

Práctica 1 (PARTE 1) Protección / Ocultamiento de información

Términos relacionados:

- Codificación, encoding, criptografía, cifrado clásico, cifrado por sustitución, cifrado por transposición, cifrado moderno, cifrado simétrico, cifrado asimétrico, RSA, firma digital, cifrado por bloques, cifrado de flujo, esteganografía, LSB

Referencias:

- Dr. Jorge Ramió Aguirre. (2006). Libro Electrónico de Seguridad Informática y Criptografía Versión 4.1
- Dan Boneh and Victor Shoup. (2017). A Graduate Course in Applied Cryptography. Sitio web: <http://toc.cryptobook.us/>
- CrackStation. Salted Password Hashing - Doing it Right. de Sitio web: <https://crackstation.net/hashing-security.htm>

Tools:

- <http://rumkin.com/tools/cipher/>
- <https://hashcat.net/hashcat/>
- <http://www.openwall.com/john/>
- <https://www.dcode.fr/>
- <https://gchq.github.io/CyberChef/>
- <http://inventwithpython.com/cracking/>

Mas información:

- Libro crypto 101 - Laurens Van Houtven - <https://www.crypto101.io/>
- De la cifra clásica al cifrado RSA:
http://www.criptored.upm.es/guienteoria/gt_m001a.htm
- Libro fundamentos de seguridad en redes 2da edición - Stallings

Codificación (encoding)

La codificación de caracteres es un método que permite convertir un carácter de un sistema de representación a otro. El proceso de codificación no requiere ninguna clave, y la conversión se realiza aplicando un conjunto de normas o reglas. Estas reglas pueden ser utilizadas para revertir lo realizado y restaurar el mensaje en la representación original. El objetivo no es mantener la información en secreto, sino asegurarse de que la misma pueda ser utilizada. Por ejemplo, el correo electrónico utiliza un mecanismo de codificación para el envío de los datos binarios que se adjuntan (fotos, pdfs, ejecutables, etc). Esto es necesario porque el correo electrónico originalmente se definió para intercambiar mensajes representados utilizando el código ASCII y en dicho sistema de representación no se pueden representar todos los valores posibles que puede tener un byte.

Ejemplos: ASCII, Unicode, Url Encoding, Base64

Ejercicio 01

Develar el mensaje que se intentó ocultar utilizando un sistema de codificación

- a. 73 67 123 99 97 108 101 110 116 97 110 100 111 95 109 111 116 111 114 101 115
125
 - b. SUN7M25jMGRIYXJfbjBfM3NfZW5jcjFwdDRyfQ==
 - c. 9-14-20-18-15-4-21-3-3-9-15-14 1 12-1 3-9-2-5-18-19-5-7-21-18-9-4-1-4
 - d. 05110006_08130308020418_0011050001041908021418
 - e. 49 43 7b 77 68 65 6e 5f 75 5f 64 69 45 5f 79 6f 75 5f 61 72 65 5f 35 37 30 30
35 7d
 - f. .- -... .-. .- -.. .- -.. .- -... .-. .-
-

Ejercicio 02

Resolver el reto alojado en el puerto **11002** del sitio ic.catedras.linti.unlp.edu.ar

Criptografía

Ciencia que hace uso de métodos y técnicas con el objeto principal de cifrar, y por tanto proteger, un mensaje o archivo por medio de un algoritmo, usando una o más claves. Esto da lugar a diferentes tipos de sistemas de cifrado, denominados **criptosistemas**, que nos permiten asegurar la confidencialidad o secreto del mensaje como así también su integridad. Dependiendo del sistema de cifrado, se podría llegar a garantizar otros atributos como la autenticidad del emisor, así como el no repudio (tanto del emisor como del receptor).

