

Maestría en Ciberseguridad

Documento final de Proyecto de Investigación Aplicada 2

Evaluación de vulnerabilidades en sitios web del gobierno central de Costa Rica a través de herramientas especializadas en el análisis de pruebas de penetración y técnicas de hacking.

Presentado por:

Calderón Miranda, Andrey

Méndez Castro, Gustavo

Marzo, 2018

DERECHOS DE AUTOR

El presente trabajo de investigación titulado: ***"Evaluación de vulnerabilidades en sitios web del gobierno central de Costa Rica a través de herramientas especializadas en el análisis de pruebas de penetración y técnicas de hacking, que contribuirán en construir un plan de recomendaciones de ciberseguridad para los encargados de gestionar los sitios web"*** es original y elaborado con fines académicos, orientado a colaborar con el gobierno de Costa Rica a través del MICITT, para efectos legales el contenido del mismo es de orden confidencial y no podrá ser reproducido por ningún medio ni en ningún formato por el lector, sin expresa autorización previa escrita por sus autores, excepto aquellas copias que sean necesarias realizar para fines académicos. El tiempo de duración es de 5 años a partir del momento que sea recibido el visto bueno del tribunal evaluador indicado más adelante en este documento.

En caso de que sea necesaria una reproducción parcial o total del documento, con previo consentimiento de sus autores, se deberá hacer expresa mención a la propiedad intelectual de sus autores sobre la información contenida en ellas.

San José, 9 de marzo, 2018

Gustavo Méndez Castro | Andrey Calderón Miranda

AGRADECIMIENTO

A mi familia, especialmente a mi esposa, por su apoyo y compresión.

A mi compañero de tesis Andrey por todo el esfuerzo, compromiso y dedicación en este proceso.

Al Profesor Cesar Rodriguez, por la guía a lo largo de toda esta investigación.

A Johnny Pan por abrirnos las puertas en el MICITT para realizar este trabajo.

Gustavo Méndez Castro

Primero quiero dar gracias a Dios y a mi esposa por haber sido el principal apoyo durante tantas noches de estudio, también a Gustavo compañero de tesis que siempre dedicó el tiempo, esfuerzo y las ganas para que sacáramos la tarea.

Quiero agradecer a nuestro tutor de tesis, el Profesor César Rodríguez por su buena voluntad y guía durante estos meses de trabajo.

Finalmente quiero dar las gracias a Johnny Pan, por darnos la oportunidad de llevar a cabo nuestro proyecto en el MICITT y por la colaboración directa que sostuvo con los distintos ministerios involucrados.

Andrey Calderón Miranda

APROBACIÓN DEL TRIBUNAL EXAMINADOR

TABLA DE CONTENIDO

DERECHOS DE AUTOR iii

AGRADECIMIENTO iv

APROBACIÓN DEL TRIBUNAL EXAMINADOR ii

TABLA DE CONTENIDO iii

TABLA DE FIGURAS viii

RESUMEN EJECUTIVO xii

INTRODUCCIÓN 2

1.1 Generalidades 3

1.2 Antecedentes 3

1.3 Definición y descripción del problema 4

1.4 Justificación 4

1.5 Viabilidad 5

1.5.1 Punto de Vista Técnico 5

1.5.2 Punto de Vista Operativo 5

1.5.3 Punto de Vista Económico 6

1.6 Objetivos 7

1.6.1 Objetivo General 7

1.7 Alcances y limitaciones 8

1.7.1 Alcances....8
1.7.2 Limitaciones....9
1.8 Marco de referencia Organizacional y Socioeconómico....9
1.8.1 Historia....9
1.8.2 Misión, Visión....11
1.8.3 Políticas Institucionales....11
MARCO TEÓRICO....16
2.1 Antecedentes investigativos....16
2.2 Fundamentación legal....16
2.3 Evaluación de vulnerabilidades en la Web....17
2.4 Tipos de Pruebas de penetración....19
2.6 Componentes de un sistema informático....21
2.7 Gestión del ciclo de vida de las vulnerabilidades....24
2.8 Estrategia para el desarrollo de la propuesta....26
2.8.1 Open Web Application Security Project (OWASP)....26
1. Pruebas para recopilación de información....28
2. Pruebas de gestión de la configuración y la implementación....31
3. Pruebas de gestión de identidad....32
4. Pruebas de autenticación....33
5. Pruebas de autorización....37

6. Pruebas de gestión de sesión 39
7. Pruebas de validación de datos de entrada 39
8. Manejo de errores 44
9. Criptografía 45
10. Pruebas de lógica de negocio 46
11. Prueba del lado del cliente 47
MARCO METODOLÓGICO 49
3.1 Tipo de investigación 49
3.2 Alcance investigativo 49
3.3 Enfoque 49
3.4 Diseño 50
3.5 Población y Muestreo 50
3.6 Instrumentos de Recolección de Datos 50
3.7 Técnicas de observación 51
3.8 Técnicas de Análisis de la Información 51
DESARROLLO DE LA PROPUESTA DE SOLUCIÓN 53
4.1 Estructura de los sitios web 53
4.2 Identificar la estructura de los sitios web 58
4.3 Identificar a los interesados 59
4.4 Herramientas utilizadas y método de análisis 61

4.5 Justificación de las herramientas 66

4.6 Interpretación de las pruebas 69

4.6.1 Desarrollo de la metodología para pruebas de penetración de sitios web 69

4.6.2 Calificación de Riesgo 70

4.6.3. Análisis de la situación al realizar las pruebas de penetración 72

4.6.4 Vulnerabilidades detectadas en todos los sitios web analizados 107

4.7 Análisis de la situación al momento de realizar las pruebas por cada Ministerio 109

4.7.1 Instituto Mixto de Ayuda Social (IMAS) 110

4.7.2 Instituto Costarricense de Turismo (ICT) 115

4.7.3 Ministerio de Trabajo y Seguridad Social 122

4.7.4 Ministerio de Ambiente y Energía (MINAE) 130

4.7.5 Ministerio de Hacienda 133

4.7.6 Ministerio de Ciencia, Tecnología y Telecomunicaciones (MICITT) 137

4.7.7 Ministerio de Seguridad Pública 139

4.7.8 Ministerio de Gobernación y policía 143

4.7.9 Ministerio de Justicia y Paz 148

4.7.10 Ministerio de Cultura y Juventud 151

4.7.11 Ministerio de Comercio Exterior (Comex) 154

4.7.12 Instituto Nacional de las Mujeres (INAMU) 160

4.7.13 Ministerio de Vivienda y Asentamientos Humanos (MIVAH) 165

4.7.14 MIDEPLAN 166
4.7.15 Ministerio de la Presidencia 171
4.7.16 Ministerio de Educación Pública (MEP) 175
4.8 Informe completo de las vulnerabilidades encontradas 180
CONCLUSIONES 186
RECOMENDACIONES 188
BIBLIOGRAFÍA 190
GLOSARIO 193
ANEXOS 197
Anexo 1 197

TABLA DE FIGURAS

Figura 1. Mapa del sitio web del Comex 54
Figura 2. Lista de interesados en cada Ministerio 60
Figura 3. Escala de puntuación de riesgo 72
Figura 4. Total de vulnerabilidades por cada categoría 72
Figura 5. Total de vulnerabilidades por nivel de riesgo 73
Figura 6. Diagrama básico de Web Application Firewall 97
Figura 7. Total de vulnerabilidades detectadas por nivel de riesgo en todos los sitios web ... 107
Figura 8 Total de vulnerabilidades detectadas por nivel de riesgo en todos los sitios web 108
Figura 9. Vulnerabilidades por categoría en cada Ministerio 108
Figura 10. Posibilidad de ingreso al sistema con SQL injection en IMAS 111
Figura 11. Revelación de información sensible en IMAS 112
Figura 12. Versión de drupal 112
Figura 13. Ingreso al sitio en HTTP IMAS 113
Figura 14. Lista de archivos expuesto del IMAS 114
Figura 15. Lista parcial de usuarios del IMAS 115
Figura 16. Direcciones encontradas de métodos inseguros 117
Figura 17. Creación de carpetas en el sitio del ICT 118
Figura 18. Formulario de acceso con HTTP en ICT 119
Figura 19. Archivos expuestos del ICT 119
Figura 20. Mensaje de error en ICT 120
Figura 21. Cifrado medio en ICT 120
Figura 22. Respuesta de DNS 121

Figura 23. POP3 habilitado 121
Figura 24. Anuncio de bienvenida del FTP 121
Figura 25. SSLv3 habilitado 124
Figura 26. Ingreso al sistema con texto plano MTSS 124
Figura 27. Listado de usuarios permite adivinar contraseñas 125
Figura 28. Página de prueba del MTSS en producción 126
Figura 29. Ingreso al sistema con usuario administrador en MTSS 126
Figura 30. Conexión válida para modificar información 127
Figura 31. Consulta realizada tipo cálculo 127
Figura 32. Catálogo de gestiones 128
Figura 33. Cifrado medio 128
Figura 34. Bloques de cifrado 129
Figura 35. Versión de Microsoft 129
Figura 36. Mensaje de bienvenida FTP 129
Figura 37. Revela la versión 130
Figura 38. Sitio del MINAE Hackeado 131
Figura 39. Amplificación de DNS 132
Figura 40. Lista de archivos expuestos 132
Figura 41. Mensaje de bienvenida del FTP 133
Figura 42. Versión de Joomla 134
Figura 43. Formulario de acceso con HTTP en Hacienda 135
Figura 44. XAMPP expuesto en Hacienda 136
Figura 45. Cifrado medio 137
Figura 46. Aplicación web desactualizada en MICITT 138

