Segurança de Redes e Sistemas de Computadores 2019/2020
1º Semestre Aula Prática (LAB 5)

Guidelines

Objetive: In this Lab we can observe how to program cryptographic constructions for the Diffie-Hellman Key-Agreement Protocols

Hands-On Activities

1) **DH.java**

   Experimental analysis on the generation of DH Public and Private Numbers and observation on the computational cost of the generation process when we want Numbers of different keysizes, ex: 512, 1024, 2048, 4096, ….

   Do you observe differences when using different cryptographic providers? Can you explain the performance differences ?

   What can we conclude from the example?

2) **TwoWayDHExample.java and TwoWayDHEExample2.java**

   In the provided code (demo) we show how to program a Key-Agreement between two principals only exchanging the DH public numbers.

   You must notice that in this case we decide to have fixed initial parameters (shared between the two involved principals): G (primitive root) and P (A prime Number to be used in the DH Modular exponentiations).

   As you can observe, the agreed (established) secret can be easily used as a seed to generate for example a symmetric keys, HMAC keys or CMAC keys, that can be used for subsequent secrecy information, in established secure channels between the principals.

   See that this example doesn't show an authenticated key agreement but only an Anonymous Agreement. What are the dangers in Anonymous DH Exchanges when the two involved principals are attacked by a Man in the Middle ?
   How can we protect from this?

   Explain the differences between **TwoWayDHExample.java and TwoWayDHEExample2.java** and the observed results.

3) **ThreeWayDHEExample**

   This example shows how we can create generalized Diffie-Hellman agreements extended to N principals. As you see, by only sharing public numbers generated by each participant in a group of principals (or processes) that will never expose the respective private numbers, it is possible to establish a common shared group key.

4) openssl-dh-agreement

This is a simple demo showing how we can use the openssl tool to generate keys (or secrecy material) performing a Diffie Hellman Agreement.

6) KEYTOOLS-KEYSTORES

Here you can find different demos, for the manipulation of Java keystores when using asymmetric cryptography and related keypairs. You have examples on the use of the Keytool Java tool to generate <private,public keys> and how to store and manage such keys in keystores (for storage of the keypairs) and how to export the public keys to trusted stores or to generate public key certificates.