IS 364 - ASSIGNMENT 02

Name: **EMANUEL JOHN M.**

Reg No: **2017-04-07362**

BSc with Computer Science

Question 1;

Secure Sockets Layer (SSL) is a protocol for securing communication on the Internet. It provides a way for enterprises to encrypt data before sending it to users, preventing third parties from reading it while it is in transit. It is a networking protocol designed for securing connections between web clients and web servers over an insecure network, such as the internet. SSL uses a combination of public key encryption and private key encryption and other cryptographic functions to secure a connection between two machines, typically a web server or mail server and a client system, communicating over the internet or another TCP/IP network. SSL provides a mechanism for encrypting and authenticating data sent between processes running on a client and server, as well as mediating the secure exchange of private keys for session encryption through the use of SSL certificate issued by a trusted certificate authority.

HOW SSL PROTOCOL WORKS/FUNCTIONS;

The SSL protocol includes two sub-protocols which are **record protocol** and **the handshake protocol**. The handshake protocol defines how a web client and web server establish an SSL connection, including the negotiation of which cryptographic systems each host is willing or unwilling to use for communication. The record protocol defines how communicating hosts exchange data using SSL, including specifications for how data is to be prepared for transmission and how it is to be verified or decrypted on receipt.

As part of the initial handshake process, a server presents its SSL certificate to authenticate itself to the client. Server certificates follow the X.509 certificate format defined by the Public Key Cryptography Standards (PKCS). The authentication process uses public key encryption to validate the digital certificate and to confirm that a server is, in fact, the server it claims to be. SSL certificates, like any digital certificate, should be issued by a trusted certificate authority. Once the server has

been authenticated, the client and server establish cipher settings and a shared key to encrypt the information they exchange during the remainder of the session. This provides data confidentiality and integrity. This whole process is invisible to the user. For example, if a webpage requires an SSL connection, the URL will change from HTTP to HTTP Secure (HTTPS), and a padlock icon will appear in the browser once the server has been authenticated. The handshake also allows the client to authenticate itself to the server. In this case, after server authentication is complete, the client must present its certificate to the server to authenticate the client's identity before the encrypted SSL session can be established.

Alternatively,

SSL works through the use of public key cryptography. Public key cryptography uses two keys a private key and a public key to transmit secure data between two systems. These keys are essential to respectively decoding and encoding secure data.

Step-by-step, how SSL works:

1. A user connects to an SSL-enabled service such as a website.
2. The user’s application requests the server’s public key in exchange for its own public key. This public key exchange provides ways for both parties to encrypt messages that only the other party can read.
3. When the user sends a message to the server, the application uses the server’s public key to encrypt the message.
4. The server receives the user’s message and decrypts it using its private key. Messages sent back to the browser are encrypted in a similar way using a public key generated by the user’s application.

Public key cryptography is similar to using a padlock. The padlock itself is the public key and the combination is the private key. The server distributes its padlock, which anyone can use to lock a door or a box. However, the padlock can’t be opened without the combination, which only the server knows.

SECURITY IMPROVEMENTS MADE FROM SSL1 TO SSL3;

- SSL 3.0 can defend against “man-in-the-middle” attack by keeping the authenticated finished message with including a hash for all the previous handshake messages.
- SSL 3.0 uses HMAC, which is more powerful than MAC. It uses 128-bit of encryption. Attacker cannot alter the record or information even sending on open private network. It also provides key message authentication.
- SSL 3.0 enables Client can interrupt in the middle of handshake and can change the algorithm and keys whenever he wants.
- SSL 3.0 has a general key exchange protocol. It permits the Diffie-Hellman and Fortezza key exchanges and non-RSA certificates.
- SSL 3.0 permits chain certificates for the client and server. It provides certificate hierarchy.
- SSL 3.0 uses SHA-1 hashing algorithm, which is more secure than MD5 algorithm. It supports extra cipher suites. It also uses BSAFE 3.0 that includes a fixing of many attacks and the SHA-1 algorithm.

Question 2;

HOW DOES TLS PROTOCOL WORKS/FUNCTIONS;

TLS uses a combination of symmetric and asymmetric cryptography, as this provides a good compromise between performance and security when transmitting data securely.

With symmetric cryptography, data is encrypted and decrypted with a secret key known to both sender and recipient; typically128 but preferably 256 bits in length (anything less than 80 bits is now considered insecure). Symmetric cryptography is efficient in terms of computation, but having a common secret key means it needs to be shared in a secure manner.

With asymmetric cryptography uses key pairs, a public key and a private key. The public key is mathematically related to the private key, but given sufficient key length, it is computationally impractical to derive the private key from the public key. This allows the public key of the

recipient to be used by the sender to encrypt the data they wish to send to them, but that data can only be decrypted with the private key of the recipient.