Figura 47. Archivos expuestos 139
Figura 48. DNS expuesto debido a la versión del servidor 141
Figura 49. Formulario de acceso en HTTP MSP 141
Figura 50. Archivos expuestos 142
Figura 51. Versión de Microsoft 143
Figura 52. Versión de WordPress desactualizada 145
Figura 53. Soporta cifrado SSLv3 146
Figura 54. No utiliza HTTPS 146
Figura 55. Instancias de SQL expuestas 147
Figura 56. Listado de directorios 149
Figura 57. Archivos expuestos 150
Figura 58. Directorio /bin expuesto 151
Figura 59. Versión de Microsoft 151
Figura 60. Método TRACE habilitado 152
Figura 61. Archivos expuestos 153
Figura 62. Página de prueba en Ministerio de Cultura 153
Figura 63. Redirecciona a otro sitio al usar HTTPS 154
Figura 64. Cifrado SSLv2 y SSLv3 156
Figura 65. Acceso remoto al servidor de producción del Comex 157
Figura 66. Mensaje de error que revela datos sensibles 158
Figura 67. Página de prueba del Comex en producción 158
Figura 68. ASP.NET expuesto 159
Figura 69. Carga de archivos sin validación en Comex 159
Figura 70. Versión de GlassFish desactualizada 161

Figura 71. Ingreso al sito sin uso de HTTPs.....162
Figura 72. Amplificación de DNS.....162
Figura 73. Base de datos de empleados del INAMU.....163
Figura 74. Lista de direcciones que contiene el sitio de la INAMU.....163
Figura 75. Información de usuario enviando en texto plano.....164
Figura 76. Listado de directorios.....166
Figura 77. Listado de directorios MidePlan.....167
Figura 78. Ingreso al sitio Mideplan con HTTP.....168
Figura 79. Exposición de WSDL.....169
Figura 80. Archivos expuestos.....170
Figura 81. Mensaje de error en la aplicación.....170
Figura 82. Listado directorio Presidencia.....172
Figura 83. Cifrados SSL.....173
Figura 84. Gestión de archivos.....173
Figura 85. Cifrado medio.....174
Figura 86. Cifrado medio.....174
Figura 87. Uso de HTTP para acceso al MEP.....176
Figura 88. Puede validar usuarios disponibles en MEP.....177
Figura 89. Archivos expuestos.....177
Figura 90. Archivos de configuración visibles en MEP.....178
Figura 91. Mensaje de actualización de Drupal.....179
Figura 92. Changelog.....179

RESUMEN EJECUTIVO

El activo más importante de las organizaciones hoy en día es la información, y se hace cada vez más difícil que puedan garantizar la protección de los datos de sus clientes. Esta investigación tiene como propósito abordar desde una perspectiva ética, la búsqueda y análisis de vulnerabilidades web en algunos de los sitios del gobierno central de Costa Rica. El principal objetivo es realizar una evaluación que permita entender la situación del país en materia de seguridad web y de las vulnerabilidades existentes al momento del análisis, que puedan de una u otra forma dañar la imagen de Costa Rica, tanto a lo interno como fuera de sus fronteras. Además se pretende con esta evaluación llevar a cabo un análisis por cada sitio web, que permita comprender cuáles son los principales riesgos a los que se enfrenta el país en materia de seguridad, para así establecer, con base en el criterio de los estudiantes, algunas recomendaciones que contribuyan a solucionar los problemas existentes, ya sea en un corto o mediano plazo según lo decida cada encargado de la gestión de estos sitios.

También se pretende llevar a cabo un estudio a profundidad utilizando tanto herramientas de penetración y vulnerabilidades como métodos manuales y técnicas de hackeo, que permitan realizar un análisis más exhaustivo y detallado de los riesgos asociados a cada sitio. Esto último con el apoyo del MICITT y la debida autorización de los departamentos de TICs de cada ministerio.

Finalmente se espera obtener un reporte, que es un extracto de este documento, de forma tal que pueda ser entregado al MICITT, para que posteriormente se encarguen de distribuirlo a las personas correspondientes, se analicen las vulnerabilidades y se tomen las acciones correctivas pertinentes.

CAPÍTULO I

INTRODUCCIÓN

La tecnología ha evolucionado a pasos agigantados en las últimas décadas y somos testigos de este desarrollo, todos los días nos vemos envueltos por la tecnología y en ocasiones ni nos damos cuenta. De igual forma los grandes desarrollos tecnológicos ayudan a las empresas a desempeñar un papel más competitivo en los mercados, pero para que las empresas puedan alcanzar el éxito a través de la tecnología requieren una gran cantidad de expertos en las distintas áreas de Tecnologías de Información y Comunicación (TICs), incluida la ciberseguridad, que contribuyan a facilitar esta labor, buscando siempre alinear los objetivos estratégicos de la organización con los de TICs. Lo anterior por supuesto requiere un gran compromiso e inversión por parte de las empresas.

Al conocer de la importancia que tienen las tecnologías hoy en día y lo vulnerables que se vuelven los sistemas informáticos, es que surge la idea de proponer este proyecto, el cual pretende llevar a cabo una evaluación de las vulnerabilidades existentes en los sitios web, que contribuya a comprender la importancia que tiene la ciberseguridad para el país.

La detección de vulnerabilidades basadas en las técnicas de ethical hacking, así como de pruebas de penetración, permite salvaguardar los sistemas de información de cualquier amenaza ya sea externa o interna, a la que se encuentra expuesta una organización pública o privada. Como consecuencia de esa detección temprana es posible tomar las medidas necesarias de forma preventiva y correctiva que ayuden a reducir el riesgo y minimizar el impacto asociado.

1.1 Generalidades

El fin del presente trabajo de investigación es brindar una visión detallada en relación con el estado actual de la seguridad en los principales sitios web del gobierno central de Costa Rica. Se tiene el apoyo del Ministerio de Ciencia, Tecnología y Telecomunicaciones MICITT para llevar a cabo los estudios requeridos.

1.2 Antecedentes

Los cimientos del Ministerio de Ciencia, Tecnología y Telecomunicaciones se encuentran en la LEY DE PROMOCIÓN DEL DESARROLLO CIENTÍFICO Y TECNOLÓGICO, LEY No. 7169 y sus reformas, vigente desde el primero de agosto de mil novecientos noventa, la cual resume en sus artículos primero, segundo, tercero y veinte el propósito u objetivo general, su objetivo de largo plazo, sus objetivos específicos y atribuciones, respectivamente.

Las organizaciones han tenido que evolucionar y ver más allá del presente. Los ataques a los sistemas, el robo de información, sustitución o suplantación de identidad, el hackeo de sitios web, el acceso indeseado a bases de datos, la utilización de la información con fines terroristas, la corrupción, el lavado de dinero, el fraude, estafas entre muchos otros crímenes ha permitido ver la necesidad en los gobiernos e instituciones de establecer medidas preventivas que ayuden a salvaguardar el activo de la información. Anteriormente las pequeñas empresas o países veían como imposible el ser víctimas de estos delincuentes, tenían la idea errónea de “mi empresa no es atractiva”, sin embargo, a estos individuos el tamaño no les es tan relevante, lo que importa para ellos es lograr su objetivo de hacer un daño; es así como podemos ver la necesidad de prevenir y establecer medidas que disminuyan el riesgo de penetración a sus sistemas y ser vulnerados.

1.3 Definición y descripción del problema

El acelerado crecimiento en las comunicaciones y la accesibilidad a la información permiten que el uso de internet sea una herramienta de uso diario para muchas personas, y por lo tanto que la exploración de los sitios de interés de las personas sea mucho más sencillo. Por esta razón cada día se ve un aumento de crímenes cibernéticos, las amenazas crecen exponencialmente conforme avanzamos a través de los años. El conocimiento que una persona puede obtener por medio del mismo internet y de la gran cantidad de recursos existentes que revelan métodos sencillos de ejecutar un ataque, contribuyen a que cualquier persona, con o sin experiencia, pueda causar un gran efecto en un sistema informático, se convierten así en la mayor amenaza a la que todos nos debemos enfrentar, personas, instituciones, empresas, gobiernos etc.

1.4 Justificación

Con el presente estudio se pretende alcanzar un objetivo, evaluar el estado actual a nivel de seguridad de los sitios web del gobierno central de Costa Rica. Se busca enfocar los esfuerzos en determinar cuáles son las necesidades inmediatas de mejora que deben atenderse con el único fin de proteger la seguridad de los sistemas y su información.

En Costa Rica no existe un organismo regulador que exija a los distintos ministerios y entidades de gobierno cumplir con una normativa de seguridad y estándares para el desarrollo de sistemas, sitios web, hosting de aplicaciones, estándares en el uso de hardware y software. Esto potencia que se cometan errores al seguir prácticas inadecuadas que vulneran el correcto funcionamiento de las plataformas web.

Por lo anterior, es una necesidad inmediata contar con un estudio que permita entender el estado actual de Costa Rica en materia de Ciberseguridad en sus sitios de gobierno, debido a que son uno de los principales medios de consulta que el país ofrece a los diversos usuarios locales y extranjeros, por lo que el identificar las potenciales vulnerabilidades y riesgos serán claves que

contribuirán en alertar y actuar en contra de las constantes amenazas que se pueden materializar a través del tiempo, afectando la imagen de la República y la seguridad de la información contenida en sus sitios web.

