OWASP

Open Web Application Security Project

Prácticas Seguras de Criptografía en Aplicaciones WEB

Henry Sánchez

*@_g05u_*

AGENDA

OWASP

Open Web Application Security Project

¿QUIEN SOY?

OWASP
Open Web Application Security Project

Security Leader en INZAFE PERU S.A., 7 años de experiencia en TI, 5 años de experiencia profesional en seguridad de la información.

(OSCP) Offensive Security Certified Profesional

(CPTE) Penetration Testing Engineer

CTF entusiasta, participante de muchos CTF como miembro del equipo NULL Life CTF Team.

Intereses principales: Reversing y exploiting

CRIPTOGRAFÍA

OWASP
Open Web Application Security Project

Del griego κρύπτος '(criptos), «oculto», y γραφη (grafé), «escritura», literalmente «escritura oculta»

Arte de escribir con clave secreta o de un modo enigmático

CODIFICAR O CIFRAR

OWASP
Open Web Application Security Project

CODIFICAR: Transformar la información a un conjunto de símbolos arbitrarios.

CIFRAR: Transformar la información con el fin de ocultarla.

CODIFICAR O CIFRAR

OWASP
Open Web Application
Security Project

OWASP

4F57415350

J5LUCU2Q

T1dBU1A=

CODIFICAR O CIFRAR

OWASP
Open Web Application Security Project

ALGORITMOS DE CODIFICACIÓN

- Base64
- UUEncode
- yEnc
- Base32

ALGORITMOS DE CIFRADO

- AES (Rijndael)
- Blowfish
- 3DES
- 3WAY

USO DE ALGORITMOS DEBILES

OWASP
Open Web Application Security Project

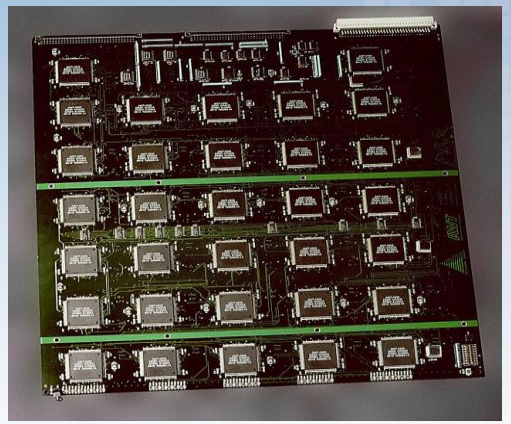
Un buen sistema de cifrado pone toda la seguridad en la clave y ninguna en el algoritmo.

Actualmente, los ordenadores pueden descifrar claves con extrema rapidez, y ésta es la razón por la cual el tamaño de la clave es importante en los criptosistemas modernos.

USO DE ALGORITMOS DEBILES

OWASP
Open Web Application Security Project

DES
Data Encryption Standard

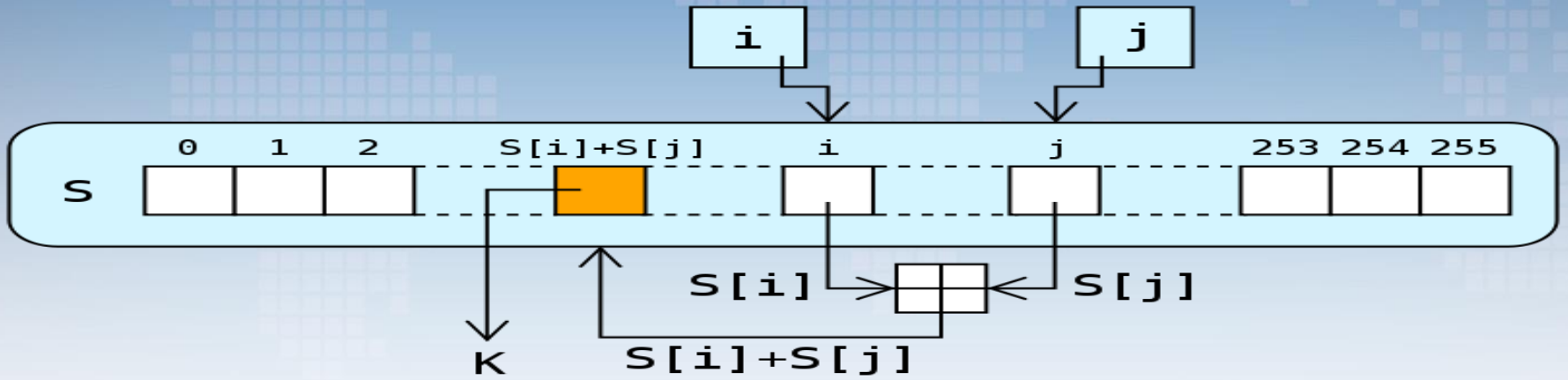


- Creado en 1972 por el NIST.
- Clave de 56 bits
- 2^{56} (72.057.594.037.927.936 claves)
- Un PC genérico puede romperlo en días.
- Hardware especializado puedo romperlo en cuestión de horas

USO DE ALGORITMOS DEBILES

OWASP
Open Web Application Security Project

RC4 (Rivest Cipher 4)



- Creado en 1984 por RSA Security .
- Cifrado de flujo
- Muchos texto cifrados con la misma clave permiten obtener el texto en plano

USO DE ALGORITMOS DEBILES

OWASP
Open Web Application Security Project

Algoritmos que no se deben usar:

