Examen de Seguridad en Sistemas Informáticos

Ene 2022

Pregunta 1 (1)

Tenemos la web con el siguiente código fuente alojada en "genius.org":

```
<?php

    $bd = mysqli_connect("127.0.0.1", "animanegra", "palangana", "HackMe");

    $PASS = $_GET['password'];

    $PASS = str_replace("or","",$PASS);$PASS = str_replace("OR","",$PASS);
    $PASS = str_replace("Or","",$PASS);$PASS = str_replace("oR","",$PASS);

    $res = mysqli_query($bd,"SELECT * FROM users WHERE password='".$PASS."'");

    if($datos = mysqli_fetch_array($res)){

        include("private.php");

    }else{

        echo "Access denied.\n";

    }

?>
```

Las passwords son suficientemente fuertes como para morir antes de que termine de obtenerlas por fuerza bruta y no estan en ningun diccionario conocido por hombres, mujeres ni jugadores de LOL. Sabiendo esto:

¿Se puede obtener acceso a la parte privada del sistema sin saber las credenciales? Diga como.

.

Pregunta 2 (1)

Hemos conseguido las siguientes parejas usuario y hash de una base de datos:

```
admin: 1f7d50c56381bb05217922bda930004c
ramon: ae7a182bc79026630f23bfe808b7c540
maruja: c77b705e74a167df9e9255bd9dc95b68
maria: e3de622eaf0624bc2bd8f9c6e89061dd
pedro: b9643603717125adc376b02980b332b6
juan pedro: 9aaf6a39c1f2241ae120114322ded631
roberto: 7997e79dd35e7fd418f3d492cfadbdaf
```

¿Si deseamos la password en limpio que se podría hacer? Diga que programas concretos utilizaría en caso de necesitarlos.

.

Pregunta 3 (1)

Explique brevemente el funcionamiento del TOTP:

.

Pregunta 4 (1)

Estamos monitorizando el tráfico de la empresa y desde un ordenador que no es XXX.XXX.XXX.XXX ni YYY.YYY.YYY.YYY. Y ejecutamos lo siguiente:

```
animanegra@burrito:~/$ sudo tcpdump port 80
...
09:24:34.701226 IP XXX.XXX.XXX.XXX.56574 > YYY.YYY.YYY.YYY.http: Flags [P.], seq
    1:409, ack 1, win 512, options [nop,nop,TS val 26498291 ecr 26498291], length
    408: HTTP: GET /admin/ HTTP/1.1
E.....@.@..B..........P.2].zu/3..........
..T...T.GET /admin/ HTTP/1.1
Host: YYY.YYY.YYY.YYY
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:78.0) Gecko/20100101 Firefox/78.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q
    =0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
DNT: 1
Authorization: Basic dXNlcjE6bGFsYWxh
Connection: keep-alive
Upgrade-Insecure-Requests: 1
Cache-Control: max-age=0
...
```

Indique si a partir de la salida que vemos existe o podría existir algún problema de seguridad en la empresa y cual o cuales son esos problemas.

.

Pregunta 5 (1)

Ejecutamos lo siguiente en la consola del `gdb`:

```
(gdb) r < <(I=0; while [ $I -lt 50 ]; do printf "A"; I=$(($I+1)); done;
             printf "\x01\x02\x03\x04\n"; )
```

y supondremos que `0x01020304` es la dirección a la que se desea saltar ya que hay una función llamada `execute_me` que me devuelve una terminal en `bash`.

En un breakpoint justo antes de un `ret` hacemos un info frame que nos devuelve lo siguiente:

```
Stack level 0, frame at 0xffffd180:
 eip = 0x8049bfc in funcion (bad1.c:23); saved eip = 0x2014141
 called by frame at 0xffffd184
 source language c.
 Arglist at 0xffffd178, args:
 Locals at 0xffffd178, Previous frame's sp is 0xffffd180
 Saved registers:
  ebx at 0xffffd174, ebp at 0xffffd178, eip at 0xffffd17c
(gdb)
```

En estas circunstancias:

¿Que cambios mínimos debería hacer en mi script para conseguir saltar a la función que deseo?

.

Pregunta 6 (1)

Disponemos de una aplicación web que da servicio en el puerto 54471 pero no dispone de cifrado. La aplicación es segura, y dispone de autenticación basada en usuario y password por lo que si no tenemos credenciales el intentar sacarlas por fuerza bruta será vano (En el tiempo que dura nuestra corta vida).

Deseamos que se pueda acceder desde otra máquina pero la red por la que deben de pasar los datos es insegura. La máquina que da servicio tiene servidor de ssh activado y la máquina desde la que conectamos es una máquina accesible a través de Internet.