Al finalizar el estudio se quiere dar a conocer al MICITT las necesidades existentes de mejora, para que posteriormente procedan a tomar las acciones preventivas y correctivas que han sido recomendadas en esta investigación.

1.5 Viabilidad

1.5.1 Punto de Vista Técnico

Lo que se evalúa en este proyecto es el aspecto a nivel de las vulnerabilidades en las páginas web del gobierno central de Costa Rica; el proyecto se enfoca en detectar las vulnerabilidades existentes, por lo tanto, no se pretende implementar cambios ni realizar modificaciones en los equipos y/o sistemas. Las herramientas que se propone utilizar han sido utilizadas en la universidad o en el área laboral, por lo tanto, se tiene un gran manejo técnico para poder interpretar los resultados y obtener el mayor provecho de dichas herramientas.

Por lo tanto, desde el punto de vista técnico es factible poder desarrollar este proyecto pues se colabora de forma directa aportando el conocimiento y aplicándolo de forma práctica con la información que se pueda recopilar y el resultado que se estará entregando. Este resultado mejorará las condiciones de seguridad de acuerdo con las acciones posteriores que se puedan tomar de las páginas del gobierno de Costa Rica.

1.5.2 Punto de Vista Operativo

Este proyecto se realiza en coordinación con el director del Equipo de Respuesta a Incidentes de Seguridad (CSIRT, por sus siglas en inglés Computer Security Incident Response Team) dentro del MICITT para poder ejecutar las pruebas de penetración (pentesting) en las páginas web. Se pretende establecer una comunicación oficial directamente del MICITT hacia cada uno

de los ministerios involucrados para obtener su aprobación, la cual es responsabilidad de los jefes de TI. Las horas en las que se realizarán las pruebas serán dispuestas por cada ministerio de acuerdo con lo que ellos consideren adecuado, lo anterior con el fin de no afectar la productividad de los empleados o bien de los usuarios finales de las páginas web.

Las pruebas por realizar no pretenden impactar en una degradación del servicio o bien afectar la funcionalidad de los distintos sitios, por lo tanto aquellas que puedan resultar intrusivas serán coordinadas en horas fuera del horario de producción.

1.5.3 Punto de Vista Económico

El proyecto por desarrollar requiere del uso de herramientas que permitan facilitar la labor de los investigadores, debido a la naturaleza de esas herramientas, la inversión económica en la que se debe incurrir se reduce al hecho de contar con acceso a internet y software de acceso libre que permita llevar a cabo la labor de investigación. Por otra parte, se tiene el apoyo del MICITT, debido a esto, si por alguna razón bien justificada fuera necesario contar con recursos económicos para el desarrollo de este proyecto, se requiere llevar el proceso de solicitud al departamento correspondiente, sin embargo, no es una prioridad llegar a ese punto.

El acceso a las herramientas de pruebas y evaluación de los sitios web serán proporcionados por los investigadores de este proyecto, y por lo tanto no existe una dependencia económica directa que los vincule con el desarrollo de dicha investigación.

Conociendo los hechos anteriormente mencionados, se concluye que la viabilidad de este proyecto a nivel económico es posible, la profundidad requerida se podrá complementar haciendo uso de diferentes herramientas, así como pruebas manuales que ayuden a dar una visión más amplia del estudio.

1.6 Objetivos

En este proyecto se ha utilizado, para plantear los objetivos, la taxonomía de Bloom, basada en un orden jerárquico por áreas de conocimiento.

1.6.1 Objetivo General

Evaluación de vulnerabilidades en sitios web del gobierno central de Costa Rica a través de herramientas especializadas en el análisis de pruebas de penetración y técnicas de hacking, que contribuirán en construir un plan de recomendaciones de ciberseguridad para los encargados de gestionar los sitios web.

1.6.1 Objetivos específicos

- Conocer a profundidad la estructura de los diferentes sitios web del gobierno central de Costa Rica para determinar la complejidad de cada uno.
- Identificar la infraestructura en la que se encuentra construido cada sitio web para obtener la tecnología utilizada, así como su posible ubicación física.
- Determinar la estructura, composición y personal a cargo que brinda el soporte a cada sitio web para comprender la importancia que se da al mantenimiento y controles de seguridad.
- Escoger las diferentes herramientas y métodos de análisis que serán utilizadas para detectar las vulnerabilidades existentes en los sitios web del gobierno central de Costa Rica.
- Justificar las razones por las que se utilizarán las herramientas escogidas para resaltar la relevancia que toman en esta evaluación.
- Ejecutar las distintas pruebas de penetración previamente seleccionadas utilizando las herramientas y métodos definidos para obtener los resultados deseados.
- Interpretar los resultados obtenidos en las pruebas para posteriormente obtener un informe detallado de cada sitio.
- Comparar los resultados obtenidos del análisis para brindar las recomendaciones con el criterio de los expertos.

1.7 Alcances y limitaciones

1.7.1 Alcances

El trabajo de tesis está determinado para ser ejecutado en un periodo correspondiente a ocho meses, siendo este el tiempo total de investigación, desarrollo, aporte de resultados y defensa, por lo tanto, es importante tomar en consideración este factor para definir el alcance del mismo. La metodología de investigación evaluativa se pretende llevar a cabo siguiendo la guía de Web Application Security Testing del Testing Guide 4.0 de OWASP, por lo que se busca obtener una serie de evidencias y documentación que permita entender las vulnerabilidades existentes o potenciales riesgos que existen en los principales sitios web del gobierno central de Costa Rica.

El alcance de la investigación estará limitado a la evaluación respectiva que se realice en los dieciséis sitios de gobierno previamente seleccionados, enfocado en algunas de las secciones de uso común para los distintos usuarios, administrador, cliente y las interacciones regulares que se dan entre la aplicación y el navegador de internet, por lo que es de mucha importancia tener un buen entendimiento de la función que se desarrolla en cada sitio.

Las fechas en las que se realicen dichas pruebas están delimitadas a partir de la aprobación de cada ministerio, previamente coordinado con el MICIT, el rango aproximado va de octubre a diciembre del 2017, por tanto los resultados aquí mostrados pueden no representar el estado actual.

Debido a la gran variedad de plataformas, sistemas operativos, lenguajes de programación y gestores de contenido, entre otros, únicamente se pretende realizar pruebas de uso conocido, como por ejemplo pruebas de cross site scripting, pruebas de inyección de SQL y pruebas de validación de usuarios, entre otras, que buscan determinar la existencia de vulnerabilidades en algunas de las secciones del sitio y que consecuentemente nos ayuden a concluir que existe una

alta posibilidad de que existan vulnerabilidades similares. Las buenas prácticas nos indican que esas vulnerabilidades no deberían de existir en otras partes del sitio.

El resultado de la investigación será de uso exclusivo del MICITT debido a la naturaleza del mismo, ya que se pueden exponer las vulnerabilidades existentes en los sitios y como consecuencia eso podría poner en riesgo la estabilidad de los mismos si se llegan a saber por un tercero.

1.7.2 Limitaciones

El desarrollo de este proyecto está basado en el análisis de los sitios web en producción, debido a que no se cuenta con un laboratorio de pruebas o ambientes de desarrollo en donde se puedan valorar los escaneos y exploraciones manuales. Por tal motivo, las pruebas que puedan resultar intrusivas o bien que puedan degradar la calidad del servicio se ejecutarán durante las horas específicas previamente coordinadas con los distintos gerentes de TI en cada uno de los ministerios involucrados, por medio de una autorización que se canalizará a través del MICITT.

Los hallazgos encontrados darán como resultado la obtención de las recomendaciones de los expertos, sin embargo, esas recomendaciones no serán implementadas como parte del desarrollo de este proyecto, sino que serán una guía que permita a los distintos departamentos involucrados tomar las acciones correctivas a futuro según lo disponga cada una de esas gerencias.

1.8 Marco de referencia Organizacional y Socioeconómico

1.8.1 Historia

En Costa Rica, anteriormente las funciones de controlar la parte relacionada a la tecnología era una función del MINAE, pero posteriormente se reformó ese ministerio y las funciones pasaron a formar parte de lo que actualmente conocemos como el MICITT. No existen en el país regulaciones, estándares o buenas prácticas que se deban seguir por ley para el desarrollo de sus

principales sitios web, y por lo tanto no existe un marco de referencia que se pueda acotar en este apartado en relación con cómo se han implementado en el pasado.

A continuación, un extracto de la ley 9046 que traslada las funciones del MINAE al MICITT en materia de tecnología y telecomunicaciones. Este es el órgano rector en Costa Rica de todo lo relacionado en materia de tecnologías de información, el cual resulta de mucho interés de acuerdo con el enfoque de este proyecto.

El nombre del Ministerio fue primeramente reformado por el artículo 116 de la Ley N.º 7554 de 4 de octubre de 1995, Ley Orgánica del Ambiente y posteriormente por el artículo 48, aparte a) de la Ley N° 8660 del 8 de agosto de 2008, Ley de Fortalecimiento y Modernización de las Entidades Públicas del Sector Telecomunicaciones, que reformó el artículo 23 inciso h) de la Ley General de la Administración Pública, el cual enumera las carteras ministeriales, siendo que el nombre correcto es MINISTERIO DE AMBIENTE, ENERGÍA Y TELECOMUNICACIONES.