- DES
- RC4
- IDEA
- RC2
- RSA (key < 2048)
- MD2, MD4, MD5*
- SHA1*

USO DE ALGORITMOS DEBILES

OWASP
Open Web Application Security Project

Algoritmos recomendados:

- AES (128, 192, 256 bits)
- AES 256 (115792089237316195423570985008687907853269984665640564039457584007913129639936)
- Twofish
- Blowfish
- 3DES
- 3WAY
- GOST
- SHA256, SHA512*

NÚMEROS ALEATORIOS

OWASP
Open Web Application Security Project

Generación segura de números aleatorios

```
int getRandomNumber()
{
    return 4;  // chosen by fair dice roll.
               // guaranteed to be random.
}
```

NÚMEROS ALEATORIOS

OWASP
Open Web Application Security Project

- **RNG:** Generador de números aleatorios.
- **PRNG:** Pseudo generador de números aleatorios.
- **CSPRNG:** Pseudo generado de números aleatorios criptográficamente seguros.

NÚMEROS ALEATORIOS

OWASP
Open Web Application Security Project

Inicialización insegura de PRNG

```php
<?php

function generate_token() {
    mt_srand(time());

    $key = '';
    $chars = 'abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789';
    for ($i = 0; $i < 20; $i++) {
        $key .= $chars[mt_rand(0, 61)];
    }

    return $key;
}

while (true) {
    $microtime = date("H:i:s.") . end(explode('.', microtime(true)));
    echo $microtime . ' : ' . time() . ' : ' . generate_token() . PHP_EOL;
}
```

Inicialización insegura del PRNG

NÚMEROS ALEATORIOS

OWASP
Open Web Application Security Project

Inicialización insegura de PRNG

microtime()	time()	Token
00:17:09.993532	1416547029	Wm3uC6cjcUiYlFYG6R5n
00:17:09.994033	1416547029	Wm3uC6cjcUiYlFYG6R5n
00:17:09.994533	1416547029	Wm3uC6cjcUiYlFYG6R5n
00:17:09.994533	1416547029	Wm3uC6cjcUiYlFYG6R5n
00:17:09.995033	1416547029	Wm3uC6cjcUiYlFYG6R5n
00:17:09.998033	1416547029	Wm3uC6cjcUiYlFYG6R5n
00:17:09.998533	1416547029	Wm3uC6cjcUiYlFYG6R5n
00:17:09.998533	1416547029	Wm3uC6cjcUiYlFYG6R5n
00:17:09.999033	1416547029	Wm3uC6cjcUiYlFYG6R5n
00:17:09.999033	1416547029	Wm3uC6cjcUiYlFYG6R5n
00:17:09.999533	1416547029	Wm3uC6cjcUiYlFYG6R5n
00:17:09.999533	1416547029	Wm3uC6cjcUiYlFYG6R5n
00:17:10.000033	1416547030	hFWt5QX1bt68qnwjvlYf
00:17:10.000033	1416547030	hFWt5QX1bt68qnwjvlYf
00:17:10.000533	1416547030	hFWt5QX1bt68qnwjvlYf
00:17:10.000533	1416547030	hFWt5QX1bt68qnwjvlYf
00:17:10.008034	1416547030	hFWt5QX1bt68qnwjvlYf
00:17:10.009034	1416547030	hFWt5QX1bt68qnwjvlYf
00:17:10.009534	1416547030	hFWt5QX1bt68qnwjvlYf
00:17:10.010035	1416547030	hFWt5QX1bt68qnwjvlYf
00:17:10.010535	1416547030	hFWt5QX1bt68qnwjvlYf
00:17:10.011035	1416547030	hFWt5QX1bt68qnwjvlYf
00:17:10.011535	1416547030	hFWt5QX1bt68qnwjvlYf
00:17:10.011535	1416547030	hFWt5QX1bt68qnwjvlYf

Token predecible al inicializar el generador de números aleatorios de forma insegura

mt_srand(time())

NÚMEROS ALEATORIOS

OWASP
Open Web Application Security Project

Uso de CSPRNG

```php
<?php

function generate_token() {
    $key = '';
    $chars = 'abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789';
    for ($i = 0; $i < 20; $i++) {
        $key .= $chars[ord(openssl_random_pseudo_bytes(1, $cstrong)) % 62];
    }

    return $key;
}

while (true) {
    $microtime = date("H:i:s.") . end(explode('.', microtime(true)));
    echo $microtime . ' : ' . time() . ' : ' . generate_token() . PHP_EOL;
}
```

NÚMEROS ALEATORIOS

OWASP
Open Web Application Security Project

Uso de CSPRNG

microtime()	time()	Token
00:58:17.997429	1416549497	icQiZlEGJza63mRfaEzp
00:58:17.997929	1416549497	z8gYBXTNnl3STjjnKaHH
00:58:17.997929	1416549497	MUCYmdjuGJbmE7gMxbI9
00:58:17.998429	1416549497	gO4EnJsPMxUYuD8O8wsI
00:58:17.998929	1416549497	acsMNbt8gV1oJRMn9woO
00:58:17.998929	1416549497	JC8ehIti4g7BApYs7p9n
00:58:17.999429	1416549497	b8Cbdcc4olbdrrs9ANTK
00:58:17.999929	1416549497	E9WGCU2Bjh89vhTSXioE
00:58:17.999929	1416549497	tKdnhCZYOs7NJd9nB9rh
00:58:18.00043	1416549498	UttLWmr7VLz5nOHL3bt5
00:58:18.00043	1416549498	QggQimJuttZ1txzGZWWQ
00:58:18.00093	1416549498	eevb84eiizZUBb9ijdnw
00:58:18.00143	1416549498	oBXVn01kOnjRcWvFhX9s
00:58:18.00143	1416549498	R5dZqiuDyUUv8UjljTSw
00:58:18.00193	1416549498	QhjbdQ18CXKpfSNn3Qf4
00:58:18.00193	1416549498	QPTv1JxPgHHOuPcgcsWJ
00:58:18.00343	1416549498	Ra7vgIOsYopHiIB7fYC6
00:58:18.00343	1416549498	u1IVuxpx6BQoWmE7dyl4