Ante esta situación me planteo 4 posibles soluciones ejecutando comandos desde la máquina que NO da el servicio:

Posibilidad 1

```
animanegra@burrito:/$ ssh -i ~/lalala usuario@servidor -L 54472:servidor:54471 -T
    -N &

y me conecto en mi navegador a la url:
http://servidor:54471/
```

Posibilidad 2

```
animanegra@burrito:/$ ssh -i ~/lalala usuario@servidor -R 54472:servidor:54471 -T
    -N &

y me conecto en mi navegador a la url:
http://servidor:54471/
```

Posibilidad 3

```
animanegra@burrito:/$ ssh -i ~/lalala usuario@servidor -L *:54472:servidor:54471
   -T -N &

y me conecto en mi navegador a la url:
http://127.0.0.1:54471/
```

Posibilidad 4

```
animanegra@burrito:/$ ssh -i ~/lalala usuario@servidor -R *:54472:servidor:54471
   -T -N &

y me conecto en mi navegador a la url:
http://127.0.0.1:54471/
```

Diga que opción se podría utilizar y por que el resto no. En caso de no funcionar ninguna, diga que cambiaría para que funcionase

.

Pregunta 7 (1)

En el siguiente programa de php:

```
<?php

        echo "<h1>Easy messaging v0.1.3.3.7</h1>Deja un texto para que lo lea el
           siguiente que se conecte<br><form><textarea name='texto'></textarea><
           br><input type='submit'></input></form>";

        echo "<h1>El texto que ha dejado el anterior usuario escrito es:</h1>";

        system("cat archivo.txt");
        if(isset($_GET['texto'])){


                file_put_contents("archivo.txt",$_GET['texto']);

                echo "<h1>El nuevo texto es:</h1>";

                system("cat archivo.txt");

        }

?>
```

¿Que problema de seguridad existe en este código y como se puede explotar? En caso de que esta página perteneciese a la parte pública de un sistema mas amplio que utilizase bases de datos, sesiones, autenticacion para acceder a partes privadas de este, etc... ¿Se podría aprovechar este problema para acceder a partes privadas del sistema? Diga como.

.

Pregunta 8 (1)

Un sistema web de código abierto genera URLs aleatorios de descarga de secretos para sus usuarios de la siguiente forma:

```
<?php

    function gen_secure_url() {
        $last_num=intval(file_get_contents("/var/data/contador"));
        $alea=md5($last_num);
        $last_num=$last_num+1;
            file_put_contents("/var/data/contador",$last_num);
        return "https://secret.io/s/".$alea;
    }

    $u=gen_secure_url();
    print('Your secret <a href="'.$u.'">here</a>');

?>
```

Cuando el usuario ha subido un fichero, si accede al url que se le devuelve puede descargar el contenido de este. Supondremos que los usuarios no van a leakear nunca las urls a sus propios archivos y que los dispositivos desde los que realizan las peticiones son seguros. Si tres usuarios diferentes suben archivos, se generarán las siguientes urls:

```
Al primero
Your secret <a href="https://secret.io/s/cfcd208495d565ef66e7dff9f98764da">here</
   a>%

Al segundo
Your secret <a href="https://secret.io/s/c4ca4238a0b923820dcc509a6f75849b">here</
   a>%

Al tercero
Your secret <a href="https://secret.io/s/c81e728d9d4c2f636f067f89cc14862c">here</
   a>%
```

Leyendo el código que se presenta ¿Qué problema tiene este sistema? ¿Existe alguna vulnerabilidad aqui?

.

Pregunta 9 (1)

Cifrar algo con la clave privada ¿Tiene algun tipo de utilidad? Evidentemente, supondremos que la clave pública es pública.

.

Pregunta 10 (1)

Disponemos de un equipo linux que funciona como router que separa las redes 10.1.X.X y la 10.2.X.X. Vemos a continucación las politicas puestas en su IP tables:

```
root@router1:~# iptables -n -L
Chain INPUT (policy DROP)
target     prot opt source               destination
ACCEPT     tcp  --  10.1.0.100          anywhere            tcp dpt:ssh
ACCEPT     tcp  --  10.2.0.101          anywhere            tcp spt:6969
ACCEPT     tcp  --  10.2.0.100          anywhere            tcp dpt:1234

Chain FORWARD (policy DROP)
target     prot opt source               destination
Chain OUTPUT (policy DROP)
target     prot opt source               destination
ACCEPT     tcp  --  anywhere            anywhere            tcp spt:ssh
ACCEPT     tcp  --  anywhere            10.2.0.101          tcp dpt:6969
ACCEPT     tcp  --  anywhere            10.2.0.100          tcp spt:1234
```

Si estoy en el equipo 10.1.0.1 e intento conectarme a un servidor que esté dando servicio tcp en el puerto 6969 del ordenador 10.2.0.100. ¿Podré hacerlo? ¿Que comandos mínimos de IPTables deberé ejecutar para poder realizar dicha conexión?

.