Artículo 1.- El Ministerio de Industria Energía y Minas se transformará en Ministerio del Ambiente y Energía(**), y asumirá, en este campo, además de las actuales responsabilidades de aquel, las que la presente ley le asigne. El ministro será el rector del sector Recursos Naturales, Energía y Minas.

(*) (El nombre del Ministerio fue así reformado el artículo 48 aparte a) de la Ley N° 8660 del 8 de agosto de 2008, Ley de Fortalecimiento y Modernización de las Entidades Públicas del Sector Telecomunicaciones.)

(**)(Modificada su denominación por el artículo 11 de la Ley "Traslado del sector Telecomunicaciones del Ministerio de Ambiente, Energía y Telecomunicaciones al Ministerio de Ciencia y Tecnología, N° 9046 del 25 de junio de 2012)

Ley y Promoción del Desarrollo Científico y Tecnológico No 7169

Título I Principios, Capítulo I Objetivos, Artículo 2.

c) Establecer estímulos e incentivos para los sectores privado, público y las instituciones de educación superior universitaria y otros centros de educación, con la finalidad de que incrementen la capacidad de generar ciencia y tecnología y de articularse entre sí.

g) Fomentar todas las actividades que apoyan el desarrollo científico y tecnológico sustantivo, los estudios de postgrado y la capacitación de recursos humanos, el mejoramiento de la enseñanza de las ciencias, las matemáticas y la educación técnica, lo mismo que la documentación e información científica y tecnológica.

1.8.2 Misión, Visión

Misión

"Dictar la política pública de ciencia, tecnología y telecomunicaciones, que permita al país potenciar el aprovechamiento del conocimiento y la innovación, para priorizar y dirigir las iniciativas del sector hacia la competitividad, el bienestar y la prosperidad."

Visión

"Ser el propulsor de un modelo país de largo plazo basado en el conocimiento y la innovación para alcanzar un desarrollo integral y sustentable con liderazgo global."

1.8.3 Políticas Institucionales

1. Fortalecer las capacidades nacionales de ciencia, tecnología e innovación a través del aumento en la inversión en investigación y desarrollo (I+D), integrando la acción los sectores académico, privado y gubernamental, para la transición hacia una economía basada en el conocimiento.

2. Incentivar la formación de recursos humanos de alto nivel en las áreas científico-tecnológicas, por medio de la articulación entre la oferta y demanda de profesionales, el

otorgamiento de becas y programas de orientación vocacional temprana, para satisfacer las necesidades de los sectores productivos.

3. Potenciar el reconocimiento de la ciencia, tecnología e innovación, por medio de diferentes estrategias de divulgación, para una mayor apropiación por parte de los ciudadanos.

4. Hacer de las telecomunicaciones una fuerza motora para el desarrollo humano de todos los costarricenses, de forma universal y solidaria, a través del desarrollo, implementación y seguimiento de las políticas públicas en telecomunicaciones.

1.9 Estado de la cuestión

Existe en la actualidad una necesidad de regular las políticas referentes al desarrollo de aplicaciones, sitios web, sistemas, etc. Como parte de esta investigación no ha sido posible determinar la existencia de un ente regulador encargado de controlar y auditar lo anteriormente mencionado. Dentro de algunos de los esfuerzos que se están realizando en el país se puede encontrar al CSIRT, que entre algunas de sus funciones tiene como responsabilidad velar por la seguridad de los principales sitios web del gobierno central, considerando que existen un conjunto de amenazas derivadas del uso malicioso de las tecnologías digitales, cuyo fin último es lesionar la integridad individual en favor del crimen organizado. Sin embargo, dicho organismo en la actualidad no tiene un proceso definido que permita al país garantizar que existan buenas prácticas de implementación y desarrollo de tecnologías, y asegurar que se cumplen ciertos lineamientos fundamentales para reducir el riesgo de que existan amenazas que puedan vulnerar los sistemas tecnológicos digitales.

De acuerdo con la Directriz N.º 40, sobre el establecimiento de sitios web en las entidades públicas, y publicada en la Gaceta N° 121 del 23/6/2005, se dice que las instituciones de gobierno que cuentan con presencia en Internet, mediante una directriz emitida por la Presidencia de la República y el Ministerio de Ciencia y Tecnología el 28 de abril del 2005, obliga a las instituciones a tomar las acciones necesarias para que ofrezcan a través de sus sitios web servicios institucionales estratégicos. Estos deben ir de acuerdo con sus prioridades y demanda de servicio de los usuarios, ser interactivos y cumplir con estándares de eficiencia, seguridad y amigabilidad.

Las siguientes leyes forman parte del estado actual en temas de seguridad en Costa Rica.

Ley N° 9048, Reforma de varios artículos y modificación de la Sección VIII, denominada Delitos informáticos y conexos, del Título VII del Código Penal, N° 214 del 6/11/2012 (Alcance N° 172).

Decreto ejecutivo Nº 37052-MICIT, Crea el "Centro de Respuesta de Incidentes de Seguridad Informática" (CSIRT-CR), Nº 72 del 13/4/2012.

Decreto ejecutivo N° 36274-MICIT, Crea la Comisión Nacional de Seguridad En Línea, N° 239 del 9/12/2010.

Decreto ejecutivo N° 35.766-PLAN-G-J, Promoción del Modelo de Interoperabilidad en el Sector Público, N° 41 del 1/3/2010.

Decreto ejecutivo N° 35205-MINAET, Reglamento sobre Medidas de Protección de la Privacidad de las Comunicaciones, N° 94 del 18/5/2009.

Decreto ejecutivo N° 37549-JP, Reglamento para la Protección de los Programas de Cómputo en los Ministerios e Instituciones Adscritas al Gobierno Central, Nº 43 del 1/3/2013.

Ley N° 8968, Ley de protección de la persona frente al tratamiento de sus datos personales, N° 170 del 5/9/2011.

Decreto ejecutivo N° 37554-JP, Reglamento a la Ley de Protección de la Persona Frente al Tratamiento de sus Datos Personales, N° 45 del 5/3/2013 (Alcance N° 42).

Política de apertura de datos

Este compromiso buscará establecer las bases técnicas para los estándares mínimos y lineamientos básicos que las instituciones deben cumplir para abrir datos de las instituciones públicas y ponerlos a disposición permanente de la ciudadanía para que pueda hacer uso de ellos.

Decreto de transparencia y acceso a la información pública

La Constitución Política de Costa Rica garantiza que las personas deben tener libre acceso a la información sobre asuntos de interés público. Sin embargo, todavía es difícil acceder a información pública.

Para evitar esta dificultad, el gobierno de Costa Rica está elaborando un decreto de acatamiento inmediato que garantiza a todos los ciudadanos los mecanismos de transparencia para cumplir con el derecho humano de acceso a la información pública en todas las instituciones públicas.

Proyecto de ley de acceso a la información pública

Costa Rica es uno de los pocos países de Latinoamérica que no tiene una ley que respalda el derecho de la gente a solicitar información. Debido a lo anterior, el Gobierno está comprometido a impulsar un proyecto de ley en la Asamblea Legislativa que promueva su discusión, que sirva para fomentar la cultura de transparencia en la función pública.

CAPÍTULO II

MARCO TEÓRICO

2.1 Antecedentes investigativos

No se ha podido confirmar la existencia en Costa Rica sobre un esfuerzo similar que en conjunto con distintas instituciones hayan llevado a cabo una investigación de este tipo. Por ello, la ejecución de este proyecto va a contribuir en brindar una perspectiva más amplia acerca de la importancia que se le debe brindar a la seguridad de la información y en específico a la parte web, así como a la imagen de un país, más aun conociendo la tendencia en los últimos años de la ciberdelincuencia y sus consecuencias en ocasiones irreparables.

2.2 Fundamentación legal

La creación del Centro de Respuesta de incidentes de Seguridad Informática CSIRT-CR, de acuerdo con la normativa No. 37052 del MICITT, es una necesidad para el país. En respuesta al alto número de incidentes informáticos y el aumento de amenazas cibernéticas, surge como una necesidad inmediata que permita reducir los tiempos de respuesta de forma efectiva en caso de que las entidades gubernamentales sufran de algún tipo de amenaza cibernética que ponga en riesgo la seguridad de los datos. De acuerdo con la normativa, se describe en el artículo 1º las competencias y responsabilidades del centro.

Artículo 1º-Créase el Centro de Respuesta de Incidentes de Seguridad Informática (CSIRT-CR) con sede en las instalaciones del Ministerio de Ciencia y Tecnología, con facultades suficientes para coordinar con los poderes del Estado, instituciones autónomas, empresas y bancos del Estado todo lo relacionado con la materia de seguridad informática y cibernética y concretar el equipo de expertos en seguridad de las Tecnologías de la Información que trabajará para prevenir y responder ante los incidentes de seguridad cibernética e informática que afecten a las instituciones gubernamentales.

2.3 Evaluación de vulnerabilidades en la Web

El principal foco de atención en la presente investigación es determinar cuáles son los factores que contribuyen a la existencia de vulnerabilidades y que representan una brecha de seguridad en los sistemas de información digitales, específicamente en sitios Web. Para poder comprender algunos conceptos claves y llevar esas ideas por el camino apropiado es necesario abarcar una serie de términos que facilitan el entendimiento.

Primero es necesario comprender ¿qué es una vulnerabilidad en tecnologías de información digital? Se puede definir como una debilidad de cualquier tipo que comprometa la seguridad del sistema informático. Por otra parte, se dice que cuando un atacante descubre alguna falla en la planificación, implementación, configuración o desarrollo de un software o sistema operativo, y es utilizado para violar la seguridad del equipo o del sistema computacional, estamos frente a una vulnerabilidad.

