Ministerul Educaţiei și Cercetării al Republicii Moldova
Universitatea Tehnică a Moldovei
Facultatea Calculatoare, Informatică și Microelectronică

Laboratory work 6:
Cryptography and Security

Elaborated:
st. gr. FAF-211 Echim Mihail

Verified:
asist. univ. Aureliu ZGUREANU

Chişinău - 2023

CRYPTOGRAPHY AND SECURITY

Subject: Hash functions and digital signatures

Task 2
Objective:
Perform key generation, sign, and validate the digital signature of message 'm' obtained during lab work number 2 using RSA signature. Ensure 'n' is at least 3072 bits. Choose the hash algorithm from the provided list based on the formula i = (k mod 24) + 1, where 'k' is the student's serial number in the group list, and 'i' represents the index of the hash function in the list.

Message Processing:
The code provided a lengthy text message, which was prepared for cryptographic processing.

Code Overview:
Hashing:
Utilized the SHA-3-256 hashing algorithm to create a digest of the message.
Converted the resulting hash into an integer and adjusted its size to match the desired hash size of 256 bits.

RSA Signature Generation:
Generated two large prime numbers (prime1 and prime2) using the generate_large_prime(bits) function.
Computed n as the product of these primes and derived phi_n as (prime1 - 1) * (prime2 - 1).
Selected a suitable public exponent e using the choose_public_exponent(phi_n) function.
Calculated the private exponent d using modular inverse.

Signature and Verification:
Generated the RSA signature for the hashed message using pow(hashed_message, d, n).
Validated the generated signature by applying the verification formula pow(signature, e, n) and compared it to the hashed message.

Conclusion:
The code successfully demonstrated the process of RSA signature generation and verification for the provided text message. The RSA signature was generated using prime numbers and their associated keys, ensuring the authenticity of the message through digital signature validation.

Task 3

Objective:

Perform the digital signature and validation of message 'm' obtained during laboratory work number 2. The signing will be done using the ElGamal signature scheme (with given 'p' and generator). Select the hash algorithm from the provided list according to the formula i = (k mod 24) + 1, where 'k' is the student's serial number in the group list, and 'i' represents the index of the hash function in the list.

Code Overview:

Message Processing:

The code provided a lengthy text message, which was prepared for cryptographic processing.

Hashing:

Utilized the SHA-3-256 hashing algorithm to create a digest of the message.
Converted the resulting hash into an integer and adjusted its size to match the desired hash size of 256 bits.

RSA Signature Generation:

Generated two large prime numbers (prime1 and prime2) using the generate_large_prime(bits) function.
Computed n as the product of these primes and derived phi_n as (prime1 - 1) * (prime2 - 1).
Selected a suitable public exponent e using the choose_public_exponent(phi_n) function.
Calculated the private exponent d using modular inverse.

Signature and Verification:

Generated the RSA signature for the hashed message using pow(hashed_message, d, n).
Validated the generated signature by applying the verification formula pow(signature, e, n) and compared it to the hashed message.

Conclusion:

The code successfully demonstrated the process of RSA signature generation and verification for the provided text message. The RSA signature was generated using prime numbers and their associated keys, ensuring the authenticity of the message through digital signature validation.