Texto en claro / plaintext: Es el mensaje original

Texto cifrado / ciphertext: Es el mensaje cifrado

Clasificación de los sistemas de cifrado

- Según su historia : Sistemas clásicos vs Sistemas modernos
- Según el tratamiento de la información a cifrar: Sistemas de cifrado en bloque vs Sistemas de cifrado en flujo.
- Según el tipo de clave utilizada: Sistemas de clave privada (o simétricos) vs Sistemas de clave pública (o asimétricos)

Sistemas de cifrado clásicos

El cifrado clásico es un tipo de cifrado que se utilizó históricamente pero que ahora, en su mayor parte, ha caído en desuso. A diferencia de los algoritmos criptográficos modernos, la mayoría de los cifrados clásicos se pueden computar y resolver prácticamente a mano. Sin embargo, también suelen ser muy simples de romper con la tecnología moderna. El término "sistemas de cifrado clásico" incluye los sistemas de cifrado simples, utilizados desde la época griega y romana, los sistemas de cifra renacentistas, la criptografía de la Segunda Guerra Mundial como la máquina Enigma y más. En contraste, la criptografía moderna se basa en nuevos algoritmos que corren sobre computadoras desarrolladas desde la década de 1970.

Los sistemas de cifrado clásicos se dividen en dos tipos: sistemas de cifrado por sustitución y sistemas de cifrado por transposición.

Sistemas de cifrado por sustitución

En un cifrado por sustitución, las letras (o grupos de letras) del texto en claro se reemplazan sistemáticamente a lo largo del mensaje por otras letras (o grupos de letras).

Un ejemplo bien conocido de un cifrado de sustitución es el cifrado César. Para cifrar un mensaje con el cifrado César, cada letra del mensaje se reemplaza por la letra tres posiciones más adelante en el alfabeto. Por lo tanto, A es reemplazado por D, B por E, C por F, etc. Finalmente, X, Y y Z son reemplazados por A, B y C respectivamente. Entonces, por ejemplo, "AL ATAQUE" cifra como "DO DWDTXH". César rotó el alfabeto 3 posiciones, pero cualquier número funciona.

Ejemplos de sistemas de cifrado por sustitución: cesar cipher, atbash cipher, vigenere cipher, autokey cipher, affine cipher, entre otros.

Sistemas de cifrado por transposición

En un cifrado de transposición, las letras mismas se mantienen sin cambios, pero su orden dentro del mensaje se codifica de acuerdo con un esquema bien definido. Muchos cifrados de transposición se realizan según un diseño geométrico. Un cifrado simple (y una vez más, fácil de descifrar) sería escribir cada palabra al revés. Por ejemplo, "Hola mi nombre es Alice". ahora sería "aloH im erbmon se ecilA".

En un cifrado de columnas, el mensaje original está dispuesto en un rectángulo, de izquierda a derecha y de arriba a abajo. Luego, se elige una clave y se usa para asignar un número a cada columna en el rectángulo para determinar el orden de reordenamiento. El número correspondiente a las letras en la clave está determinado por su lugar en el alfabeto, es decir, A es 1, B es 2, C es 3, etc. Por ejemplo, si la palabra clave es CAT y el mensaje es THE SKY IS BLUE , así es como organizarías tu mensaje:

C A T
T H E
S K Y
I S B
L U E

The sky is blue -> HKSUTSILEYBE

Luego, toma las letras en orden numérico y así es como transpondrías el mensaje. Primero toma la columna debajo de A, luego la columna debajo de C, luego la columna debajo de T, como resultado, su mensaje "The sky is blue" se ha convertido en:

Ejemplos de sistemas de cifrado por transposición: rail fence cipher, route cipher, columnar transposition, double transposition, entre otros.

Ejercicio 03

Revele los siguientes mensajes. Para cada uno, indique qué cifrado se utilizó y si es de transposición o sustitución. Puede utilizar el sitio <http://rumkin.com/tools/cipher/>

- }ratnelac_a_odnazepmE{CI
- FZ{BPQL PF NRB PB ZLJMIFZX}
Pista: Mensaje que data del año 60 a.c
- QK{IPWZI CV XWKW UIA}
Pista: Mensaje recibido de Roma
- pp epnwfus dvjipèym jx ln dtjcefv jfxrhw rq hkmmwjetfd wpvkla ij teznfxgymx t ceuced hgs knkielb féwcy ntwdaoos pwwva hfiekghvgz csf kacwe, wpctiif kejyd hg cqljeèrf, byp wg baf hfqw poexl. mq hzfslhv hg cqljeèvm rv yp jqkwrpd oi dyuaqyztmón flqrsmtcibwjlfévpkt. rlc jvhr! nh nqfx et tg{g1k3plz3_wzc3w} . arjyí czí!
Pista: passphrase:le chiffre indechiffable
- Descifra esta extraña palabra: CROITSFRIRACANIPSOOPN
Pista: 7 x 3
- Militar exfiltration

