Perfect Forward Secrecy (PFS) Overview

Difference between no PFS (with RSA) and PFS (with DH) in TLS/SSL/IPsec Connections

Key Exchange via RSA (no PFS)

Client

The client generates a session key, encrypts it via RSA, and sends it to the server.

Server

- Generates session key K_Sess
- (RSA Encryption) Encrypts K_Sess with the public long term key from the server K_Pub and sends it to server

Encrypted K_Sess

- (RSA Decryption) Decrypts K_Sess with its private key K_Priv

Communication encrypted with symmetric cipher using K_Sess

If the third party has access to K_Priv some day, it can subsequently decrypt all communication since it can reproduce all session keys.

Complete communication is stored by a third party

Evil Third Party

Key Agreement via DH (with PFS)

Client

Client and server deliver input to derive the session key. The session key itself is not transmitted through the network.

Server

- (Diffie-Hellman) Generates random value a and computes A
- Sends A to the server
- **Computes K_Sess from input of itself (a) and the server (B).**

A

B

- (Diffie-Hellman) Generates random value b and computes B
- Sends B to the client
- **Computes K_Sess from input of itself (b) and the client (A).**

Communication encrypted with symmetric cipher using K_Sess

Since K_Sess is freshly generated for each session, not transmitted on the net, and not encrypted with a long term key, a third party cannot decrypt the communication unless it breaks every single session key.

Complete communication is stored by a third party

Evil Third Party

by Johannes Weber | http://blog.webernetz.net | Last Change: 2014-07-18