"Una organización puede utilizar herramientas para la evaluación de vulnerabilidades, que permiten conocer la situación real de un sistema y mejorar su seguridad, verificando que los mecanismos de seguridad funcionan correctamente. Así mismo, estas herramientas pueden analizar y evaluar las vulnerabilidades del sistema informático, estableciendo un ranking en función de su severidad" (Gómez, 2000).

Al hablar acerca de vulnerabilidades en el área de tecnologías de información el campo es bastante amplio; en términos generales lo que se busca en la gestión de la seguridad es proteger los activos más importantes, estos serían los recursos que forman parte del sistema. De forma general los podemos agrupar en diferentes aspectos, los cuales nos darán como resultado un amplio campo de análisis para el tema de las vulnerabilidades.

I. El hardware, todo elemento físico del sistema informático como procesadores, medios de almacenamiento y cableado, entre otros. Las vulnerabilidades son la probabilidad de que ocurran

fallas físicas en cualquiera de los elementos del hardware que conforman el sistema. Pueden darse fallas por defectos de fabricación o mal diseño del hardware, pero igualmente podrían darse por falta de mantenimiento, o como resultado del mal uso.

II. El software, elementos lógicos o programas que se ejecutan sobre el hardware, algunos fallos en del software facilitan el acceso y por lo tanto son menos confiables. Los programas, ya sean de sistema operativo o bien de paquetería, se pueden utilizar como medios para atacar a otros sistemas superiores, las causas son debido a errores de programación, por la falta de aspectos como controles de acceso, seguridad al momento de ser diseñado. En todo caso, los dos factores contribuyen a que el sistema sea susceptible a las amenazas de software.

III. Los datos, es la información lógica procesada por el software, información estructurada en bases de datos o paquetes procesados por la red.

IV. Otros elementos como el acceso físico determinado por la facilidad con la que se pueda ingresar a un lugar, la infraestructura que incluye la transmisión de la información, sistemas con inadecuados sistemas de transmisión permiten que se puedan interceptar las comunicaciones en cualquier dirección, las personas tanto internas como externas. Las vulnerabilidades de este tipo pueden presentarse debido a malas prácticas en las políticas de acceso al personal a los sistemas y medios físicos de almacenamiento, y permiten la extracción de datos de forma no autorizada.

V. Las redes, son una serie de dispositivos conectados entre sí compartiendo recursos. Las redes pueden llegar a ser sistemas muy vulnerables, por ejemplo, es posible atacar a toda la red si se logra penetrar en un solo dispositivo. La prioridad de este sistema es la transmisión de la información, por lo tanto, las vulnerabilidades van a estar relacionadas con la intercepción de información por personas no autorizadas y que tengan fallas en la disponibilidad del servicio. Podemos ver a las vulnerabilidades de red como una combinación de las vulnerabilidades de hardware, software y físicas.

Cuando hablamos de vulnerabilidades a nivel de sistemas web en Costa Rica, se hace referencia a sistemas críticos que son de uso público por los ciudadanos y que una vulnerabilidad en ellos podría representar un riesgo para los usuarios, para la organización o hasta para el país. Si estas vulnerabilidades se llegan a conocer, el potencial riesgo al que se enfrentaría el país es bastante amplio, ya que las personas con intenciones maliciosas siempre van a existir y están a la espera de una oportunidad para sacar sus habilidades a relucir frente al mundo, exponiendo no solo la seguridad de los sitios sino la posibilidad de manipular y revelar información confidencial de la ciudadanía.

Al hacer uso de herramientas para el análisis de la evaluación de vulnerabilidades de un sistema informático, se deben considerar varios aspectos, como por ejemplo:

- Definición del alcance y objetivos de las pruebas a realizar.
- Conocimiento y experiencia del equipo que analiza las vulnerabilidades
- Actualización periódica de la base de datos de vulnerabilidades a analizar.
- Periodicidad de las pruebas.

Es muy importante tomar en cuenta todas las consideraciones mencionadas anteriormente, así como llevar a cabo una adecuada gestión de las herramientas para poder ser utilizadas de la manera más eficiente posible que permita obtener un beneficio para cualquier organización, sea privada o pública. El objetivo final es de brindar a los usuarios la confianza de que están utilizando un sitio seguro, o por el contrario evidenciar vulnerabilidades con el fin de que estas sean atendidas y así evitar un evento no deseado. Debido a la relevancia en materia de ciberseguridad para el país es que se pretende enfocar este estudio en las vulnerabilidades críticas.

2.4 Tipos de Pruebas de penetración

Existen diferentes tipos de pruebas de penetración en general, existen las pruebas de Caja Negra, pruebas de Caja Blanca o pruebas de Caja Gris.

Caja Negra: En este tipo de pruebas el experto posee poca o ninguna información por adelantado sobre el objetivo a evaluar. Este tipo de pruebas es más frecuente en pruebas de penetración a redes.

Caja Blanca: Las pruebas de caja blanca se realizan normalmente por un equipo interno de la organización. Estas pruebas son parte del ciclo de vida de desarrollo del software, en donde se tiene acceso al código fuente para su revisión y conocimiento del funcionamiento, así se reportan las vulnerabilidades conociendo de antemano cuál es el funcionamiento esperado.

Caja Gris: Este es uno de los tipos de pruebas más común, en donde se da una combinación de las pruebas de caja blanca y pruebas de caja negra. Se evalúa tanto las aplicaciones desde lo externo, como investigar a nivel de código buscando vulnerabilidades.

2.5 Métodos de prueba

Existen diferentes métodos de prueba en los análisis de detección de vulnerabilidades.

Manual

Este método es utilizado para detectar a lo que se enfrenta el analizador para así poder determinar qué herramientas se pueden usar y estimar cuánto va a ser la duración de las pruebas. Es un proceso relativamente lento comparándolo con otros métodos.

Automático

Existen herramientas que realizan de forma automática las pruebas, estas pueden hacer un escaneo de los sitios de forma rápida y genera un informe de las vulnerabilidades detectadas.

Híbrido

Este método permite tener un mayor control de las pruebas, al generar resultados basándose en métodos manuales y pruebas automáticas. Las herramientas automáticas se recomienda que sean utilizadas para tomarlas como punto de inicio para continuar con las pruebas manuales. Se necesita validar los resultados que fueron generados de forma manual con el fin de detectar falsos

positivos y poder interpretarlos de la mejor forma conociendo de antemano la aplicación o sistema. Así se puede conocer qué es considerado realmente una vulnerabilidad en la configuración actual del sistema.

2.6 Componentes de un sistema informático

"Una red es un conjunto de dispositivos interconectados físicamente, ya sea por medio de una conexión alámbrica o inalámbrica que comparten recursos y se comunican entre sí a través de reglas, protocolos de comunicación" (Martínez, 2011).

Las redes son conocidas mundialmente y su objetivo es compartir recursos, eso se logra mediante un conjunto de dispositivos interconectados que facilitan la transmisión de información. Entre los componentes principales de una red tenemos los servidores, aquí se ejecuta el sistema operativo y ofrece los servicios de red a las estaciones de trabajo. Las estaciones de trabajo son los equipos conectados a la red y que se pueden considerar un nodo, las tarjetas de red son las que permiten la interacción entre el equipo cliente y los equipos de red. La conexión entre dispositivos más conocida es a través de un medio físico como lo es un cable Ethernet, sin embargo, también existen las redes inalámbricas que utilizan la modulación de ondas electromagnéticas a través del espacio. Finalmente tenemos los recursos y periféricos compartidos como lo son las unidades de almacenamiento e impresoras, entre otros.

Las funciones de los equipos son variadas, por ejemplo:

Clientes, solicitan servicios o datos en la red a los equipos que se les denomina servidores, que son equipos que proporcionan datos y servicios realizando diversas tareas complejas. Existe una gran variedad de servidores, entre los que se pueden mencionar.

- Servidores web: Su función principal es procesar una aplicación o sistema que pueda ser consumida de forma bidireccional por un cliente.

- Servidor de archivos: Su función principal es compartir recursos de archivos de forma centralizada.
- Servidor de bases de datos: Su función es almacenar una gran cantidad de datos de forma centralizada y ponerlos a la disposición de los usuarios sin que tenga que descargarlos por completo.
- Servidor de correo: Tienen la misma función de compartir de forma centralizada una aplicación para ser consumida por los clientes.
- Servidor de dominio ActiveDirectory: Entre algunas de sus funciones se encuentra proveer de una identidad a los clientes dentro de un dominio común conocido por la organización a la que pertenece el usuario.
- Servidor tipo proxy: Funciona como intermediario dentro de una red recibiendo las solicitudes realizadas por un usuario tipo cliente hacia otro servidor, permitiendo de esta forma tener un mejor control de las conexiones.

Es esencial que los sistemas informáticos cuenten con un nivel mínimo de seguridad dentro de su infraestructura. En ocasiones las técnicas de hacking utilizadas por las personas malintencionadas que buscan detectar una vulnerabilidad para poder sacar un beneficio de la misma, permiten que se pueda alcanzar el objetivo y en la mayoría de las ocasiones eso sucede debido a débiles sistemas de seguridad que no incluyen la configuración de un Web Application Firewall (WAF), dispositivo ya sea tipo hardware o software que permite proteger los servidores web de cierto tipo de ataques por medio de internet.