An unauthorized encoded message was sent this morning. This may be very dangerous. Based on previous SIGINT our cryptographers have been told that to read it "a rail fence is needed". Can you help us read the message?

TSaeile nh umrnrwl ev tnoebi laao

Ejercicio 04

Resolver el reto alojado en el puerto **11004** del sitio ic.catedras.linti.unlp.edu.ar

Funciones de Hash

Las funciones hash son funciones que toman una entrada de longitud indeterminada y producen un valor de longitud fija, también conocido como "digest o resumen".

Las funciones de hash tienen muchas aplicaciones. Se utilizan para construir estructuras de datos, algoritmos de cifrado/descifrado, algoritmos generadores de números pseudoaleatorios, en firma digital, para sumas de verificación, pruebas de integridad de contenidos, en herramientas de autenticación y control de acceso, etc.

Estas funciones de hash realmente solo garantizan una cosa: por dos entradas idénticas, producirán una salida idéntica. Si bien es deseable que dos entradas no puedan ser manipuladas para que el resultado sea el mismo hash, es importante destacar que no hay garantía de que dos salidas idénticas implique que las entradas fueron las mismas. Eso sería imposible: solo hay una cantidad finita de resultados o resúmenes, ya que son de tamaño fijo, pero hay una infinidad de entradas posibles. Una buena función hash también es rápida para calcular.

La función de hash h será segura si tiene las siguientes características:

- Unidireccionalidad: conocido un resumen $h(M)$, debe ser computacionalmente imposible encontrar M a partir de dicho resumen.
- Compresión: a partir de un mensaje de cualquier longitud, el resumen $h(M)$ debe tener una longitud fija.
- Facilidad de cálculo: debe ser fácil calcular $h(M)$ a partir de un mensaje M .
- Difusión: el resumen $h(M)$ debe ser una función compleja de todos los bits del mensaje M : si se modifica un solo bit del mensaje M , el hash $h(M)$ debería cambiar.

Ejemplos de funciones de hash: MD5, SHA-1, SHA-256, HAVAL, Whirlpool, Tiger.

Dado que uno de los usos más frecuentes es para almacenar contraseñas de usuario (en lugar de guardar la password, se guarda el hash y cuando el usuario se autentica se compara el hash almacenado con el hash de la password ingresada), hay sitios donde puede averiguar la password a partir de un hash. Lo que se hace es buscar en "lookup tables" que son bases de datos de hashes con las respectivas posibles passwords que dan dicho hash. Estos hashes que se permiten consultar, en dichos sitios han sido calculados con anterioridad.

Para contrarrestar esto, los sistemas calculan el hash de la combinación de la contraseña con el SALT [https://es.wikipedia.org/wiki/Salt_\(criptografía\)](https://es.wikipedia.org/wiki/Salt_(criptografía))

Otra forma de dar con la contraseña que generó un hash determinado es mediante fuerza bruta. En este tipo de ataque, se intenta calcular el hash correspondiente a las distintas posibles contraseñas hasta que se encuentra una contraseña con la que se obtiene el hash que se está buscando. A medida que aumenta la longitud de las posibles contraseñas o mensajes a utilizar, la cantidad de tiempo, en promedio, para encontrar el mensaje original aumenta exponencialmente.

Ejercicio 05

Averigue el mensaje al que se le aplicó la función de hash para generar los siguientes resúmenes:

Pista: Deduzca la función de hash a partir del formato (longitud) del resumen o hash.