NÚMEROS ALEATORIOS

OWASP
Open Web Application Security Project

Auditoría del 2012 mostrando varias aplicaciones vulnerables (el punto indica el compromiso de cuentas o paneles de administración a través de diferentes tipos de ataques:

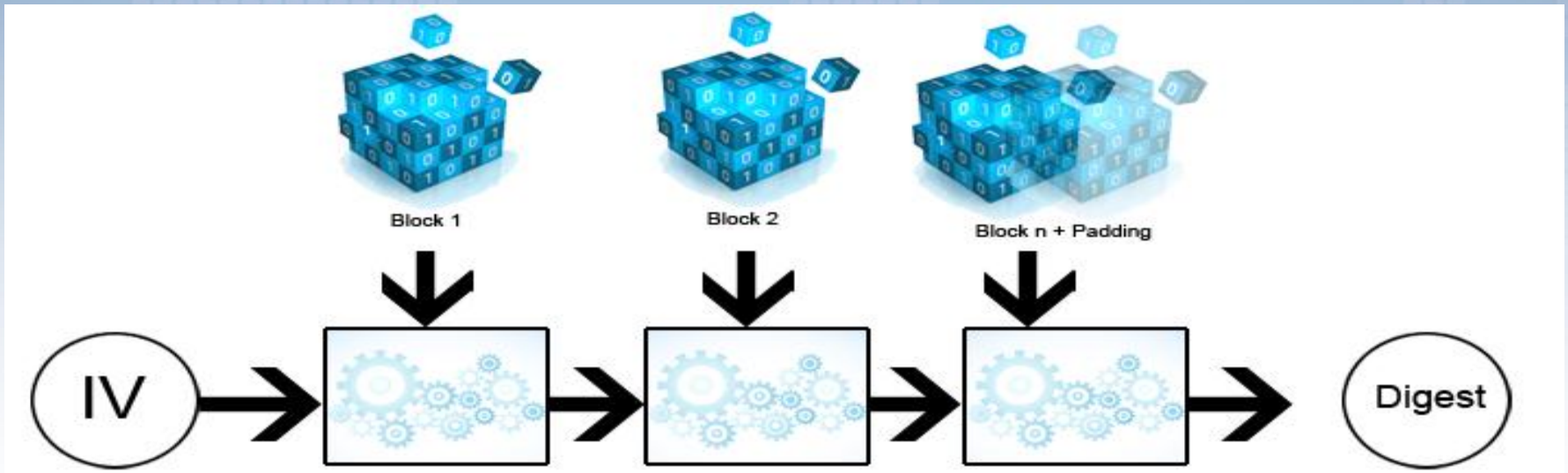
App / Attack	Time		Seed			State recovery	
Section	ATS	RT	4.1	4.2	4.3	5.3	5.4
mediawiki				•	•	•	
Open eClass				•	•		•
taskfreak				•	•	•	
zen-cart	•	•					
osCommerce 2.x	•	•					
osCommerce 3.x				•	•		•
elgg	$\bullet^c$			•	•		
Gallery		$\bullet^c$	$\bullet^c$	$\bullet^c$			
Joomla					•		
MyBB	$\circ^c$		$\circ^c$			$\circ^c$	
IP Board	$\bullet^c$		$\bullet^c$	$\bullet^c$			
phorum				•	•	•	
HotCRP				•	•	•	
gazelle					•	•	
tikiWiki				•	•		•
SMF	$\circ^c$				$\circ^c$		

Figure 13: Summary of audit results. The *c* superscript denotes that the attack need to be used in combination with other attacks with the same superscript. The • denotes a full attack while ∘ denotes a weakness for which the practical exploitation is either unverified or requires very specific configurations.

HASH LENGTH ATTACK

OWASP
Open Web Application Security Project

HASH LENGTH ATTACK EXTENSION



HASH LENGTH ATTACK

OWASP
Open Web Application Security Project

¿Qué es un hash?

- Los hash o funciones de resumen son algoritmos que consiguen crear a partir de una entrada una salida única de tamaño fijo.
- Es unidireccional (irreversible).
- Se usa para verificar la integridad de mensajes, firma digital.
- El conjunto de salida es finito.
- Es vulnerable a colisiones.

