Exercises UD06

1. Training

1. **Exercise 1: Integrity check.**

   Program an application that receives a file and its SHA-512 HASH as input parameters and that certifies the integrity of the file. You can get test files and HASH from the Hak5 downloads section: https://downloads.hak5.org/

2. **Exercise 2: Salted hashes**

   Investigate what salted Hashes are and why it is necessary to add "salt" to them. to our hashes. Investigate and explain how the GNU/Linux operating system stores the salted hash in the `/etc/shadow` file. Modify **`Example2`** so that it incorporates "salt" to the hashes. You must add salt when registering a credential and take it into account when validating the password.

3. **Exercise 3: Secure storage.**
   Write a Java application that takes a file and a password as input parameters, encrypts the content using AES, stores the result under a different filename, and deletes the original file. Also program the application that performs the inverse transformation.

4. **Exercise 4: Confidentiality and identity.**

   Program a Java application that, given a file, encrypts it first with the private key of the sender and, later, with the public key of the recipient. Also program the application that decrypts the file.

5. **Exercise 5: Digital signing of documents.**
   Design and develop a system in Java that requests a file and generates its digital signature. Create the program capable of validating that the signed document has not been altered and that the issuer is who he claims to be.

6. **Exercise 6: Secure socket**

   Making pairs or trios, prepare the environment of **`Example6`**, so that one student performs the server tasks, and another or others as the client (not simultaneously). Generate all the necessary keys/certificates/stores and when the system is working show it to the teacher for evaluation.

2. Questions

1. What is the identifier-password pair of a security system called?
   a) Profile.
   b) Credential.
   e) Authorization.
   d) Role.
2. Which of the following physical characteristics cannot be a biometric access control?
   a) The voice.
   b) The fingerprint.
   c) Facial recognition.
   d) The height.
3. Why should password digests be stored instead of passwords?
   a) To save space.
   b) To prevent obtaining the password.
   c) To more quickly validate credentials.
   d) To prevent two users from having the same password.
4. What part of a cryptographic system must be secret?
   a) The algorithm used to encrypt.
   b) The length of the passwords.
   c) Passwords.
   d) User identifiers.
5. Which of the following characteristics is not characteristic of a symmetric key cryptographic system?
   a) Es reversible.
   b) The output generated by the algorithm has a constant size.
   c) You cannot lose information.
   d) The key is unique and valid for both encryption and decryption.
6. Which of the following algorithms is not a HASH algorithm?
   a) MD5.
   b) SHA-2.
   e) SHA-3.
   d) DES.
7. Which of the following algorithms is a public key algorithm?
   a) AES.
   b) RC5.
   c) Blowfish.
   d) RSA.
8. Which of the following algorithms is used to perform the digital signature?
   a) RSA.
   b) DSA.
   e) AES.
   d) SHA-2.
9. Which of the following algorithms is used to store encrypted information?
   a) Blowfish.
   b) AES.
   e) RSA.
   d) DES.

10. What is guaranteed with the digital signature?
    a) The integrity of the message.
    b) The identity of the recipient.
    c) Confidentiality.
    d) The security of the communication channel.
11. What is the weak point of using a symmetric key algorithm in data transfer?
    a) The channel.
    b) The distribution of passwords.
    c) The intrinsic security of the algorithm.
    d) The existence of collisions.

3. Application Activities

1. Some cryptographic algorithms with the passage of time have ceased to be secure. State the main reasons.
2. Defines what credential-based access control consists of.
3. Describes the process of storing credentials using hashes (HASH).
4. Explains the credential verification procedure using hashes (HASH).
5. Indicate why hashes (HASH) should be used to store passwords instead of using symmetric algorithms.
6. Explain how a HASH algorithm can be used to guarantee the integrity of a message.
7. HASH algorithms can have collisions. Explain what they are and why it is accepted as valid that there is a remote possibility that they occur.
8. Explain what the digital signature of a document consists of.
9. Describes the inherent risk of using non-secure protocols (TCP or HTTP sockets) in the transfer of information.
10. Explain why it is convenient to change passwords from time to time.
11. Make a presentation on the reasons to avoid using websites that are capable of displaying passwords in the clear.
12. Explain the process by which the identity of the sender and the confidentiality of a message can be guaranteed using a public key algorithm.
13. Find at least three recommendations on how to create a strong password and make your own recommendation.

4. Expansion activities

1. Deepen your knowledge of the Caesar cipher. Find out how to crack a ciphertext with this system.
2. Explain the implication of the symmetric and asymmetric key algorithms in VPN connections. Are both used? only one? which is better? It explains how to establish the connection and the subsequent communication in a secure way.
3. Look for websites that generate HASH. It checks that the summaries that all of them generate are the same given the same inputs.
4. Find out how stream ciphers differ from block ciphers.
5. Investigate the use of the digital signature. Discover what it can be used for in the Public Administration and what online tools exist to verify that a document is correctly signed.
6. Find out how the following AES algorithm versions differ: AES-CBC, AES-CFB, AES-OFB.
7. Study how the HTTPS protocol works.
8. Find out what functionality the Java Authentication and Authorization Service (JAAS) service provides.
9. Find out how useful the Java Secure Socket Extension (JSSE) is.

5. Information sources

- Wikipedia
- Programación de servicios y procesos - FERNANDO PANIAGUA MARTÍN [Paraninfo]
- Programación de Servicios y Procesos - ALBERTO SÁNCHEZ CAMPOS [Ra-ma]
- Programación de Servicios y Procesos - Mª JESÚS RAMOS MARTÍN - [Garceta] (1ª y 2ª Edición)
- Programación de servicios y procesos - CARLOS ALBERTO CORTIJO BON [Sintesis]
- Programació de serveis i processos - JOAR ARNEDO MORENO, JOSEP CAÑELLAS BORNAS i JOSÉ ANTONIO LEO MEGÍAS [IOC]
- GitHub repositories:
  - https://github.com/ajcpro/psp
  - https://oscarmaestre.github.io/servicios/index.html
  - https://github.com/juanro49/DAM/tree/master/DAM2/PSP
  - https://github.com/pablohs1986/dam_psp2021
  - https://github.com/Perju/DAM
  - https://github.com/eldiegoch/DAM
  - https://github.com/eldiegoch/2dam-psp-public
  - https://github.com/franlu/DAM-PSP
  - https://github.com/ProgProcesosYServicios
  - https://github.com/joseluisgs
  - https://github.com/oscarnovillo/dam2_2122
  - https://github.com/PacoPortillo/DAM_PSP_Tarea02_La-Cena-de-los-Filosofos