Solución al Reto Hacking v2.0 de Informática 64

Febrero 2007 © Daniel Kachakil

Introducción

Este documento describe dos soluciones posibles al segundo Reto Hacking de Informática 64 que se publicó el 10 de febrero de 2007 en la siguiente dirección web:

http://retohacking2.elladodelmal.com

El reto consistía en conseguir acceso a la administración del sitio web.

Pistas

En esta ocasión, la única pista inicial a nuestra disposición era la siguiente frase (disponible en el blog www.elladodelmal.com, en la que se publicaba el reto):

> *Para ello tendrás que unir tres cosas: lo que se ve, lo que no se ve ni verás, y lo que ya sabes que está donde siempre.*

Análisis inicial

Entrando en la página principal, inicialmente disponemos de un menú de navegación con las siguientes opciones:

- **Zona privada:** Formulario solicitando usuario y contraseña
- **Eventos:** Un listado de eventos (fecha, ciudad y nombre del evento)
- **Regístrate:** Formulario para obtener una nueva cuenta
- **Ganadores del reto:** Lista de ganadores que han superado el reto

Para conseguir la mayor información posible, el primer paso necesario es el de registrarse, rellenando un pequeño formulario con los siguientes datos: nombre de usuario, contraseña, dirección de correo electrónico y ciudad.

Una vez completado el registro, accedemos a la opción "Zona privada" e introducimos el usuario y contraseña que habíamos elegido en el paso anterior. El menú lateral cambia y se nos muestran las siguientes opciones:

- **Recordar eventos:** El listado de eventos anterior, más un botón "Recordar"
- **Ver recordatorios:** Listado de eventos en los que hemos solicitado recordatorio
- **Ganadores del reto:** La misma lista de ganadores

Lo que se ve

Cuando en la sección "Recordar eventos" marcamos un evento para recordar, se nos solicita un comentario opcional en un campo de texto multilínea. Una vez aceptado dicho texto, volvemos a la misma página en la que se nos mostrará resaltado el evento seleccionado. Por otro lado, el botón de "Recordar evento" cambia a "Eliminar evento". Para borrar un recordatorio, simplemente pulsamos sobre dicho botón y se nos solicitará una confirmación en una nueva página.

Al entrar en la sección "Ver recordatorios" veremos un listado con los eventos que hayamos seleccionado en la sección anterior. Cada uno de los eventos aparece con un botón "Ver comentario" que muestra el comentario que hemos introducido (en caso de haberlo hecho) y un botón "Ver documentos" para descargar un PDF con la información del evento.

Estos botones nos llevan respectivamente a las siguientes páginas:

```
http://retohacking2.elladodelmal.com/ZonaPrivada/getComentario.aspx?id=XX
http://retohacking2.elladodelmal.com/ZonaPrivada/descargar.aspx?id=XX
```

Hasta aquí podríamos decir que hemos cubierto la primera parte de la pista, es decir, lo que se ve. De momento tenemos varios candidatos para probar si son vulnerables a la inyección de SQL (parámetros, campos de texto, etc) y comprobamos que el parámetro "id" de la página de descargas admite inyección ciega de SQL.

Lo que no se ve ni verás

Una vez encontrada la primera vulnerabilidad, trataremos de explotarla inyectando SQL en el parámetro. Recordemos que una forma fácil de comprobar la existencia de un parámetro vulnerable es añadirle "AND 1=1" y "AND 1=0". En el primer caso debería comportarse de forma normal y en el segundo caso la condición es siempre falsa, por lo que normalmente no devolverá nada o tomará un valor por defecto.