HASH LENGTH ATTACK

OWASP
Open Web Application Security Project

El Programa Vulnerable

http://example.com/download?file=report.pdf&mac=563162c9c71a17367d44c165b84b85ab59d036f9

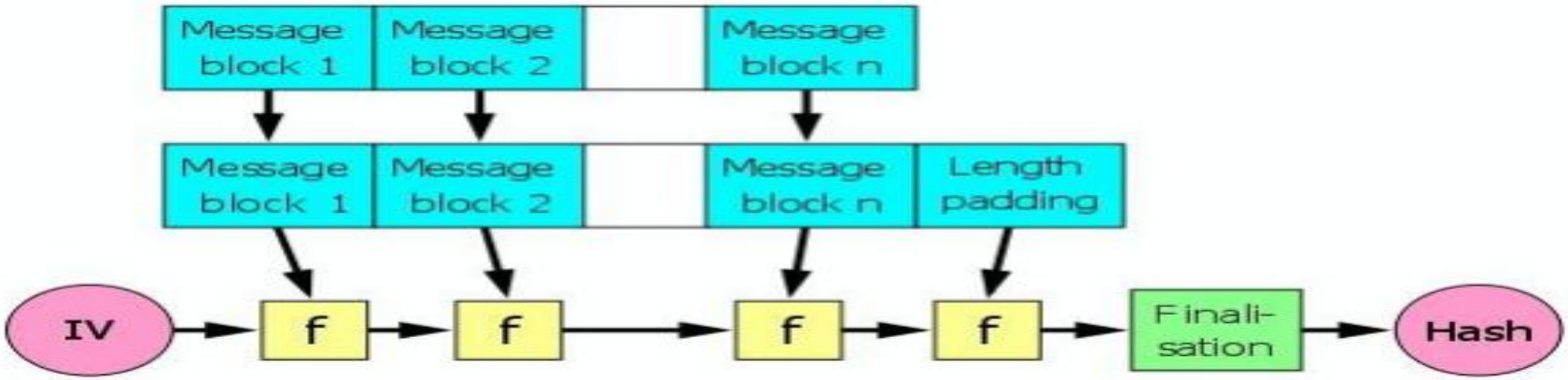
```php
<?php
    $SECRET = 'xxxxxxxxxxxxxxxxxxxxxxxxxxxx';
    $file = $_GET['file'];
    $mac = $_GET['mac']

    if (md5($_SECRET . $file) === $mac) {
        echo file_get_contents($file); //se descarga el archivo
    }
    ....
```

HASH LENGTH ATTACK

OWASP
Open Web Application Security Project

El Ataque

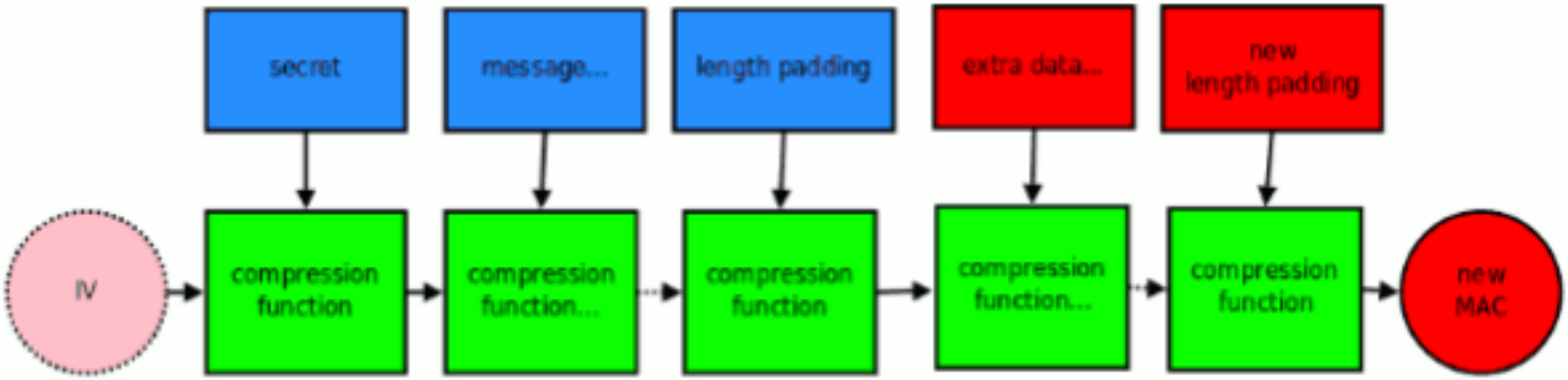
- Cada algoritmo trabaja en bloques fijos.
- MD5 trabaja en bloques de 512 bits.
- Si el mensaje a procesar es menor se completa con un relleno (padding).



HASH LENGTH ATTACK

OWASP
Open Web Application Security Project

One-Way Hash Function MAC Broken
With Merkle-Damgaard Strengthening



Flaw

Anyone can still tack data and a new length onto the end of the message and generate a new MAC

HASH LENGTH ATTACK

Open Web Application Security Project

El Ataque

http://example.com/download?file=report.pdf&mac=
563162c9c71a17367d44c165b84b85ab59d036f9

http://example.com/download?file=report.pdf%80%0
0%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%0
0%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%0
0%00%00%00%00%00%00%00%00%A8/../../../../../../.
./etc/passwd&mac=ee40aa8ec0cfafb7e2ec4de20943b67
3968857a5

HASH LENGTH ATTACK

CASOS CONOCIDOS

- Scribd http://www.scribd.com/
- Vimeo http://www.vimeo.com/
- Zooomr http://www.zooomr.com/
- Flickr

http://www.flickr.com/services/auth/?api_key=44fefa051fc1c61f5e76f27e620f51d5&extra=/login&perms=write&api_sig=38d39516d896f879d403bd327a932d9e

http://www.flickr.com/services/auth/?api_key=44fefa051fc1c61f5e76f27e620f51d5&extra=/loginpermswrite%80%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%60%02%00%00%00%00%00%00&api_key=44fefa051fc1c61f5e76f27e620f51d5&extra=http://vnsecurity.net&perms=write&api_sig=a8e6b9704f1da6ae779ad481c4c165a3

HASH LENGTH ATTACK

OWASP
Open Web Application Security Project

¿Cómo solucionarlo?

