# Assignment 3

## Question 1

1. Variables and stored

|  |  |
| --- | --- |
| variables | stored |
| greetingMsg | Heap segment |
| Name | stack |
| Greeting[] | stack |
| Len | Stack |
| i | Stack |

1. Bugs
2. Line 16: use strlen instead of sizeof. Otherwise, will cause a warning. What we need here is the size of the string length, not the storage space occupied by the variable.
3. Line 20: if the length of argv longer than 256, it will cause the buffer overflow. More safety way using strncpy (name, argv[i], sizeof(name))
4. Line 22,23,26: has the same problem with the line 20, if the argc bigger than 256 will cause the buffer overflow.

## Question 2

First of all, Alice needs to perform a digital signature on the sent information, and the digital signature should be performed using third-party authentication software. Then, the message with the digital signature is encrypted with Alice's private key. Then send the encrypted message to Bob. Bob uses Alice's public key to decrypt, and then passes third-party authentication software to verify that the digital signature belongs to Alice. After verification, confirm that the message was sent from Alice.

## Question 3

1. PKI (Public Key Infrastructure) is a standard-compliant technology and specification that uses public key encryption technology to provide a secure basic platform for e-commerce. It can provide cryptographic services such as encryption and digital signatures and the necessary key and certificate management system for all network applications. In short, PKI is an infrastructure that uses public key theory and technology to provide security services. Internet users can use the services provided by the PKI platform to conduct secure electronic transactions, communications, and various activities on the Internet.
2. Digital signature: is a mathematical technique used to validate the authenticity and integrity of a message, software or digital document.

Digital certificate is issued by a trusted third party which proves sender’s identity to the receiver and receiver’s identity to the sender.

Digital signature is used to verify authenticity, integrity, non-repudiation. it is assuring that the message is sent by the known user and not modified, while digital certificate is used to verify the identity of the user, maybe sender or receiver. Thus, digital signature and certificate are different kind of things but both are used for security.

1. HTTPS: it is a transmission protocol for secure communication through computer networks. HTTPS communicates via HTTP but uses SSL/TLS to encrypt data packets. The main purpose of HTTPS development is to provide identity authentication for website servers and protect the privacy and integrity of exchanged data.

Digital signature work: When a signer electronically signs a document, the signature is created using the signer’s private key, which is always securely kept by the signer. The mathematical algorithm acts like a cipher, creating data matching the signed document, called a hash, and encrypting that data. The resulting encrypted data is the digital signature. The signature is also marked with the time that the document was signed. If the document changes after signing, the digital signature is invalidated.

Digital certificate work: First, the sender hashes the data package using a hashing algorithm. After hashing comes encryption. Then sender generates the keys upon initial transmission. Before transferring data, the sender encrypts the hashed data using the private key. The resulting output is the digital signature. The digital signature, along with the public key, is then appended to the original, unmodified data and sent to the recipient. The receiver, upon receiving the data pack, decrypts the signature using the attached public key to reveal the message digest. If the signature can be decrypted using the public key, then the recipient can be sure that the data came from the expected sender. The receiver then performs the same hashing function on the unmodified data to generate their own data-digest. If the resulting digest matches the one decrypted, then it confirms to the receiver that the data hasn’t been tampered with.

1. Issues in HTTPS

The security of the HTTPS ecosystem is ultimately dependent on the set of CAs that are entrusted to sign browser-trusted certificates. Except in a small handful of cases, any organization with control of a signing certificate that chains to a browser-trusted root can sign a leaf certificate for any domain. As such, the entire ecosystem is as fragile as the weakest CA. However, because there is no central, public registry of browser-trusted intermediate authorities, the organizations that control these signing certificates may be unknown until certificates they have signed are spotted in the wild.

* Ignoring foundational principles
* Standards and working groups
* Browsers to lead the way
* Failing to recognize cryptographic reality

1. Solution

HTTPS two-way authentication process:

1. The browser sends a connection request to the secure server.
2. The server sends its own certificate and information related to the certificate to the client browser.
3. The client browser checks whether the certificate sent by the server is issued by a trusted CA center. If yes, continue to execute the agreement; if not, the client browser will give the client a warning message: warn the client that this certificate is not trustworthy, and ask the client whether they need to continue.
4. Then the client browser compares the messages in the certificate, such as the domain name and public key, to see if they are consistent with the relevant message just sent by the server. If they are the same, the client browser recognizes the legal identity of the server.
5. The server requires the client to send the client's own certificate. After receiving it, the server verifies the client’s certificate, and if it fails the verification, the connection is refused; if the verification is passed, the server obtains the user’s public key.
6. The client browser tells the server which communication symmetrical password scheme it can support.
7. The server selects a password scheme with the highest encryption level from the password schemes sent by the client, and informs the browser after encrypting it with the public key of the client.
8. For this password scheme, the browser selects a call key, then encrypts it with the server’s public key and sends it to the server.
9. The server receives the message sent by the browser, decrypts it with its own private key, and obtains the call key.
10. The next communication between the server and the browser uses a symmetric cryptographic scheme, and the symmetric key is over-encrypted.

## Feedback

5-6hours for me to complete this homework. It is ok. And it helps me to learn the material of sending a encrypt message. Lecture is clear.

## References

Digital Signatures and Certificates Retrieved from: <https://www.geeksforgeeks.org/digital-signatures-certificates/>

How do digital signatures work? Retrieved from: <https://www.docusign.com/how-it-works/electronic-signature/digital-signature/digital-signature-faq>

What are digital certificates and how do they work? Retrieved from: <https://www.itworldcanada.com/article/what-are-digital-certificates-and-how-do-they-work/435269>