- a. 21232f297a57a5a743894a0e4a801fc3
 - b. e731a7b612ab389fcb7f973c452f33df3eb69c99
 - c. 796DD619207C4E357FD432FDF962C958BA1DF4CD6785246937223BC8D
C4FBF01794EBFF0159A175D9BE65B8EA4E7F46B80CCFFA4ED2A21773
D358C523DDDD382
-

Ejercicio 06

Resolver el reto alojado en el puerto **11006** del sitio ic.catedras.linti.unlp.edu.ar

Ejercicio 07

Resolver el reto alojado en el puerto **11007** del sitio ic.catedras.linti.unlp.edu.ar
El servidor informa el hash SHA-256 de una contraseña. El servicio pide que se crackee y devuelva la password que se usó para generarlo.

Pista: La password está entre las primeras 100 passwords del diccionario rockyou.

El diccionario lo puede descargar de <https://wiki.skullsecurity.org/Passwords>

Ejercicio 08

Crackear usando hashcat o johntheripper el siguiente hash md5 con salt:

```
85f978e2c12fedbf8869b219a1b2576a
```

Pista: Formato MD5(\$salt.\$pass)

Pista: Usa como salt el prefijo "IntroCiberseguridad"

Pista: rockyou.txt

Algoritmos de cifrado modernos

Los algoritmos de cifrado modernos se pueden agrupar en algoritmos de cifrado simétrico por un lado o en algoritmos de cifrado asimétrico por otro.

Sistemas de cifrado simétricos (También llamados de clave privada)

En un algoritmo de cifrado simétrico (por ejemplo, DES o AES), el emisor y el receptor deben tener una clave compartida configurada por adelantado y mantenida en secreto; el remitente usa esta clave para el cifrado, y el receptor usa esta misma clave para el descifrado. La mayoría de los algoritmos de cifrado de bloques se basan en este modo de operación.

El principal problema con los sistemas de cifrado simétrico no está ligado a su seguridad, sino al intercambio/distribución de claves. Una vez que el remitente y el destinatario hayan intercambiado las claves pueden usarlas para comunicarse con seguridad, pero ¿qué canal de comunicación que sea seguro han usado para transmitir las claves? Sería mucho más fácil para un atacante intentar interceptar una clave que probar las posibles combinaciones del espacio de claves.

Ejercicio 09

AES en modo ECB

El siguiente es el resultado de cifrar un string utilizando AES y codificando luego la salida en base64. Para cifrar se utilizó el algoritmo AES-128 modo ECB con la clave "CLAVE RE SECRETA" (sensible a mayúsculas, sin comillas)

```
dV5t6M4m2AcjYwsxC9i0+YXlc0r0ClfwyTGtpuWdPh9fvH+8cejJWOHYq1qH7qA+Kj
7Lci133Awj3rnoq42p532+fvbN64oZ8R/TlMkhw47nmIM5gPN+rt45985jeiIDbdpC
u1ig09Rzep14/kawM1AzFtoMzTvadmX11qSFp+UD81yiRz6HjaFLIIIIQnbzFrmcOI
OGEQ6LBEYz2cTW6JPBs7MHpqDrcrzZoLcb7Ah2jQSIId+YZ90JmRt83yTe66a60kqL
5SoW7/463Suyyp9xDhrgFu6YS3ScNDgOamADICmLUTxrvYooZIjL7s+thek3aBPrv
```

```
/yB84YNUhX7M0xjiTiP02nBJ1E1d0A0ew75BeARB4cHKVfLMnPMkjSYyiQ2eTwqYd4  
cZ+14Z9joNVA1Uei8Pg4KITPfJYy3Mc=
```

Develar el mensaje mediante un script en python que use una librería como pycrypto.

Ejercicio 10

Cifrado XOR

El siguiente es el resultado de cifrar un string utilizando XOR y codificando luego la salida en hex. El cifrado XOR utilizó una clave de 1 byte (1 carácter). Programar un script para encontrar la clave utilizada y develar el mensaje original.

Pista: Usar fuerza bruta para probar todas las posibles claves.

```
08296632232822342f27356637332366252f2034273466252928661e09146a66252e236  
866162334296624332328296a662a2766202a2721662223662335322366342332296623  
35660f053d092c7619257628193e7634676767737e7f737f73737f192527352f192e272  
52d232334343b
```
