

192

Enter Secret Key used for Encryption

shaktishaktishaktishakti

Output Text Format Plain-Text Base64

Decrypt

AES Decrypted Output

i am not shakti

AES Encryption

Enter Plain Text to Encrypt

i am the flash

Select Cipher Mode of Encryption

ECB

Select Padding

PKCS5Padding

Key Size in Bits

256

Enter Secret Key

shaktishaktishaktishaktishaktizz

Output Text Format Base64 Hex

Encrypt

AES Encrypted Output

Mm1XgYi55EgrjayYfeq0sQ==

AES Decryption

AES Encrypted Text

Mm1XgYi55EgrjayYfeq0sQ==

Select Cipher Mode of Decryption

ECB

Select Padding

PKCS5Padding

Key Size in Bits

256

Enter Secret Key used for Encryption

shaktishaktishaktishaktishaktizz

Output Text Format Plain-Text Base64

Decrypt

AES Decrypted Output

i am the flash

AES Encryption

Enter Plain Text to Encrypt

i am spiderman

Select Cipher Mode of Encryption

CBC

Select Padding

PKCS5Padding

Enter IV (Optional)

helloworld123456

Key Size in Bits

128

Enter Secret Key

shaktishaktishak

Output Text Format Base64 Hex

Encrypt

AES Encrypted Output

6G30Xw11jXYAfmDfp40gCw==

AES Decryption

AES Encrypted Text

6G30Xw11jXYAfmDfp40gCw==

Select Cipher Mode of Decryption

CBC

Select Padding

PKCS5Padding

Enter IV Used During Encryption(Optional)

helloworld123456

Key Size in Bits

128

Enter Secret Key used for Encryption

shaktishaktishak

Output Text Format Plain-Text Base64

Decrypt

AES Decrypted Output

i am spiderman

AES Encryption

Enter Plain Text to Encrypt

i am spiderman

Select Cipher Mode of Encryption

CBC

Select Padding

PKCS5Padding

Enter IV (Optional)

helloworld123456

Key Size in Bits

192

Enter Secret Key

shaktishaktishaktishakti

Output Text Format Base64 Hex

Encrypt

AES Encrypted Output

2o5e2NrWQ8TeYPcNd1hL/w==

AES Decryption

AES Encrypted Text

2o5e2NrWQ8TeYPcNd1hL/w==

Select Cipher Mode of Decryption

CBC

Select Padding

PKCS5Padding

Enter IV Used During Encryption(Optional)

helloworld123456

Key Size in Bits

192

Enter Secret Key used for Encryption

shaktishaktishaktishakti

Output Text Format Plain-Text Base64

Decrypt

AES Decrypted Output

i am spiderman

AES Encryption

Enter Plain Text to Encrypt

i am spiderman

Select Cipher Mode of Encryption

CBC

Select Padding

PKCS5Padding

Enter IV (Optional)

helloworld123456

Key Size in Bits

256

Enter Secret Key

shaktishaktishaktishaktishaktiok

Output Text Format Base64 Hex

Encrypt

AES Encrypted Output

wOlBHek8V33r7VBOdBxZqg==

AES Decryption

AES Encrypted Text

wOlBHek8V33r7VBOdBxZqg==

Select Cipher Mode of Decryption

CBC

Select Padding

PKCS5Padding

Enter IV Used During Encryption(Optional)

helloworld123456

Key Size in Bits

256

Enter Secret Key used for Encryption

shaktishaktishaktishaktishaktiok

Output Text Format Plain-Text Base64

Decrypt

AES Decrypted Output

i am spiderman