Ministry of Education, Culture and Research of the Republic of Moldova
Technical University of Moldova
Department of Software and Automation Engineering

REPORT

Laboratory work no. 6
Hash functions & Digital signatures

Elaborated:
st. gr. FAF-213 Botnari Ciprian

Verified:
asist. univ. Cătălin Mîțu

Chişinău – 2023

Table of Contents

Topic: Hash functions & Digital signatures

Tasks

1. Generate keys, create and validate the RSA digital signature for the message obtained in the second laboratory work. The value of n must be at least 3072 bits (925 digits). The hash algorithm to be applied is MD-512.
2. Generate keys, create and validate the ElGamal digital signature for the message obtained in the second laboratory work.. The values of prime and generator are given below. The hash algorithm to be applied is MD-256.

Theoretical notes

RSA, or Rivest-Shamir-Adleman, is a widely used asymmetric-key cryptosystem that offers secure communication and digital signatures. The RSA algorithm relies on the mathematical properties of large prime numbers. Here are the key phases of the RSA algorithm:

1. Key Generation: RSA begins with key generation, which involves selecting two large prime numbers, p and q. The product of these primes, n = p * q, becomes the modulus used for encryption and decryption. The public key (e, n) and private key (d, n) are generated, with e as the encryption exponent and d as the decryption exponent.
2. Encryption: To encrypt a message M, the sender uses the recipient's public key (e, n). The message is raised to the power of e and then taken modulo n to obtain the ciphertext $C = M^e \bmod n$.
3. Decryption: The recipient uses their private key (d, n) to decrypt the ciphertext C. The ciphertext is raised to the power of d and taken modulo n to recover the original message: $M = C^d \bmod n$.
4. Digital Signatures: RSA is also used for digital signatures. To sign a message, the sender applies their private key to a hash of the message, creating a signature. Recipients can verify the signature using the sender's public key.
5. Key Security: The security of RSA relies on the difficulty of factoring the modulus n, given that it is a product of two large prime numbers. The prime factorization problem forms the basis of RSA's security.

ElGamal is another asymmetric-key cryptosystem, primarily used for secure key exchange and digital signatures. It involves the following phases:

1. Key Generation: ElGamal starts with key generation. The sender selects a large prime number, p, and a primitive root modulo p, g. The sender also generates a private key, x, which is a random number, and computes the corresponding public key, $y = g^x \bmod p$.
2. Key Exchange: In the key exchange phase, two parties can securely exchange a shared secret by using their respective private keys and the other party's public key. This shared secret can be used as a symmetric key for encryption and decryption.
3. Encryption: ElGamal encryption is probabilistic and involves generating a random number, k. The message is encrypted as a pair of ciphertexts (c1, c2), where $c1 = g^k \bmod p$ and $c2 = (M * y^k) \bmod p$, with M being the plaintext message.
4. Decryption: The recipient uses their private key, x, to decrypt the ciphertext (c1, c2). They calculate the shared secret, S = c1^x mod p, and then compute the plaintext as M = (c2 / S) mod p.
5. Digital Signatures: ElGamal can also be used for digital signatures, where the sender signs a message using their private key, and the recipient verifies it using the sender's public key.

MD6-512 is a cryptographic hash function designed by Ronald Rivest. It is part of the MD (Message Digest) family of hash functions. MD6 is known for its high level of security and resistance to various cryptographic attacks.

1. Append the original message with a 1-bit, followed by a series of 0-bits, and then the length of the original message in bits (padded to 128 bits).
2. Set the initial state variables (w, k, d) based on the security parameter and desired hash size (512 bits in this case).
3. Initialize chaining variables by hashing the padded message with a compression function.
4. Divide the padded message into blocks of a fixed size.
5. For each block, apply the compression function to update the chaining variables.
6. Perform mixing and permutation operations on the state variables to ensure diffusion of input changes.
7. Concatenate the final chaining variables to obtain the hash value.

