CSCI 530 - Assignment-2

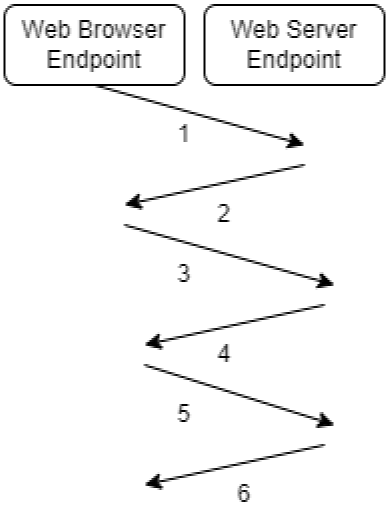
Rishit Saiya (rsaiya@usc.edu)

To ensure secure connection between two communication endpoints, the IPSec protocol suite is utilized. Lower tiers of the protocol stack use it. IPSec uses a variety of protocols and security measures to enable authentication, confidentiality, and integrity. In IPSec, authentication takes place on a computer or system rather than a person. In the described scenario, authentication will take place on computers that have web servers and browsers installed. In the course of authentication, the origin of IP packets is confirmed. Hashing and cryptography secure the integrity of packets, and securing the payload ensures confidentiality.

If IPSec is required between a web browser and a web server to preserve confidentiality and integrity, the appropriate mechanism must be taken into consideration because capability varies from mechanism to mechanism. Depending on the situation, IPSec offers two different types of cryptographic services. In contrast to SSL/TLS, which implements secrecy and integrity through a single structured procedure, IPSec offers two choices in the form of ESP and AH. Data integrity and source authentication are provided by the Authentication Header (AH). Encapsulated Security Payload (ESP) offers confidentiality in addition to the services provided by AH, although it cannot be utilized for confidentiality. IKE (Internet Key Exchange), which is used for key management, is one of the essential parts of IPSec. It facilitates the exchange of these keys and cryptographic algorithms between two endpoints (in this case, on the endpoints where the web browser and web server will be installed). IKE helps two communicating parties form security connections as well. IKE must first negotiate and establish an end to end secure channel before securing data between endpoints (on one endpoint, a web browser will be installed, and on the other endpoint, a web server will be deployed). A session is essentially a stream that is used between two endpoints and is known as a security association. IKE is used to establish sessions. In essence, a security association is a group of properties connected to the connection. The Security association Database will provide information about security associations (SADB). The security association in the example will be between the web browser and the web server. The way the specific associations are negotiated can vary depending on the session. IPSec operates on the network layer whereas the SSL/TLS operate upon the application layer which makes the IPSec usage difficult in practice as compared to SSL/TLS.

In order to establish an IPSec connection, the system undergoes 2 Phases:

Phase 1 is used to create a secure channel from end to end between two endpoints (systems with installed web servers and browsers). During this Phase, mutual authentication is done. In Phase 2 discussions, it is frequently utilized. Only once is this Phase established between two ends. Phase 1 can be carried out in 2 different ways: the main mode and the rapid mode. Main mode is employed in the majority of instances. In comparison to rapid mode, major mode has more options and utilizes six messages across three round trips.



1: Its a request that consists of Header and the security association from the initiator/endpoint where the web browser is installed to web server endpoint
2: Its a request that consists of Header and the security association from the responder/endpoint where the web server is installed to web browser endpoint
3: Its a request that consists of Header, Key exchanged value and Nonce of Initiation endpoint with the attached certificate
4: Its a request that consists of Header, Key exchanged value and Nonce of responder endpoint with the attached certificate
5: Its a request that consists of Header, the initiator, the certificate and the signature
6: Its a request that consists of Header, the responder, the certificate and the signature

Once Phase 1 gets completed, we enter into Phase 2. Phase 2 uses the fast mode to create unique secure routes between two endpoints (in this case, web client and web server). Web browsers and servers can communicate with one another once Phase 2 is complete using a mutually agreed-upon encryption suite to protect the confidentiality and integrity of data.

The management of the data going to the Web server and the management of the data coming from the Web server will each be handled by a separate association. Similar to this, there will be two associations, one for managing data entering the web browser and the other for managing data leaving the web browser, and both affiliations are likely to have distinct encryption keys. The operating system's memory of the endpoint, where web browsers and web servers are installed, contains a database of security associations. So, all the keys will be stored in a database that will be on your local machine.

IPSec Particulars examples are as follows:
Hash Algorithm: MD5, SHA1
Authentication method: Preshared key (PSK), Public key, Digital Signature
Key Exchange method: Diffie Hellman
Symmetric key algorithm: AES, 3DES

SSL/TLS:

TLS (Transport Layer Security) and SSL (Secure Sockets Layer) are technologies for creating secure connections between various systems. Only websites that have an SSL certificate are able to use SSL. SSL certificates are issued by certificate authorities (CA). Web browsers and web servers engage in an initial handshake authentication process to make sure they are who they say they are. Authentication by the client is optional. The following actions will be carried out jointly by the client and server during an SSL/TLS handshake:

Web browser and web server will Specify which version of TLS they will use. Then they will decide on which cipher suites they want to use. Client (web browser) will authenticate the identity of the web server via the web server's public key and the SSL certificate authority's digital signature. Post that session keys will be generated in order to use symmetric encryption after the handshake is complete.

There are differences between the handshake process that takes place in SSL and IPSec. In SSL process is as below:
The handshake is started by the web browser by sending the server a hello message, client random, and supported cipher suite. The web server replies to the client's greeting message by sending a message that includes the server's SSL certificate, its selected cipher suite, and its "server random." The SSL certificate of the server will be checked by the web browser with the certificate authority that issued it. The browser will then send the premaster secret after that. The web server's public key will be used to encrypt the premaster secret, which can only be decrypted by the server using the private key. With the use of its private key, the server will decrypt the premaster secret. They will come to the same conclusion. A finished message that has been encrypted with a session key will be sent by web browsers. The finished message will be sent by the web server and will be session key encrypted. The handshake is now complete, and the session keys will be used for any subsequent communication. Web servers and browsers can now connect in a secure setting that safeguards data integrity and confidentiality. With a digital signature certificate, cipher-suite (RSA, DH, HMAC-SHA1, Symmetric Key Algorithm: AES) is utilized for confidentiality, integrity, and authenticity.

Therefore, IPSec must be a two-Phase protocol. While SSL follows a single organized procedure in which negotiation is completed only once, initiator and responder can start Phase 2 negotiation only after completing Phase 1. Additionally, client authentication (web browser) is optional with SSL but required with IPSec for both parties engaged in communication. There are two methods for end point authentication with IPSec connections. Preshared keys and another certificate make up one. Preshared keys pose significant security risks. An attacker may crack or seize the pre-shared key if the key exchange is not done securely. Unauthorized access could be gained by attackers if a preshared key is compromised. The preshared key has a modest size and is quite easy to crack. Furthermore, updating the key on all systems (in this example both computers) is a challenging operation even if we learn that the key has been compromised. Since public key cryptography is always used to establish a handshake and then securely exchange encryption keys, SSL does not have this issue.