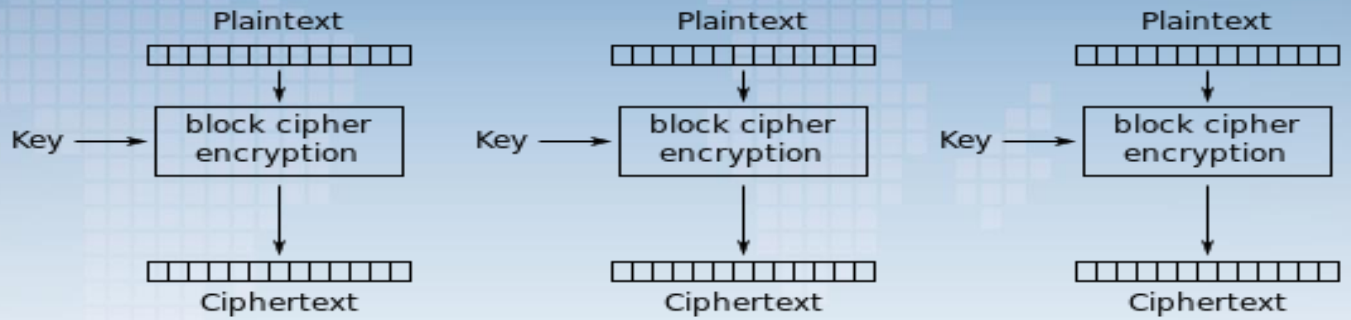
- Usando funciones HMAC (Hash Message Authention Code)
- Es una función que valiéndose de un algoritmo HASH autentica a dos usuarios mediante una clave secreta.
- Como se define?
- HMAC_HASH = HASH(SECRETO + HASH(SECRETO + INPUT))

