Information Security & Cryptography

ASSIGNMENT- 7

U20CS005
BANSI MARAKANA

Use any crypto library (available in Java, Python/ C++/ .net) to implement AES and SHA.
Implementation of AES in Python:

```
from Crypto.Cipher import AES
from Crypto.Util.Padding import pad, unpad
from Crypto.Random import get_random_bytes
import base64

# Generate a random 128-bit AES key
key = get_random_bytes(16)
# Create an AES cipher object with CBC mode
cipher = AES.new(key, AES.MODE_CBC)
iv = cipher.iv
# Take input plaintext from user
plaintext = input("Enter plaintext: ").encode()
# Encrypt the plaintext
ciphertext = cipher.encrypt(pad(plaintext, AES.block_size))
# Print the ciphertext as base64
ciphertext_base64 = base64.b64encode(ciphertext).decode('utf-8')
print("Ciphertext (base64):", ciphertext_base64)
# Take input ciphertext from user
ciphertext_base64_input = input("Enter ciphertext (base64): ")
ciphertext_input = base64.b64decode(ciphertext_base64_input)
# Create a new AES cipher object with the same key and IV for decryption
cipher_dec = AES.new(key, AES.MODE_CBC, iv)
# Decrypt the ciphertext
decrypted_data = cipher_dec.decrypt(ciphertext_input)
plaintext = unpad(decrypted_data, AES.block_size)
# Print the decrypted plaintext
print("Decrypted plaintext:", plaintext.decode())
```

OUTPUT:

```
Enter plaintext: Bansi Marakana
Ciphertext (base64): UbueEbvI+TpW7GCKw7U4wg==
Enter ciphertext (base64): UbueEbvI+TpW7GCKw7U4wg==
Decrypted plaintext: Bansi Marakana
```

Implementation of SHA in Python:

```
import hashlib
# Take input text from user
plaintext = input("Enter plaintext: ").encode()
# Create a SHA-256 hash object
hash_object = hashlib.sha256()
# Update the hash object with the input text
hash_object.update(plaintext)
# Get the hexadecimal representation of the hash
hash_hex = hash_object.hexdigest()
# Print the hash
print("SHA-256 hash:", hash_hex)
```

OUTPUT:

```
Enter plaintext: Bansi Marakana
SHA-256 hash: b469ec71bef156e69ecbe341706aa30769420399fdd147b892d957be02c03815
```