Problem 1.

The IPSec architecture documents states that when two transport mode SA are bundled to allow both AH and ESP protocols on the same end-to-end flow, only one ordering of security protocols seems appropriate: performing the ESP protocol before performing the AH protocol. Why is this approach recommended rather than authentication before encryption?

Solution

This order of processing facilitates rapid detection and rejection of replayed or bogus packets by the receiver, prior to decrypting the packet, hence potentially reducing the impact of denial of service attacks. It also allows for the possibility of parallel processing of packets at the receiver, i.e., decryption can take place in parallel with authentication.

Problem 2.

Consider the following threats to web security and describe how each is countered by a particular feature of IPSec.

a. Brute force cryptoanalitic attack. An exhaustive search of the key space for a conventional encryption algorithm.

b. Known plaintext dictionary. Many messages will have predictable plaintext such as HTTP DET command. An attacker can construct a dictionary containing every possible encryption of the known plaintext message. When an encrypted message is intercepted, the attacker takes the portion containing the encrypted known plaintext and looks up the ciphertext in the dictionary. The ciphertext should match against an entry that was encrypted with the same secret key. This attack is especially effective against small key sizes (e.g. 40-bit keys).

c. Replay attack. Earlier SSL handshake messages are replayed.

d. Man in the middle. An attacker interposes during key exchange, acting as the client to the server and a server to the client.

e. Password sniffing. Passwords in HTTP or other application traffic are eavesdropped.

f. IP spoofing. Uses forged IP addresses to fool a host into accepting bogus data.

g. IP hijacking. An active, authenticated connection between two hosts is disrupted and the attacker takes the place of one of the hosts.

h. SYN flooding. An attacker sends TCP SYN messages to request a connection but does not respond to the final message to establish the connection fully. The attacker TCP module typically leaves “half-open connections” around for a few minutes. Repeated SYN messages can clog the TCP module.

Solution

**a. Brute Force Cryptanalytic Attack:** The conventional encryption algorithms use key lengths ranging from 40 to 168 bits.
b. **Known Plaintext Dictionary Attack:** SSL protects against this attack by not really using a 40-bit key, but an effective key of 128 bits. The rest of the key is constructed from data that is disclosed in the Hello messages. As a result the dictionary must be long enough to accommodate $2^{128}$ entries.

c. **Replay Attack:** This is prevented by the use of nonces..

d. **Man-in-the-Middle Attack:** This is prevented by the use of public-key certificates to authenticate the correspondents.

e. **Password Sniffing:** User data is encrypted.

f. **IP Spoofing:** The spoofer must be in possession of the secret key as well as the forged IP address..

g. **IP Hijacking:** Again, encryption protects against this attack..

h. **SYN Flooding:** SSL provides no protection against this attack.

**Problem 16.2 from the class book**

**Solution**

If Bob wants a cookie, have Bob reply to a message without a cookie with a “try again, this time returning this “cookie”.

**Problem 16.9 from class book**

**Solution**

Use the key in the Kerberos ticket to do a shared secret mutual authentication PFS protocol. For instance, do a Diffie-Hellman exchange, and have the session key be a function of the key in the ticket and the Diffie-Hellman key.

**Problem 16.10 from class book**

**Solution**

By having the cookie be a function of the time and including the time in the cookie, Bob can enforce a time limit on how long he will accept a cookie. If he remembers all successfully used nonces until they would be rejected as expired, he can detect and reject all replay attempts (either as replay or as expired).

**Problem 17.4 from class book**

**Solution**

Yes to both questions. Since it is the receiver that defines the SPI, it can assign different SPIs to ESP SAs vs. AH SAs. This will interwork with implementations that allow the same SPI to be assigned to both, and distinguish which SA it belongs to based on whether it is AH or ESP.