Results

Message to hash: modern western cryptology emerged directly from the flowering of modern diplomacy. the ambassadors' reports were sometimes opened and read, and, if necessary, crypt-analyzed. by the end of the century, cryptology had become important enough for most states to keep full-time cipher secretaries occupied in making up new keys, enciphering and deciphering messages, and solving intercepted dispatches.
sometimes the cryptanalysts were separate from the cipher secretaries and were called in only when needed. perhaps the most elaborate organization was venice's. it fell under the immediate control of the council of ten, the powerful and mysterious body that ruled the republic largely through its efficient secret police. venice owed her preeminence largely to giovanni soro, who was perhaps the west's first great cryptanalyst. soro, appointed cipher secretary in 1506, enjoyed remarkable success in solving the ciphers of numerous principalities. his solution of a dispatch of mark anthony coloana, chief of the army of the holy roman emperor maximilian i, requesting 20,000 ducats or the presence of the emperor with the army, gave an insight into colonna's problems. so great wassoro's fame that other courts sharpened their ciphers, and as early as 1510 the papal curia was sending him ciphers that no one in rome could solve. but venice had no monopoly. in 1589, henry of navarre, who was destined to become the mostpopular king in the history of france (he coined the slogan "a chicken inevery peasant's pot every sunday"), ascended to the throne as henry ivand found himself embroiled still more fiercely in his bitter contestwith the holy league, a catholic faction that refused to concede that aprotestant could wear the crown. the league, headed by the duke of mayenne, held paris and all the other large cities of france, and was receiving large transfusions of men and money from philip of spain. henry was tightly hemmed in, and it was at this juncture that some correspondence between philip and two of his liaison officers,commander juan de moreo and ambassador manosse, fell into henry'shands.it was in cipher, but he had in his government at the time onefrancois viete, the seigneur de la bigotiere, a 49-year-old lawyer from poitou who had risen to become counselor of the parlement, or court ofjustice, of tours and a privy counselor to henry. viete had for yearsamused himself with mathematics as a hobby"never was a man moreborn for mathematics," said tallement des reaux. as the man who firstused letters for quantities in algebra, giving that study its characteristiclook, viete is today remembered as the father of algebra. a year before,he had solved a spanish dispatch addressed to alessandro farnese, theduke of parma, who headed the spanish forces of the league. henryturned the new intercepts over to him to see if viete could repeat hissuccess. Here are the results for each one.

RSA

This is an example of a valid signature where the message hasn’t been tampered with.

md6 hash initial:
5f93cb5f8eb10fef92b00e6dbad040844ca12ad1b9ada32441bf3d8465b781a0d42fee4348eb0b678a7e21d534df01e3227c2caa85f4549dabd1c8f417b7d070

md6 hash current:
5f93cb5f8eb10fef92b00e6dbad040844ca12ad1b9ada32441bf3d8465b781a0d42fee4348eb0b678a7e21d534df01e3227c2caa85f4549dabd1c8f417b7d070

[Status] Signature is valid. Message is not tampered with.

This is an example of a invalid signature where the message has been tampered with by removing some lines from the text.
md6 hash initial:
5f93cb5f8eb10fef92b00e6dbad040844ca12ad1b9ada32441bf3d8465b781a0d42fee4348eb0b678a7e21d534df01e3227c2caa85f4549dabd1c8f417b7d070
md6 hash current:
55fe5064cbb936a58370fa336aa11aa53de24d02e60d53b929895bba7095622b0da6dad6682dfc190e3e448ddb2cda7333c68ed9cb3e55de5c5589115091ad61
[Status] Signature is invalid. Message is tampered with.

ElGamal

This is an example of a valid signature where the message hasn't been tampered with.
md6 hash initial: 784214a6ba0f9bb12c9e2f622dee9504e0c91e772730b001f45758dadc98e07a
md6 hash current: 784214a6ba0f9bb12c9e2f622dee9504e0c91e772730b001f45758dadc98e07a
[Status] Signature is valid. Message is not tampered with.

This is an example of a invalid signature where the message has been tampered with by clearing the whole file.
md6 hash initial: 784214a6ba0f9bb12c9e2f622dee9504e0c91e772730b001f45758dadc98e07a
md6 hash current: 2b0a697a081c21269514640aab4d74ffafeb3c0212df68ce92922087c69b0a77
[Status] Signature is invalid. Message is tampered with.

Conclusion

In conclusion, the exploration and application of RSA and ElGamal encryption algorithms, along with the utilization of hash functions MD-512 and MD-256 for digital signatures, have significantly advanced my comprehension of contemporary cryptographic methodologies. This study has provided hands-on experience in generating keys, creating, and verifying digital signatures, contributing to the broader landscape of secure communication and data integrity.

RSA, a widely adopted public-key cryptosystem, has demonstrated its prowess in ensuring data confidentiality and the verifiability of digital signatures. Its strength, rooted in the factorization problem, establishes RSA as a robust choice for securing digital transactions and communications in an era marked by increasing cyber threats.

ElGamal, another influential public-key encryption method, has showcased its relevance, particularly in key exchange and digital signature applications. With its security hinging on the discrete logarithm problem, ElGamal adds diversity to the arsenal of cryptographic tools, offering flexible solutions for securing various facets of digital communication.

As part of the practical exercises, the generation and validation of digital signatures have been executed using MD-512 for RSA and MD-256 for ElGamal. This hands-on experience underscores the crucial role of hash algorithms in fortifying the security of cryptographic systems, ensuring the integrity and authenticity of digital signatures.

GitHub

Sufferal/cryptography-labs (github.com)