Se puede decir que un WAF ayuda a mitigar el riesgo de ataque como los que se mencionan a continuación:

- Cross-site scripting: Consiste en la inclusión de código script malicioso en el cliente que consulta el servidor web.

- SQL injection: Consiste en introducir un código SQL que vulnere la Base de Datos del servidor.
- Denial-of-service: Consiste en que el servidor de aplicación sea incapaz de servir peticiones correctas de usuarios.

Los WAF se pueden categorizar en 2 tipos, los que son un elemento más de la misma red, por lo tanto, residen en ella y aquellos que se basan en el servidor de aplicaciones, por lo tanto, residen en el servidor.

Otros tipos de ataques que existen son:

- Ataques por intercepción: La información es interceptada por un tercero en el momento que el emisor transmite al receptor, este ataque es más conocido como man-in-the-middle.
- Ataques por suplantación de identidad: El atacante se hace pasar por el emisor generando información y transmitiendo al receptor, es decir, se permite el acceso a personas no autorizadas.
- Ataque por modificación: Este tipo de ataque es aquel en que la información es enviada por el emisor, se intercepta y es alterada para posteriormente enviarla al receptor.

Para reducir el riesgo de que las vulnerabilidades sean atacadas, existen en la actualidad herramientas que ayudan a realizar escaneos y que brindan una información detallada de las potenciales vulnerabilidades, así como su nivel de riesgo y cómo solucionarlas. De aquí surgen los exploits, que podemos definirlos como un programa o código que se aprovecha de una brecha de seguridad en un sistema o aplicación, de tal manera que pueda ser utilizada en beneficio del atacante. Los podemos distinguir en dos tipos, los conocidos y los desconocidos (0-day).

1. Los conocidos: Son aquellos que sabemos que existen y se pueden tomar medidas para evitar que se afecten los sistemas, son de conocimiento general por todas las personas y

publicados en sitios web especializados, como por ejemplo cve.mitre.org, y todos los días aparecen, así como las vulnerabilidades que se tratan de aprovechar.

2. Los desconocidos o 0-days: Son utilizados en vulnerabilidades que aún no han sido reportadas al público y suponen una gran amenaza, son muy atractivos por los atacantes para dirigirlos hacia gobiernos o empresas, son prácticamente indetectables.

2.7 Gestión del ciclo de vida de las vulnerabilidades

La evaluación y gestión de las vulnerabilidades es esencial para el manejo del riesgo en las tecnologías de información. Es un proceso continuo de identificación, evaluación y corrección en los sistemas de información de una organización, debido a que existen amenazas constantes y los criminales no descansan. Este proceso va más allá de la evaluación, ya que califica los activos de una organización y los clasifica de acuerdo con el riesgo que representan, por lo tanto, es un medio que contribuye en gran medida a la protección de la infraestructura tecnológica, y resulta un proceso económicamente atractivo para las empresas.

La importancia que conlleva la gestión de vulnerabilidades es un reto constante, debido a la gran complejidad que existe en los sistemas, sean estos distribuidos (que se ubican geográficamente en distintas zonas o en diferentes equipos) o centralizados (que todo se aloja en un solo equipo), es una tarea compleja y difícil de cuantificar. Además, si a lo anterior le sumamos la complejidad que existe en la infraestructura de telecomunicaciones, las cuales permiten la interacción entre las distintas aplicaciones, sistemas y bases de datos desde cualquier punto de conexión, hacen aún más difícil la labor de gestionar eficientemente los potenciales problemas en seguridad.

Una adecuada gestión de las vulnerabilidades en los sistemas informáticos contribuirá para que las organizaciones puedan dar prioridad a solucionar las fallas de seguridad en aquellos sistemas con mayor exposición al peligro, de esa forma aquellos sistemas que sean menos vulnerables

podrán gestionarse en segundo plano con un plan que permita reducir en la medida de lo posible ataques que afecten directamente a la productividad, finanzas e imagen de la empresa. Es más fácil actuar de forma proactiva que tratar de solucionar los problemas en situaciones inesperadas, también contribuye al aspecto de cumplimiento con el manejo de regulaciones del medio, como por ejemplo la norma ISO27001.

Para realizar un análisis de amenazas y vulnerabilidades de forma efectiva es importante aplicar una metodología consistente que examine las amenazas técnicas o de negocio a una aplicación o servicio. Dentro de los conceptos importantes conocidos que se aplican de forma estándar a las vulnerabilidades tenemos:

CVE - Common Vulnerabilities and Exposures: Es un diccionario de nombres comunes disponible al público que facilita poder compartir la información en las diferentes bases de datos de seguridad o herramientas, entre otras. El estándar CVE se emplea a la hora de publicar parches por parte de los fabricantes, así como los informes de las herramientas automáticas de análisis de vulnerabilidades.

CVSS - Common Vulnerability Scoring System: Se utiliza como un sistema de medición estándar para las organizaciones y gobiernos que necesitan puntuaciones precisas y consistentes.

CWE – Common Weakness Enumeration: Facilita de manera que ayuda a enumerar las diferentes causas de vulnerabilidades de seguridad que se encuentran en la arquitectura del código, diseño o sistema.

Por otro lado, existen algunos estándares para asegurar la calidad de los trabajos realizados y su evaluación por parte de terceros, por lo tanto, es importante mencionarlos como parte de esta investigación. Por ejemplo, OSSTMM (Open Source Security Testing Methodology Manual) del ISECOM (Institute for Security and Open Methodologies) es un manual con una serie de secciones compuestas por módulos que incluyen las distintas pruebas que se podrían realizar en

una auditoría técnica de seguridad. Otro proyecto que se puede mencionar es el OWASP (Open Web Application Security Project) para evaluar la seguridad de las aplicaciones web, así como las guías de pruebas de seguridad de red (Guideline on Network Security Testing). Para el desarrollo de esta investigación la metodología para realizar pruebas de penetración que se va a utilizar es la recomendada por OWASP, que se explicará en detalle en el siguiente capítulo.

2.8 Estrategia para el desarrollo de la propuesta

2.8.1 Open Web Application Security Project (OWASP)

Nace en abril 2004 y se establece como una fundación sin fines de lucro en los Estados Unidos. OWASP es una comunidad abierta que se dedica a proveer a las organizaciones, herramientas y buenas prácticas para desarrollo seguro de aplicaciones. Todas las herramientas de OWASP son gratis y abiertas para las personas que las quieran utilizar para mejorar sus aplicaciones, sus procesos y su tecnología, o ser parte de su comunidad.

Los valores de OWASP:

- **Abierta**: Todo en OWASP es radicalmente transparente, desde el código hasta la parte financiera.
- **Innovación**: OWASP fomenta y soporta la innovación y experimentos para soluciones a los retos de seguridad en los programas de software.
- **Global**: Cualquier persona en el mundo está empoderada a participar en la comunidad OWASP.
- **Integridad**: OWASP es una comunidad honesta, veraz, neutral y global.

Entre sus proyectos más conocidos se puede mencionar:

OWASP Top 10

El top 10 de OWASP contempla una lista de los 10 riesgos de las aplicaciones web que son más críticos. Para cada riesgo provee la descripción, vulnerabilidades de ejemplo, ataques de ejemplo, una guía de cómo evitarla y referencias a recursos relacionados con ese riesgo.

OWASP Zed Attack Proxy (ZAP)

Es una aplicación que sirve como herramienta para analizar vulnerabilidades, también como proxy para ver los llamados que hacen las aplicaciones.

OWASP Testing Guide

El "Testing Guide 4.0" de OWASP es una metodología para la revisión y evaluación de vulnerabilidades en aplicaciones web.

Brinda una serie de buenas prácticas para realizar pruebas, las cuales están distribuidas en las siguientes categorías:

1. Recopilación de información
2. Pruebas de gestión de la configuración y la implementación
3. Pruebas de gestión de la identidad
4. Pruebas de autenticación
5. Pruebas de autorización
6. Pruebas de gestión de sesión
7. Pruebas de validación de datos de entrada
8. Manejo de errores
9. Criptografía
10. Pruebas de lógica de negocio
11. Prueba del lado del cliente

La metodología que se utilizará en este proyecto se basa en el OWASP Testing Guide versión 4, adaptada a pruebas de caja negra específicamente para sitos web, lo que permitirá enfocarse en las áreas consideradas por la comunidad de OWASP como mejores prácticas a la hora de validar la seguridad en aplicaciones web. Por lo anterior, esta es una metodología adaptada que incluye solamente los controles que aplican para pruebas de caja negra y manuales.

La prueba de seguridad de esta metodología consiste en dos fases:

La fase pasiva: En esta fase el experto que ejecuta las pruebas intentará entender la lógica de la aplicación, utilizando la aplicación como un usuario normal tratando de entender su funcionamiento. Para esta fase el experto puede utilizar herramientas como un HTTP proxy para interceptar todos los llamados y poder entender el funcionamiento de la aplicación, sus interacciones con otros servidores, los parámetros que envía, los encabezados HTTP.

La fase activa: En esta fase el experto empieza las pruebas usando la metodología descrita en esta metodología, la cual está distribuida en las once subcategorías anteriormente mencionadas y detalladas a continuación. Dentro de las once subcategorías que se explorarán no todos los controles serán utilizados, ya que como se ha venido mencionando, es una metodología adaptada y aceptada por la comunidad.

1. Pruebas para recopilación de información

Entender la aplicación que se va a probar es uno de los pasos más importantes para poder determinar cómo se realizarán las pruebas.

