SECURED GMT PROVIDER

ASSIGNMENT - 2

Submitted by:
Avantika Chhabra(2015MCS2334)
Vamsi Yalavarthi(2015MCS2358)

PROBLEM STATEMENT:

This application relates to providing a certified copy of the current time and date in a secure manner. In particular the client requests and obtain a digitally signed copy of the current GMT data and time. It is implemented using the access to a web based "GMT data and time server" which then returns a file which contains the current GMT data and time, suitably signed digitally with the server's RSA-based private key and is verified by the client using public key.

INTRODUCTION:

The system named as "Secured GMT Provider" is developed to fulfil the requirements of Assignment 2 of Network Security(SIL765). The system is developed to fetch secured GMT date and time from the GMT server.

This system offers the users a way to secure their information and provide authenticity and integration to the clients.

SOFTWARE MODULES:

There are basically two different modules.

1. Server Module
2. Client Module

SERVER MODULE: The main functions of this module are:

1. To fetch GMT time and date from the GMT server .
2. To digitally sign the GMT information using RSA algorithm.
3. To transfer the message as well as the signature using a file to the requested client.

CLIENT MODULE: The main functions of this module are:

1. Client requests for the current GMT date and time by clicking the link on client site.
2. Server responds with a text file containing the message and its digital signature to verify authenticity of the received message.

GENERATION OF KEYS:

To generate the digital signature we use the RSA algorithm imported from the libraries. The encryption is done using the private key of the server and it is decrypted by the public key of client which is obtained by key distribution server.

The private key is available only with the server and similarly the client only knows the corresponding public key of the server.

DIGITAL SIGNATURE:

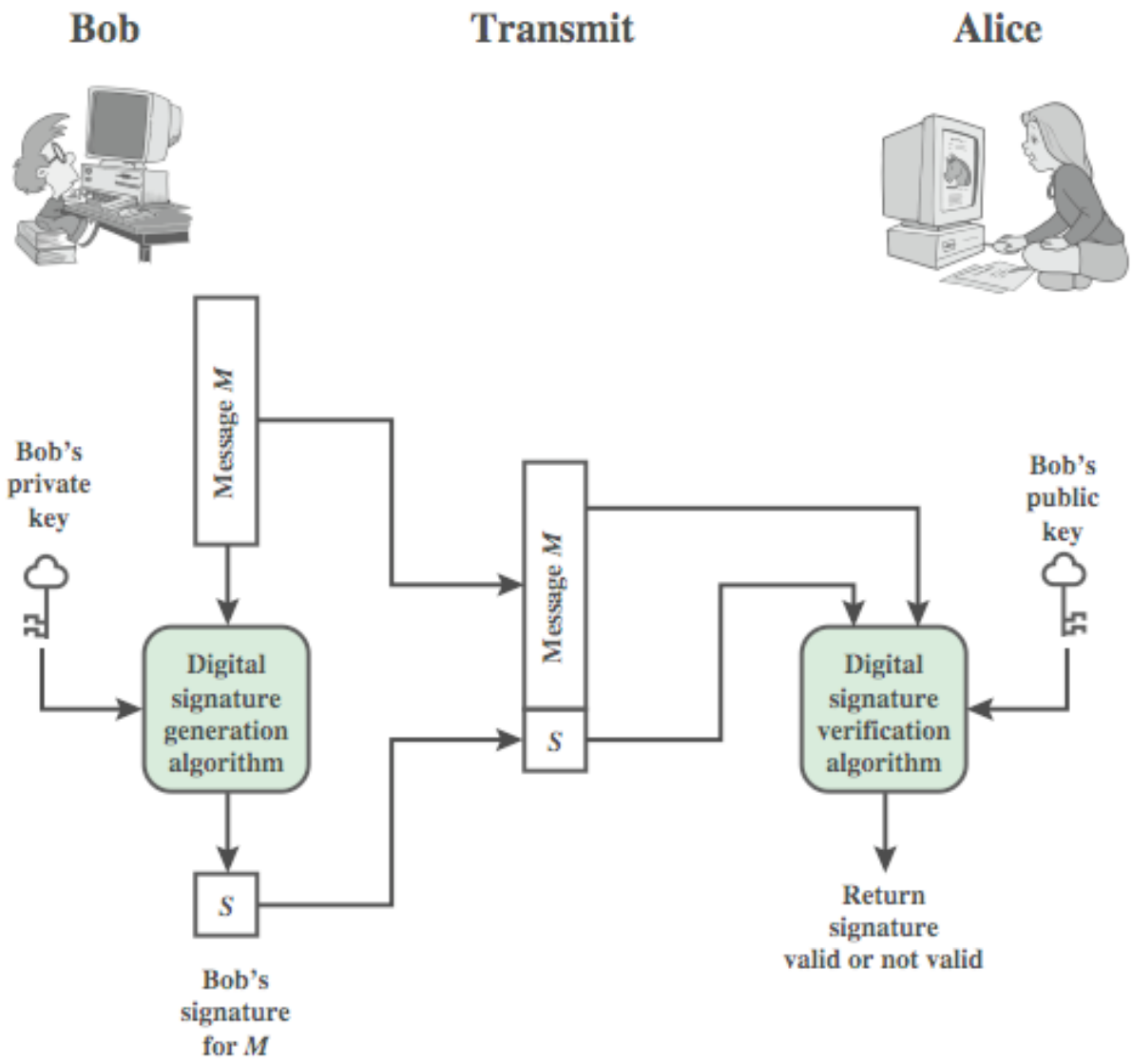
Digital signatures provide the ability to:

1. Verify author, date and time of signature.
2. Authenticate message contents
3. Verified by the third parties to resolve disputes.