Llegados a este punto, podemos intentar la unión de los resultados con otra sentencia SELECT con los resultados que queramos. Usaremos un valor inexistente del campo "id" (por ejemplo, el cero) para asegurarnos que el único resultado de la unión sea el valor que hayamos inyectado.

```
http://retohacking2.elladodelmal.com/ZonaPrivada/descargar.aspx?id=0
```

Podemos probar sustituyendo el parámetro "id" de la URL anterior con las siguientes consultas de unión:

```
id=0 UNION SELECT 0
id=0 UNION SELECT 1
id=0 UNION SELECT null
id=0 UNION SELECT 'a'
```

Comprobamos que en el primer caso intenta descargarse un fichero con el nombre "0". En el segundo caso, intentará descargar un fichero con nombre "1" (y no el fichero con la id=1). En el tercer caso, la aplicación descargará un fichero con nombre "descargar.aspx". Por último, en el cuarto caso, aparecerá un mensaje de *"Fichero no encontrado"*

Vamos a analizar los resultados, comprobando el contenido de cada uno de los ficheros descargados. Observamos que todos ellos tienen el mismo contenido: el texto HTML mostrándonos el típico mensaje de error de ASP.NET.

No obstante, todo indica que la página de descargas es capaz de volcar el contenido de cualquier fichero, pero a su vez parece que está filtrando las cadenas de texto (falla ante cualquier comilla simple). Entonces, ¿cómo nos podemos descargar el fichero que queramos? Pues en realidad basta con hacer la unión con una tabla en la que tengamos posibilidad de escribir y que conozcamos o cuyo nombre sea fácilmente deducible. En este caso, la tabla se llama "Comentarios", el campo de texto se llama "Comentario" y está identificado por el campo "idComentario".

El procedimiento es sencillo: añadimos un recordatorio para un evento cualquiera, e introducimos la ruta completa y el nombre del fichero que queramos descargar en el comentario. Nos vamos a "Ver recordatorios" y comprobamos el identificador autonumérico que se le ha asignado a nuestro comentario. Una vez hecho esto, simplemente añadimos la siguiente instrucción en el parámetro:

```
id=0 UNION SELECT Comentario FROM Comentarios WHERE idComentario=XXX
```

Bien, ahora ya podemos descargarnos algunos ficheros del servidor comprometido (según cómo tenga configurados los permisos), pero en realidad ¿nos hemos parado a pensar qué fichero queremos? Podemos bajarnos los ASPX y muchos otros ficheros de la aplicación y del sistema operativo, pero no nos serán de gran utilidad. Hay que buscar algo más…

Recordemos que el reto consiste en administrar el sitio, pero en realidad tampoco sabemos desde dónde se hace esto. Podríamos deducir que existen roles y se accede desde la misma zona privada, pero descartamos esta opción tras analizar el fichero "web.config". Haciendo pruebas con nombres de ficheros y URLs típicas, al final descubrimos que existe una URL predecible que nos solicita una contraseña:

```
http://retohacking2.elladodelmal.com/ZonaPrivada/admin
```

Las credenciales se enviarán utilizando el método de autentificación básica, por lo que tratándose de un servidor Windows con IIS 6, podemos deducir que el usuario que habrá que introducir debe existir en el sistema operativo y no en una base de datos o en un fichero de configuración.

Lo que ya sabes que está donde siempre

Lo que está donde siempre es precisamente el fichero donde se encuentran todos los usuarios locales del sistema operativo en cuestión. Se trata del fichero SAM (Security Account Manager), que forma parte del registro de Windows y almacena toda la información que necesitamos para superar la última fase. Bueno, en realidad no es suficiente con este fichero, porque parte de dicha información se almacena cifrada con una clave (SysKey), que también se encuentra almacenada en el registro de Windows, pero en el fichero SYSTEM.

Los nombres de usuario están almacenados en el fichero SAM en forma de texto en claro, pero no así las contraseñas, ya que a pesar de estar cifradas con la SysKey, son simples hashes de la contraseña original.

¿Pero dónde se encuentran estos dos ficheros? En una instalación típica, en el directorio *"c:\windows\system32\config"*, pero esto nos servirá de poco ya que estos ficheros siempre están en uso por el propio sistema operativo y, por tanto, no podremos acceder a ellos directamente, que es precisamente lo que estamos buscando. Hay que acceder a la copia de seguridad, que se encuentra en *"c:\windows\repair"*

Juntando todas las piezas

Ahora que sabemos los ficheros que necesitamos, cómo conseguirlos y dónde introducir las credenciales, ya podemos rematar el trabajo.