Realizar a través de motores de búsqueda, información que ayude a detectar fugas de información o información expuesta (OTG-INFO-001)

¿Cómo se puede probar? Utilizando los operadores de búsqueda como “URL” o “SITE” es posible restringir al buscador para que entregue todos los resultados relacionados específicamente a un sitio web o a un URL.

Google Hacking Database

La base de datos de hacking de Google es una lista de búsquedas parametrizadas para diferentes categorías.

Fingerprinting Servidor Web (OTG-INFO-002)

Esta técnica permite conocer la versión y tipo del servidor web que corre en ese sitio. Con esto, el experto puede determinar vulnerabilidades conocidas y qué tipo de exploits utilizar durante las pruebas.

Pruebas de revisión de metaarchivos del servidor web en busca de fugas de información (OTG-INFO-003)

El objetivo de estas pruebas es buscar por información de directorios o rutas del web server que no están registradas para ser invocadas desde la página.

Enumerar las aplicaciones en el Servidor Web (OTG-INFO-004)

Descubrimiento de aplicaciones web, consiste en detectar qué otras aplicaciones existen en ese mismo servidor web. El objetivo es buscar las aplicaciones que comparten el mismo servidor. Algunas herramientas como Nessus o Nmap son esenciales para este proceso.

Revisión de los comentarios del sitio web y los metadatos en busca de fugas de información (OTG-INFO-005)

Para los programadores es muy común incluir comentarios detallados y metadatos en su código fuente. Sin embargo, estas prácticas pueden incluir información sensible que podría ser valiosa para potenciales atacantes. Para estas pruebas se recomienda realizar de forma manual,

búsquedas a criterio del experto, como por ejemplo con el “View Source” del navegador que se utilice.

Identificar puntos de entrada de la aplicación (OTG-INFO-006)

Identificar la aplicación y su superficie de ataque es clave antes de iniciar otras pruebas, le permite al experto identificar áreas de debilidad. Para esto es necesario entender cómo se realizan los llamados de la aplicación y cuáles son las respuestas usuales.

El experto debe tomar nota de cualquier cosa que le parezca sospechosa o interesante y pueda ser valiosa a futuro. Una opción para realizar estas pruebas puede ser incluir herramientas tipo proxy como ZAP, Fiddler.

Creación de mapas de rutas de ejecución a través de la aplicación (OTG-INFO-007)

Otro punto necesario es conocer la estructura de la aplicación a probar, mapeando y entendiendo los principales flujos de trabajo. En pruebas de caja negra esto es usualmente muy difícil de realizar, ZAP proporciona una característica de la aplicación que permite enlistar la estructura y todos los llamados que se realizan.

Fingerprint Web Application Framework (OTG-INFO-008)

Conocer el tipo de framework permite tener una ventaja para las vulnerabilidades que se quieren probar, así como configuraciones específicas, estructura de archivos, etc. Para esto se puede basar en pruebas a nivel de código fuente, búsqueda de directorios específicos o archivos específicos.

Fingerprint Web Application (OTG-INFO-009)

Con este tipo de pruebas se busca conocer los componentes de la aplicación web, con el objetivo de adaptar las pruebas específicamente a la aplicación. La forma más simple para revisar esto es mediante los “cookies” de la aplicación, que normalmente le da al usuario información del

framework utilizado. Otro método de detección es mediante el código fuente de la página, buscando rutas específicas y nombres de carpetas o archivos específicos.

Mapa de arquitectura de la aplicación (OTG-INFO-010)

Estas pruebas buscan determinar todos los posibles componentes que interactúan con la aplicación. Para ello es importante que se trace un mapa de la arquitectura de la aplicación, por medio de pruebas para determinar qué tipo de arquitectura es la que se va a probar. Por ejemplo, si el sitio posee un WAF, o un proxy reverso, dichos hallazgos se pueden encontrar en los resultados de los escaneos de red que se pueden realizar al servidor web, así como los resultados obtenidos por la herramienta encargada de realizar los escaneos de puertos.

2. Pruebas de gestión de la configuración y la implementación

Pruebas de manipulación de extensiones de archivos buscando información sensible (OTG-CONFIG-003)

Las extensiones de los archivos en los servidores web son normalmente utilizadas para determinar fácilmente qué tecnología está corriendo la aplicación. Para estas pruebas el experto puede realizar envíos HTTP(s) a la página solicitando diferentes archivos con diferentes extensiones. Identificar utilizando la estructura del server anteriormente mencionada, cuáles son los repositorios que puedan tener archivos con información sensible. Por ejemplo, un archivo llamado connection.inc que al abrirlo trae la cadena de conexión al servidor de base de datos.

Las siguientes extensiones no deben ser devueltas por el servidor web.

- *.asa*
- *.inc*

Las siguientes extensiones de archivos deben ser validadas por el administrador de la página para asegurarse de que deben ser descargados o servidos cuando alguien los identifica.

- .zip, .tar, .gz, .tgz, .rar, (extensiones de archivos comprimidos)
- .java: Los archivos de código java no deberían ser accesibles
- .txt: Archivos de texto
- .pdf: Los documentos en formato PDF
- .doc, .rtf, .xls, .ppt: Los documentos con extensiones de Microsoft office.
- .bak, .old new, test: Los archivos que son renombrados que se dejan como respaldo o como prueba.

Para realizar estas validaciones, una opción es realizar pruebas manuales con el escáner de vulnerabilidades, por ejemplo, Nessus.

Pruebas de Métodos HTTP (OTG-CONFIG-006)

HTTP ofrece múltiples métodos para realizar acciones en los servidores web, muchos de estos métodos están diseñados para ayudar a los desarrolladores a publicar y probar las aplicaciones.

Algunos de estos métodos pueden representar un riesgo de seguridad potencial en una aplicación web. Los métodos que se recomienda deberían estar deshabilitados son:

- **PUT**: Permite al cliente subir archivos al servidor web, por lo tanto, un atacante podría explotar este método subiendo archivos maliciosos.
- **DELETE**: Permite borrar archivos de un servidor web y comprometer su funcionalidad.
- **CONNECT**: Le permite a un cliente usar el servidor web como proxy.
- **TRACE**: Devuelve al cliente cualquier string que haya sido enviado al servidor.

Nessus o FOCA pueden descubrir los métodos permitidos por el servidor web.

3. Pruebas de gestión de identidad

El dominio tres de esta metodología no será incluido, debido a que los sitios web no cumplen con las características aquí descritas.

4. Pruebas de autenticación

Pruebas de credenciales por defecto (OTG-AUTHN-002)

Actualmente muchas aplicaciones web hacen uso de programas comerciales que se instalan con una mínima intervención en la configuración o personalización del administrador del servidor. En ocasiones estas aplicaciones al instalarse no se configuran adecuadamente, por lo que las contraseñas y usuarios que vienen configurados, por defecto no son modificadas. Estas contraseñas son de fácil acceso y conocidas mundialmente por atacantes, lo que facilita la posibilidad de acceder a las aplicaciones.

Algunos ejemplos de prueba se pueden ver a continuación.

En los formularios de ingreso se pueden realizar intentos de conexión utilizando los siguientes usuarios: admin, administrator, administrador, guest, root, operador, operator, super, test, qa, test1, testing.

Es muy común ver que los anteriores usuarios son utilizados. Podría ser posible hacer un intento de validación, ingresando en el campo de usuario y contraseña los usuarios anteriormente citados, en caso de que la aplicación sea vulnerable al método de enlistar usuario, sería posible poder identificar el uso de cualquiera de los usuarios de uso común. Por otro lado, si se han dejado usuarios por defecto es probable que también existan contraseñas por defecto del fabricante, entre algunas se puede mencionar:

123456,12345, pass123, password123, admin, guest, invitado, contraseña, prueba, contraseña vacía,. Cualquier otra combinación de usuarios y contraseñas de uso común y de conocimiento general podrían ser probadas para validar su seguridad.

Los usuarios tipo administrador normalmente son configurados como parte del nombre de la empresa o bien del sitio web, por lo que, si se está probando una aplicación en una empresa

llamada “empresa”, sería adecuado probar el usuario “empresa” o usuarios con la palabra “empresa” como parte del usuario, por ejemplo:

empresa1 ,adminempresa, userempresa, usuarioempresa, etc.

Revisar el código fuente de la página web, se puede hacer búsquedas a referencias, usuarios y contraseñas que en ocasiones son dejados erróneamente por los desarrolladores en forma de comentarios. El experto en seguridad puede realizar algunas pruebas manuales, con el fin de intentar identificar algún usuario/contraseña que le permite ingresar como usuario válido en la aplicación.

Pruebas para eludir el esquema de autenticación (OTG-AUTHN-004)

Aún y cuando la mayoría de las aplicaciones proveen mecanismos de autenticación para obtener acceso a la información o para ejecutar tareas restringidas, no todos estos métodos proveen una adecuada seguridad. Algunos mecanismos debido a una mala implementación o desarrollo permiten a los usuarios saltar el proceso de autenticación y llamar directamente un recurso que debería estar protegido.

Para realizar este tipo de pruebas se pueden utilizar varios métodos.

1. Solicitud directa de la página

Este método consiste en solicitar o invocar directamente en el URL para forzar la página a solicitar este nuevo recurso.

2. Modificación de parámetros

En ocasiones en el querystring de la página se agregan parámetros para definir el nivel de autenticación o rol del usuario. El usuario podría modificar el valor del parámetro para intentar engañar a la aplicación.