Hence the digital signature provides authentication in addition to confidentiality and non-repudiation provided by asymmetric key cryptography of RSA.

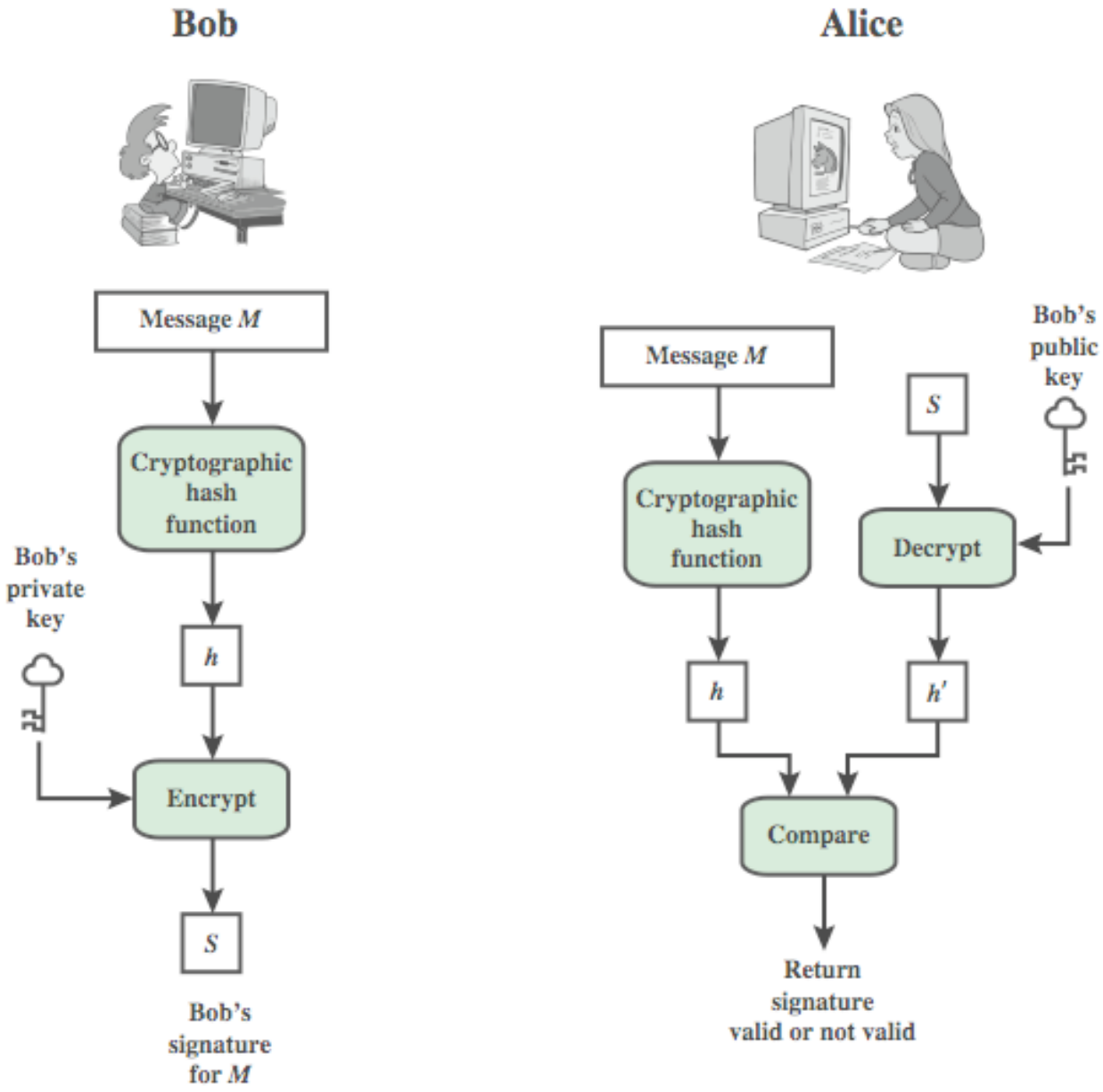
The digital signature is generated by encryption of hash of the message using the private key of the server using RSA algorithm.

DIGITAL SIGNATURE MODEL



There are two types of digital signature usages.

1. Signing a message w by a user A so that any user can verify the signature . i.e $d_A(w)$.
2. Signing a message w by a user A so that only user B can verify the signature. i.e $e_B(d_A(w))$.
3. Sending a message w and a signed message digest of w obtained by using a hash function standard h. *i.e* $(w, d_A(h(w)))$.



AUTHENTICATION

This solution permits only authenticated users to gain access to the received data by client verifying its signature at the client's end. The verification is done by decryption of message digest at the client with the public key that is in pair with the private key of the server. This mechanism ensures authentication.

NON-REPUDIATION

The user at the client end want to ensure that these details are from a genuine server and later server cannot deny about these details. This part of non repudiation is achieved by using RSA algorithm. Server signs the message containing details of DL and sends to client. At the client side, this signature is verified using server's RSA public key.

INTEGRITY

All the communication in the system is having integrity code in each message .This integrity code is generated involving hash of shared secret of the user , therefore any tampering of data can be detected at both the ends.

CONFIDENTIALITY

The server encrypts the message with RSA using its private key and the client decrypts the message using its public key available from the key distribution.

STEPS TO RUN CODE:

1. Open Xampp server and start running it.
2. Open web browser.
3. Type the following url: http://localhost/NSC/gmt.php
4. Now we can view the link to fetch GMT date and time.
5. Click the link and then download the document and save it at your desired location.
6. The "message.txt" downloaded contains the GMT details and its signature that needs is verified at the client side.