ORACLE PADDING ATTACK

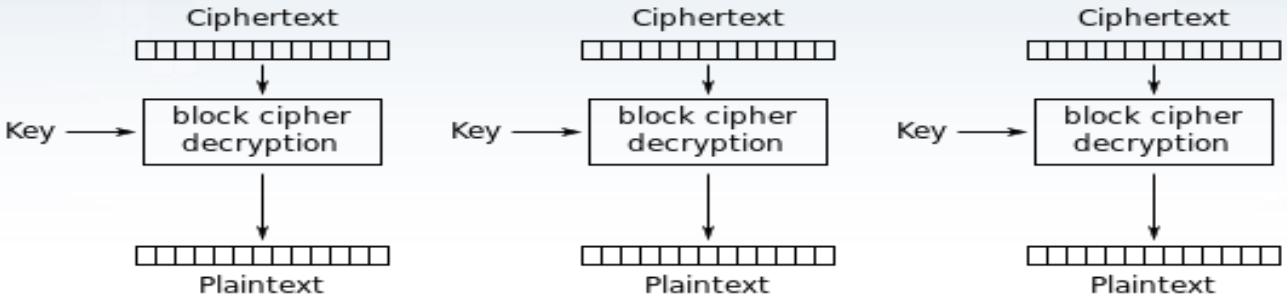
ORACLE PADDING ATTACK

OWASP
Open Web Application Security Project

Modo de cifrado/descifrado ECB



Electronic Codebook (ECB) mode encryption

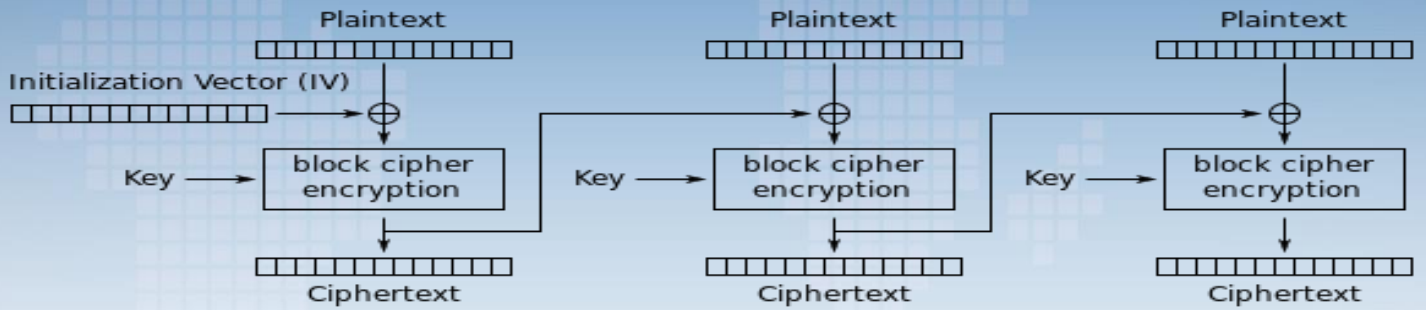


Electronic Codebook (ECB) mode decryption

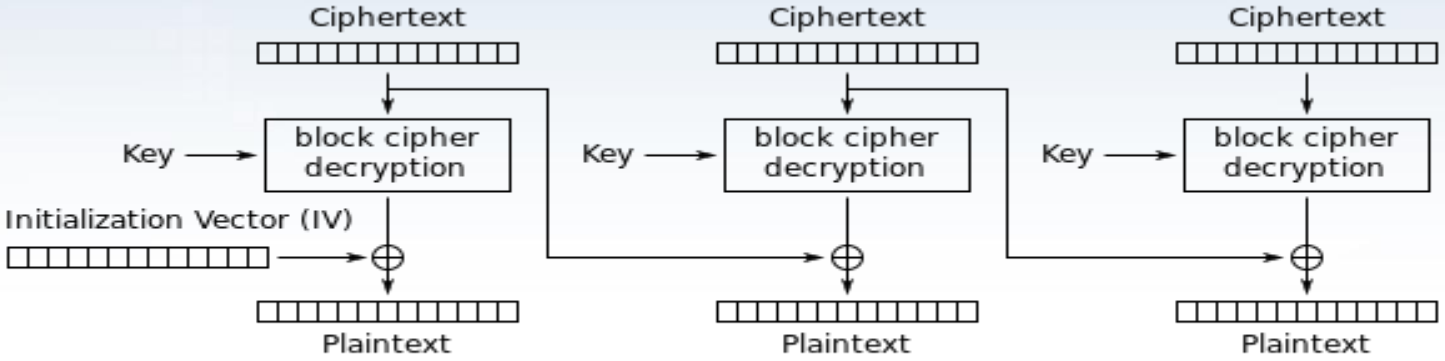
ORACLE PADDING ATTACK

OWASP
Open Web Application Security Project

Modo de cifrado/descifrado CBC



Cipher Block Chaining (CBC) mode encryption



Cipher Block Chaining (CBC) mode decryption

ORACLE PADDING ATTACK

OWASP
Open Web Application Security Project

El escenario

- Se usa un modo de cifrado CBC.
- Los mensajes generados por la aplicación nos permite identificar si hubo un descifrado correcto o no.
- Se puede controlar el IV.
- El texto cifrado es: BRIAN;12;2
- http://appvuln/money.jsp?UID=**7B216A634951170FF851D6CC68FC9537858795A28ED4AAC6**

	INITIALIZATION VECTOR								BLOCK 1 of 2								BLOCK 2 of 2							
	1	2	3	4	5	6	7	8	1	2	3	4	5	6	7	8	1	2	3	4	5	6	7	8
Plain-Text	-	-	-	-	-	-	-	-	B	R	I	A	N	;	1	2	;	1	;					
Plain-Text (Padded)	-	-	-	-	-	-	-	-	B	R	I	A	N	;	1	2	;	1	;	0x05	0x05	0x05	0x05	0x05
Encrypted Value (HEX)	0x7B	0x21	0x6A	0x63	0x49	0x51	0x17	0x0F	0xF8	0x51	0xD6	0xCC	0x68	0xFC	0x95	0x37	0x85	0x87	0x95	0xA2	0x8B	0xD4	0xAA	0xC6

ORACLE PADDING ATTACK

OWASP
Open Web Application Security Project

	BLOCK 1 of 2								BLOCK 2 of 2							
	1	2	3	4	5	6	7	8	1	2	3	4	5	6	7	8
Initialization Vector	0x7B	0x21	0x6A	0x63	0x49	0x51	0x17	0x0F	0xF8	0x51	0xD6	0xCC	0x68	0xFC	0x95	0x37
	⊕	⊕	⊕	⊕	⊕	⊕	⊕	⊕	⊕	⊕	⊕	⊕	⊕	⊕	⊕	⊕
Plain-Text (Padded)	B	R	I	A	N	;	1	2	;	1	;	0x05	0x05	0x05	0x05	0x05
	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓
Intermediary Value (HEX)	0x39	0x73	0x23	0x22	0x07	0x6A	0x26	0x3D	0xC3	0x60	0xED	0xC9	0x6D	0xF9	0x90	0x32
	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓
	TRIPLE DES								TRIPLE DES							
	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓
Encrypted Output (HEX)	0xF8	0x51	0xD6	0xCC	0x68	0xFC	0x95	0x37	0x85	0x87	0x95	0xA2	0x8E	0xD4	0xAA	0xC6

ORACLE PADDING ATTACK

OWASP
Open Web Application Security Project

	BLOCK 1 of 2								BLOCK 2 of 2							
	1	2	3	4	5	6	7	8	1	2	3	4	5	6	7	8
Encrypted Input (HEX)	0xF8	0x51	0xD6	0xCC	0x68	0xFC	0x95	0x37	0x85	0x87	0x95	0xA2	0x8E	0xD4	0xAA	0xC6
	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓
	TRIPLE DES								TRIPLE DES							
	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓
Intermediary Value (HEX)	0x39	0x73	0x23	0x22	0x07	0x6a	0x26	0x3D	0xC3	0x60	0xED	0xC9	0x6D	0xF9	0x90	0x32
	⊕	⊕	⊕	⊕	⊕	⊕	⊕	⊕	⊕	⊕	⊕	⊕	⊕	⊕	⊕	⊕
Initialization Vector	0x7B	0x21	0x6A	0x63	0x49	0x51	0x17	0x0F	0xF8	0x51	0xD6	0xCC	0x68	0xFC	0x95	0x37
	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓
Plain-Text (Padded)	B	R	I	A	N	;	1	2	;	1	;	0x05	0x05	0x05	0x05	0x05

VALID PADDING

ORACLE PADDING ATTACK

OWASP
Open Web Application Security Project

Mensajes de la Aplicación

- Se realiza un descifrado correcto con datos válidos.
- Se realiza un descifrado correcto con datos inválidos.
- No se descifro.