For this reason, TLS uses asymmetric cryptography for securely generating and exchanging a session key. The session key is then used for encrypting the data transmitted by one party, and for decrypting the data received at the other end. Once the session is over, the session key is discarded.

Step-by-step, how TLS works:

TLS can be used on top of a transport-layer security protocol like TCP. There are three main components to TLS: Encryption, Authentication, and Integrity.

- **Encryption:** hides the data being transferred from third parties.
- **Authentication:** ensures that the parties exchanging information are who they claim to be.
- **Integrity:** verifies that the data has not been forged or tampered with.

A TLS connection is initiated using a sequence known as the TLS handshake. The TLS handshake establishes a cypher suite for each communication session. The cypher suite is a set of algorithms that specifies details such as which shared encryption keys, or session keys, will be used for that particular session. TLS is able to set the matching session keys over an unencrypted channel thanks to a technology known as public key cryptography.

The handshake also handles authentication, which usually consists of the server proving its identity to the client. This is done using public keys. Public keys are encryption keys that use one-way encryption, meaning that anyone can unscramble data encrypted with the private key to ensure its authenticity, but only the original sender can encrypt data with the private key.

Once data is encrypted and authenticated, it is then signed with a message authentication code (MAC). The recipient can then verify the MAC to ensure the integrity of the data. This is kind of like the tamper-proof foil found on a bottle of aspirin; the consumer knows no one has tampered with their medicine because the foil is intact when they purchase it.

SECURITY IMPROVEMENTS MADE FROM TLS1 to TLS3;

- TLS 1.3 offers improved speed. The faster speed for encrypted connections stems from features such as Zero Round Trip Time (0-RTT) and TLS false start. TLS 1.3 only needs to complete one round-trip. This reduces encryption latency by one-half. With this feature, users will be able to browse websites faster and with greater security.

- **Enhanced Security**
  Previously webmasters and system administrators struggled to consistently configure properly and thus made connections to websites vulnerable to attacks such as the RC4 and BEAST exploits, TLS 1.3 has removed the deprecated features that caused these issues, including SHA-1, RC4, DES and AES-CBC, among others. With this streamlined approach, Web developers and administrators are now less susceptible to misconfiguring protocols, thus making websites safer for users in terms of confidentiality and integrity as well as reducing the risk of cyberattacks.

Question 3;

HOW THE PKI WORKS:

PKI (or Public Key Infrastructure) is the framework of encryption and cybersecurity that protects communications between the server (your website) and the client (the users). It works by using two different cryptographic keys: a public key and a private key. The public key is available to any user that connects with the website. The private key is a unique key generated when a connection is made, and it is kept secret. When communicating, the client uses the public key to encrypt and decrypt, and the server uses the private key. This protects the user's information from theft or tampering.

PKI works for both **Authentication** as well as **Encryption**. And this is as follows;

During Authentication;

A Public Key Infrastructure requires several different elements for effective use. A Certificate Authority (CA) is used to authenticate the digital identities of the users, which can range from individuals to computer systems to servers. Certificate Authorities prevent falsified entities and manage the life cycle of any given number of digital certificates within the system.

Second in command is the component of a Registration Authority (RA), which is authorized by the Certificate Authority to provide digital certificates to users on a case-by-case basis. All of the certificates that are requested, received, and revoked by both the Certificate Authority and the Registration Authority are stored in an encrypted certificate database.

Certificate history and information is also kept on what is called a certificate store, which is usually grounded on a specific computer and acts as a storage space for all memory relevant to the certificate history, including issued certificates and private encryption keys. Google Wallet is a great example of this.

By hosting these elements on a secure framework, a Public Key Infrastructure can protect the identities involved as well as the private information used in situations where digital security is necessary, such as smart card logins, SSL signatures, encrypted documents, and more.

During Encryption;

PKI uses the keys that it generates for encryption. Whether these keys are public or private, they encrypt and decrypt secure data. PKI merges the use of both asymmetric and symmetric encryption. Symmetrical encryption protects the single private key that is generated upon the initial exchange between parties the digital handshake. This secret key must be passed from one party to another in order for all parties involved to decrypt the information that was exchanged.

Asymmetric encryption is fairly new to scope and one may know it better as “public key cryptography.” Asymmetric encryption uses two keys to encrypt plain text, both a public key and a secret key.

PKI ENHANCEMENTS/IMPROVEMENTS MADE TO DATE:

- PKI has fixed algorithms which makes things cheaper for organizations.
- PKI is often used by federal organisations. These organisations often consider PKI to be strategic to their security goals. Defence, health and banking also rely on PKI for authentication and authorisation.