Añadiremos dos recordatorios, introduciendo en los comentarios los textos *"c:\windows\repair\SAM"* y *"c:\windows\repair\SYSTEM"*, respectivamente. Luego obtenemos sus identificadores asociados y los introducimos en la sentencia de unión SQL descrita anteriormente y descargamos ambos ficheros.

Una vez obtenidos los ficheros, usamos una herramienta como SamInside (www.insidepro.com) para extraer la información almacenada (usuarios y hashes). Asumiendo que no tenemos una licencia completa de SamInside, usaremos otras herramientas como Caín (www.oxid.it), o www.plain-text.info para obtener la contraseña mediante un ataque a la hash del usuario en cuestión. En caso de existir, optaremos siempre por la hash NT, ya que es mucho más débil que la LM.

No diré cual es el nombre del usuario que necesitaremos para acceder a la administración del sitio web, ya que es trivial deducirlo teniendo en nuestro poder el fichero SAM. Lógicamente, tampoco revelaré su contraseña, lo siento… ;-)

Una vez introducidas las credenciales, nos encontramos con una página que nos indica que no se permite el listado de dicho directorio, por lo que en realidad nos queda un último paso más: deducir el nombre del fichero concreto de la página de inicio. Esto también os lo dejo a vosotros. Ánimo, que tampoco necesita de mucha imaginación.

Otra solución: Un atajo para descargarnos los ficheros

La versión inicial del reto permitía ver los comentarios de los demás de una forma trivial (modificando el parámetro *id* de la página de comentarios, parámetro que además es secuencial y consecutivo). Pensé que esto podía dar demasiadas pistas al resto de concursantes, por lo que me planteé otra forma de obtener los ficheros sin que otros pudieran aprovecharse de ello y la encontré.

Solamente hay que obtener el valor ASCII de cada uno de los caracteres de la ruta y formar una cadena como la que muestro a continuación (teniendo en cuenta que hay que codificar el carácter "+" en su forma URL, es decir, "%2b"):

```
http://retohacking2.elladodelmal.com/ZonaPrivada/descargar.aspx?
   id=0 UNION SELECT char(99)+char(58)+char(92)+char(119)+char(105)+…
```

De esta forma tan sencilla de inyectar una cadena sin utilizar las comillas nos evitamos muchos pasos en el proceso descrito antes, como habréis podido comprobar. El resto de pasos, lógicamente son los mismos.

Comentarios y agradecimientos

En el último punto hablaba de la versión inicial del reto. Algunos os estaréis preguntando por qué existieron varias versiones y me parece justo aclararlo. Los primeros días no funcionaba la página de descargas, por lo que no era posible solucionar el reto por un problema de permisos mal configurados. No obstante, esto no era motivo suficiente para modificar el código de la aplicación.

En realidad el motivo fue otro, ya que el primer día encontré otros fallos de seguridad más importantes como la posibilidad de utilizar herramientas que permitían obtener toda la estructura e incluso los datos de la base de datos, incluyendo por ejemplo la tabla de participantes y la tabla de ganadores (en la que podría estar la solución detallada).

Ese día le mandé un e-mail a Chema Alonso detallando el problema que veía, así como el tema de poder ver los comentarios de los demás. Me confirmó que no estaba previsto y que se solucionaría. Por este motivo, el reto estuvo offline durante varias horas y se modificaron todos estos puntos. Aun así, seguía existiendo un fallo que impedía dar con la solución y que no se corrigió hasta un par de días más tarde (y además sin avisar). El fichero SAM no era el correcto y el fichero SYSTEM no se podía descargar. Una vez corregido, aproveché para comentar en el blog que había un fallo.

A pesar de todos los fallos iniciales, hay que reconocer el trabajo que hay detrás del reto, ya que su diseño y su programación indudablemente han costado tiempo y su alojamiento online también tiene su coste y sus posibles riesgos. Por todo ello, quiero terminar este documento agradeciendo el trabajo de **Chema Alonso** y de todo el equipo de **Informática64** que nos ha hecho posible participar en el mismo. Y por supuesto, esperando que te haya sido útil y didáctico, gracias a ti que estás leyendo esto.

Saludos,

Daniel Kachakil
dani@kachakil.com