ORACLE PADDING ATTACK
EL ATAQUE

OWASP
Open Web Application Security Project

`http://appvuln/money.jsp?UID=0000000000000000F851D6CC68FC9537`

	BLOCK 1 of 1							
	1	2	3	4	5	6	7	8
Encrypted Input	0xF8	0x51	0xD6	0xCC	0x68	0xFC	0x95	0x37
	↓	↓	↓	↓	↓	↓	↓	↓
	TRIPLE DES							
	↓	↓	↓	↓	↓	↓	↓	↓
Intermediary Value	0x39	0x73	0x23	0x22	0x07	0x6a	0x26	0x3D
	⊕	⊕	⊕	⊕	⊕	⊕	⊕	⊕
Initialization Vector	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00
	↓	↓	↓	↓	↓	↓	↓	↓
Decrypted Value	0x39	0x73	0x23	0x22	0x07	0x6a	0x26	0x3D

INVALID PADDING

ORACLE PADDING ATTACK
EL ATAQUE

OWASP
Open Web Application Security Project

`http://appvuln/money.jsp?UID=0000000000000001F851D6CC68FC9537`

	BLOCK 1 of 1							
	1	2	3	4	5	6	7	8
Encrypted Input	0xF8	0x51	0xD6	0xCC	0x68	0xEC	0x95	0x37
	↓	↓	↓	↓	↓	↓	↓	↓
	TRIPLE DES							
	↓	↓	↓	↓	↓	↓	↓	↓
Intermediary Value	0x39	0x73	0x23	0x22	0x07	0x6a	0x26	0x3D
	⊕	⊕	⊕	⊕	⊕	⊕	⊕	⊕
Initialization Vector	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x01
	↓	↓	↓	↓	↓	↓	↓	↓
Decrypted Value	0x39	0x73	0x23	0x22	0x07	0x6a	0x26	0x3C

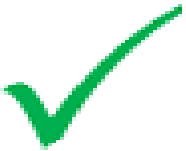
INVALID PADDING

ORACLE PADDING ATTACK
EL ATAQUE

OWASP
Open Web Application Security Project

`http://appvuln/money.jsp?UID=0000000000000003CF851D6CC68FC9537`

	Block 1 of 1							
	1	2	3	4	5	6	7	8
Encrypted Input	0xF8	0x51	0xD6	0xCC	0x68	0xFC	0x95	0x37
	↓	↓	↓	↓	↓	↓	↓	↓
	TRIPLE DES							
	↓	↓	↓	↓	↓	↓	↓	↓
Intermediary Value	0x39	0x73	0x23	0x22	0x07	0x6a	0x26	0x3D
	⊕	⊕	⊕	⊕	⊕	⊕	⊕	⊕
Initialization Vector	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x3C
	↓	↓	↓	↓	↓	↓	↓	↓
Decrypted Value	0x39	0x73	0x23	0x22	0x07	0x6a	0x26	0x01



VALID PADDING

ORACLE PADDING ATTACK
EL ATAQUE

OWASP
Open Web Application Security Project

`http://appvuln/money.jsp?UID=0000000000000003FF851D6CC68FC9537`

	Block 1 of 1							
	1	2	3	4	5	6	7	8
Encrypted Input	0xF8	0x51	0xD6	0xCC	0x68	0xFC	0x95	0x37
	↓	↓	↓	↓	↓	↓	↓	↓
	TRIPLE DES							
	↓	↓	↓	↓	↓	↓	↓	↓
Intermediary Value	0x39	0x73	0x23	0x22	0x07	0x6a	0x26	0x3D
	⊕	⊕	⊕	⊕	⊕	⊕	⊕	⊕
Initialization Vector	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x3F
	↓	↓	↓	↓	↓	↓	↓	↓
Decrypted Value	0x39	0x73	0x23	0x22	0x07	0x6a	0x26	0x02

INVALID PADDING

ORACLE PADDING ATTACK
EL ATAQUE

OWASP
Open Web Application Security Project

`http://appvuln/money.jsp?UID=0000000000000243FF851D6CC68FC9537`

	1	2	3	4	5	6	7	8
Encrypted Input	0xF8	0x51	0xD6	0xCC	0x68	0xFC	0x95	0x37
	↓	↓	↓	↓	↓	↓	↓	↓
	TRIPLE DES							
	↓	↓	↓	↓	↓	↓	↓	↓
Intermediary Value	0x39	0x73	0x23	0x22	0x07	0x6a	0x26	0x3D
	⊕	⊕	⊕	⊕	⊕	⊕	⊕	⊕
Initialization Vector	0x00	0x00	0x00	0x00	0x00	0x00	0x24	0x3F
	↓	↓	↓	↓	↓	↓	↓	↓
Decrypted Value	0x39	0x73	0x23	0x22	0x07	0x26	0x02	0x02

