Lab 6

Author: Lukas Kmitas

Alice and Bob wish to communicate with each other over the Internet. Each uses RSA, the common asymmetric cryptography protocol. Thus, each has his/her own private key and knows the public key of the other. Let us denote private key of Alice as *Pr(A)*, private key of Bob as *Pr(B)*, public key of Alice as *Pu(A)*, and public key of Bob as *Pu(B)*.

Please use the following notation in presenting your answers:

: Message *M* is encrypted using key *K*

: Message *M* is decrypted using key *K*

: One way hash or secure digest of message *M*

1. Alice wants to send a message M to Bob so that no one else can read it. Let us denote the message Alice sent as .

How would Alice send the message?

How would Bob decipher the message?

1. In this situation, Alice does not care if anyone can read her message. But she does care that no one in the middle can change the message (in an undetectable manner). Let us denote the message Alice sent as

If computational efficiency is a concern, how would Alice send the message?

What would Bob do to verify that the message indeed came from Alice?

**Q1)**

**Alice Sending the Message (M\_1)**

To ensure that no one other than Bob can read the message, Alice must encrypt her message with Bob's public key. This way, only Bob can decrypt the message using his private key, which is known only to him.

Step for Alice to send M\_1:

1. Alice encrypts the message M using Bob's public key, **Pu(B)**.
2. The encrypted message, denoted as **M\_1**, can be represented as **E\_{Pu(B)}(M)**.

**Bob Deciphering the Message**

Upon receiving the message **M\_1**, Bob will use his private key to decrypt it. Since the message was encrypted with Bob's public key, Bob's private key is the only key capable of decrypting it correctly.

Step for Bob to decipher M\_1:

1. Bob decrypts the received message **M\_1** using his private key, **Pr(B)**.
2. The decryption process can be denoted as **D\_{Pr(B)}(M\_1)**.
3. Since **M\_1 = E\_{Pu(B)}(M)**, the decryption process essentially retrieves the original message M, example. D\_{Pr(B)}(E\_{Pu(B)}(M)) = M.

This procedure ensures that the message remains confidential between Alice and Bob. Only Bob, who possesses the corresponding private key to the public key used for encryption, can decrypt and read the message sent by Alice.

**Q2)**

**Alice Sending the Message (M\_2)**

Step for Alice to send M\_2:

1. First, Alice computes a hash of the message M, obtaining H(M). This hash acts as a unique digest or fingerprint of the message.
2. Alice then encrypts this hash with her private key, Pr(A), creating a digital signature: E\_{Pr(A)}(H(M)).
3. Alice sends the original message M (which is M\_2 in this context) along with the digital signature.

The message M\_2 is sent in plain text (since confidentiality is not a concern), accompanied by the digital signature.

**Bob Verifying the Message**

Upon receiving the message M and the digital signature, Bob performs the following steps to verify its integrity and authenticity:

Step for Bob to verify M\_2:

1. Bob decrypts the received digital signature using Alice's public key, Pu(A). This step retrieves the original hash value that Alice computed: D\_{Pu(A)}(E\_{Pr(A)}(H(M))) = H(M).
2. Independently, Bob computes the hash of the received message M to get H(M').
3. Bob then compares the decrypted hash he obtained in step 1 with the hash he computed in step 2. If H(M) = H(M'), the message has not been altered, and it is verified to have come from Alice because only Alice has the private key used to create the original digital signature.

This process uses the principles of a digital signature and hashing to ensure message integrity and authenticity without requiring the message itself to be encrypted. This method is efficient compared to encrypting the entire message because hashing and signing typically involve less computational overhead than full message encryption.