VALID PADDING

ORACLE PADDING ATTACK
EL ATAQUE

OWASP
Open Web Application Security Project

`http://appvuln/money.jsp?UID=327B2B2A0F622E35F851D6CC68FC9537`

	1	2	3	4	5	6	7	8
Encrypted Input	0xF8	0x51	0xD6	0xCC	0x68	0xFC	0x95	0x37
	↓	↓	↓	↓	↓	↓	↓	↓
	TRIPLE DES							
	↓	↓	↓	↓	↓	↓	↓	↓
Intermediary Value	0x39	0x73	0x23	0x22	0x07	0x6a	0x26	0x3D
	⊕	⊕	⊕	⊕	⊕	⊕	⊕	⊕
Initialization Vector	0x31	0x7B	0x2B	0x2A	0x0F	0x62	0x2E	0x35
	↓	↓	↓	↓	↓	↓	↓	↓
Decrypted Value	0x08	0x08	0x08	0x08	0x08	0x08	0x08	0x08

VALID PADDING

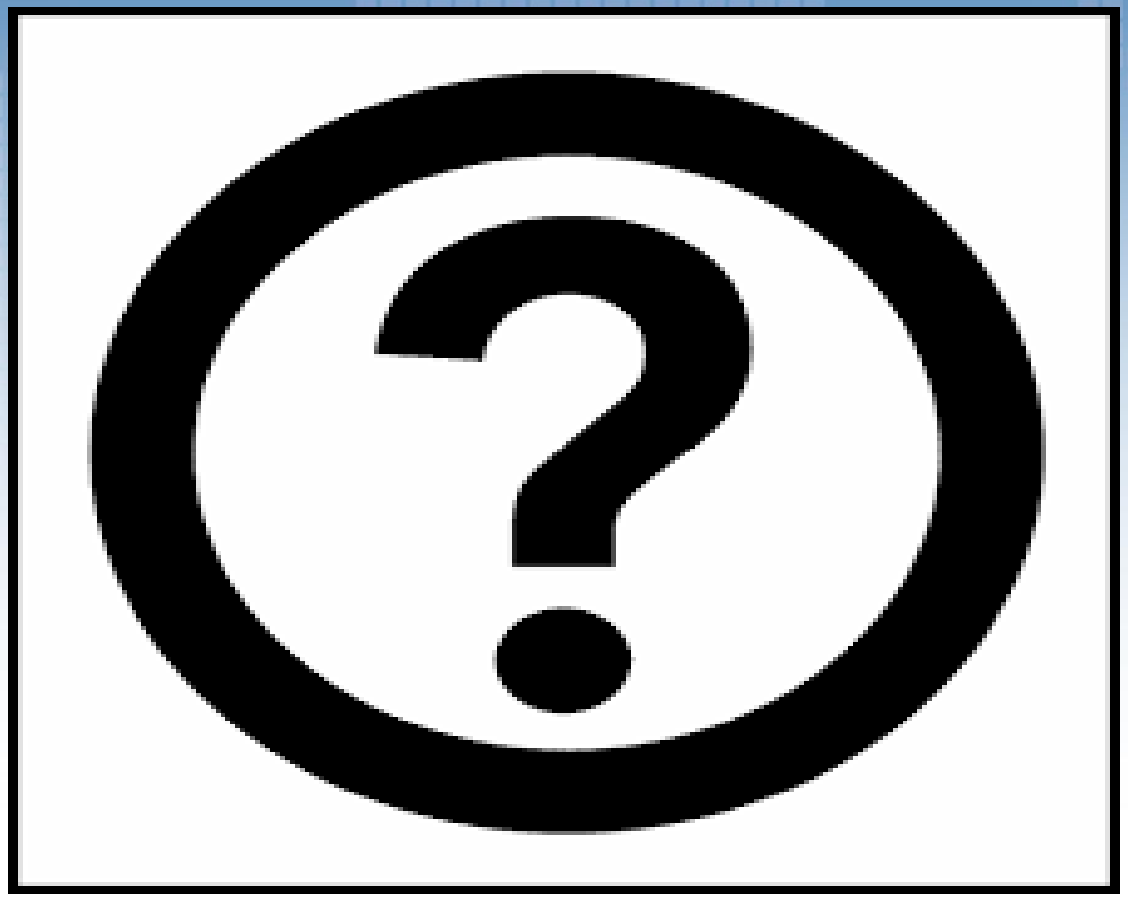
ORACLE PADDING ATTACK
EL ATAQUE

OWASP
Open Web Application Security Project

CIFRANDO ARBITRARIAMENTE

	1	2	3	4	5	6	7	8
Encrypted Input	0xF8	0x51	0xD6	0xCC	0x68	0xFC	0x95	0x37
	↓	↓	↓	↓	↓	↓	↓	↓
	TRIPLE DES							
	↓	↓	↓	↓	↓	↓	↓	↓
Intermediary Value	0x39	0x73	0x23	0x22	0x07	0x6a	0x26	0x3D
	⊕	⊕	⊕	⊕	⊕	⊕	⊕	⊕
Initialization Vector	0x6D	0x36	0x70	0x76	0x03	0x6E	0x22	0x39
	↓	↓	↓	↓	↓	↓	↓	↓
Decrypted Value	T	E	S	T	0x04	0x04	0x04	0x04

¿PREGUNTAS?



ARGOZ

OWASP
Open Web Application Security Project

- Una comunidad contra el fraude bancario.
- No existen sitios en español.
- No brindan ningún beneficio.
- Lanzamiento el 21 de Junio.
- En 14 países simultáneamente.
- Beneficios para universidades.
- Atentos Twitter @OWASP_Peru