Por ejemplo:

http://www.site.com/page.asp?authenticated=no

En este sitio web, el atacante podría manipular el parámetro "authenticated" cambiándole el valor de "no" a "yes", si este método no funciona, se puede probar realizando el cambio del valor de "true" o "sí".

Para este tipo de prueba también es posible utilizar un proxy para interceptar los llamados que viajan por método POST, con el fin de interceptar y modificar los llamados.

3. Predicción del ID de la sesión

Muchos sitios web gestionan la sesión mediante un identificador, por lo que, si la generación de un identificador es predecible, un usuario malicioso podría encontrar una sesión válida y obtener acceso. La variable de sesión se puede buscar en el cookie o en el URL y validar si es enviado de esa forma.

4. Inyección SQL en el formulario de autenticación

Inyección de código SQL es una técnica muy conocida para saltarse la autenticación de formularios y acceder a la base de datos. Para estas pruebas se puede utilizar la herramienta SQLmap.

Prueba para cambios de clave débil o funcionalidad de resetear clave (OTG-AUTHN-009)

La modalidad de cambiar la contraseña es un mecanismo que le permite al usuario de forma independiente, realizar el cambio directamente a través de la página web sin la intervención de un administrador. Para este control, es necesario probar:

1. Si los usuarios pueden cambiar la contraseña de otros usuarios.
2. Si los usuarios pueden manipular el proceso para cambiar la contraseña de otros usuarios.

Además de lo anterior, es importante revisar si existen mecanismos de validación junto al correo, como por ejemplo una pregunta secreta, que sea necesario responder para iniciar el proceso de recuperación de contraseña, lo cual está mal porque se confía solamente en la seguridad del correo electrónico. Por otro lado, si existen preguntas secretas que se realicen, entonces el siguiente paso es validar la seguridad de las mismas.

¿Cuál es el mecanismo para comunicar la contraseña al usuario?

Si el mecanismo para restablecer la contraseña muestra la contraseña en la misma página, es la modalidad más débil para restablecer una contraseña, eso facilita la capacidad del atacante para ingresar a la cuenta de la víctima de forma sencilla.

Otro escenario aún más inseguro sería si la herramienta para restablecer la contraseña obliga al usuario a cambiar inmediatamente la contraseña.

Un escenario más recomendable, que aun así no garantiza la seguridad completa, es enviando un correo electrónico previamente registrado al usuario que solicita que se restablezca la contraseña, para esto el atacante obligatoriamente necesita saber o adivinar cuál es el correo electrónico al que se ha enviado la nueva contraseña, para posteriormente intentar acceder a ese correo, y poder tener acceso a la contraseña o al link enviado para restablecer la contraseña, lo que requiere más información.

Para limitar una denegación de servicio, la aplicación debería enviar un link por email al usuario, con un token aleatorio y solo si el usuario visita el link, el procedimiento para resetear la contraseña es realizado. Esto ayuda a asegurar que la contraseña actual va a ser válida hasta que el reseteo de la contraseña sea confirmado.

5. Pruebas de autorización

Las pruebas de autorización consisten en habilitar acceso a recursos de la página, para habilitar el ingreso únicamente a usuarios que realmente lo necesitan. Para esto se requiere entender cómo funciona la página web y así poder investigar por dónde se podría realizar las pruebas de autorización.

Testing Directory traversal/file include (OTG-AUTHZ-001)

Tradicionalmente las aplicaciones web implementan mecanismos para controlar acceso a recursos y archivos, con el propósito de que el administrador pueda limitar a los usuarios a qué recursos pueden acceder dentro del sitio. Estos mecanismos de acceso se manejan mediante listas de control de acceso.

Los ataques de file/traversal provocan que un atacante pueda tener acceso a estos archivos protegidos, leer los archivos, acceder a directorios a los que no debería tener acceso.

Para este control de OWASP solo se evaluará lo correspondiente a aplicaciones web.

Para probar este control es necesario analizar todas las partes de la página que aceptan contenido del usuario.

Es necesario revisar los parámetros y rutas en las que se hacen llamados al realizar alguna acción.

- http://example.com/getUserProfile.jsp?item=ikki.html
- http://example.com/index.php?file=content
- http://example.com/main.cgi?home=index.htm

En los llamados anteriores, la página dinámica llamada getUserProfile.jsp carga información de un archivo y muestra el contenido al usuario.

Un atacante podría insertar la cadena de texto maliciosa "../../../../etc/passwd" para incluir el contenido del archivo hash de un sistema Linux/Unix.

Un ejemplo de dicha prueba se muestra a continuación:

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

Se debe considerar que lo anterior no tendría sentido en ambientes que tengan un sistema IIS.

También se podría incluir contenido de otra página, por ejemplo

http://example.com/index.php?file=http://www.google.com

Para tener control sobre el código de un componente CGI (Common Gateway Interface, por sus siglas en inglés), se puede hacer de la siguiente manera:

http://example.com/main.cgi?home=main.cgi

Las herramientas que se pueden utilizar para este tipo de validaciones son ZAP, pruebas manuales y herramientas de Enconding/Decoding.

Pruebas de Insecure Direct Object References (OTG-AUTHZ-004)

Esto ocurre cuando la aplicación provee acceso directo a objetos con base en la información provista por el usuario. Por ejemplo, al acceder a una factura en la que el sistema se basa en el parámetro "invoice", para indicar la factura a desplegar, se puede obtener información utilizando un llamado como el que se muestra.

http://foo.bar/somepage?invoice=12345

Por lo tanto, si un usuario no autorizado en ingresar a esa factura logra ganar acceso y cambia el valor del "invoice", por ejemplo, de 12345 a 12346, podría acceder con un consecutivo a las siguientes facturas de forma sencilla y sin restricciones a modificarlas.

6. Pruebas de gestión de sesión

Estas pruebas no se realizarán, ya que las páginas web a las que se le aplicará está metodología no cuentan con acceso a los usuarios y no es parte del alcance de este proyecto.

7. Pruebas de validación de datos de entrada

Una de las debilidades de seguridad más comunes en las aplicaciones web es la incorrecta validación de las entradas que provee el usuario o el cliente. Nunca se debe confiar en la información de entidades externas o clientes, debido a que podría haber sido manipulada por un atacante.

Pruebas de Reflected Cross site scripting (OTG-INPVAL-001)

Reflected Cross Site Scripting ocurre cuando un atacante inyecta código ejecutable por un navegador en una respuesta HTTP. Normalmente este código está escrito en javascript. El Reflected Cross Site Scripting no está almacenado dentro de la aplicación, únicamente afecta a usuarios que eventualmente puedan abrir un link con código malicioso.

¿Cuál es la forma de probarlo?

Detectar todos los posibles parámetros o variables de la aplicación en donde se podría inyectar Cross Site Scripting, parámetros HTTP, información que viaja por POST. Analizar cada parámetro o variable con alguna secuencia de código JavaScript que puede fácilmente detectar la vulnerabilidad. Algunos ejemplos se muestran a continuación.

<script>alert('test')</script>

"><script>alert(document.cookie)</script>

Para cada una de las pruebas, el experto en seguridad analizará los resultados obtenidos y determinará si representan o no una vulnerabilidad.

Como una prueba válida se puede mencionar que, en un link compuesto, al hacer clic, si se obtiene una ventana o pop up que muestra una alerta, entonces se puede confirmar la existencia de la vulnerabilidad de Reflected Cross Site Scripting.

http://example.com/index.php?user=%3Cscript%3Ealert(123)%3C/

Existen diferentes técnicas que permiten burlar la protección contra este tipo de vulnerabilidades, por ejemplo, sintaxis distintas o encoding. Los siguientes ejemplos tienen todos una misma función.

“ onfocus=”alert(document.cookie)

“><script >alert(document.cookie)</script >

“><script >alert(document.cookie)</script >

“%3cscript%3ealert(document.cookie)%3c/script%3e

Para llevar a cabo pruebas que ayuden con este control, es posible realizar pruebas manuales tomando como base el “OWASP XSS Filter Evasion Cheat Sheet”, pruebas con herramientas conocidas como ZAP y Acunetix.

Pruebas de Cross Site Scripting almacenado(OTG-INPVAL-002)

El Cross Site Scripting almacenado puede ser explotado en los sitios que permiten a los usuarios almacenar información. Se da cuando la aplicación acepta la información del usuario para ser utilizada posteriormente, la información guardada no es debidamente filtrada, por lo tanto, es tomada como código de la página web. Esta vulnerabilidad puede conducir a ataques, por ejemplo:

- Robo de sesiones de usuarios.
- Capturar información sensible de otros usuarios.
- Entregar archivos maliciosos a otros usuarios.

Esta vulnerabilidad no requiere de un link malicioso para ser explotada, a continuación, se resume la manera como podría ser explotada por un atacante.

El usuario ingresa a una sección del sitio web, esa sección lo que hace es desplegar información, por ejemplo, comentarios que son ingresados por otros usuarios. Una vez que esto ha sido descubierto, el atacante decide ingresar código para explotar "*Cross Site Scripting Almacenado"*. Ese código es ingresado de la misma manera que se hace con un comentario, una vez almacenado en el momento que otro usuario ingresa a esa misma sección, el código malicioso es liberado o ejecutado.

El proceso para identificar este tipo de vulnerabilidad es similar al proceso anterior de "*Reflected Cross Site Scripting"*. Se identifica primero todas las secciones donde un usuario puede ingresar información, para poder validar que es posible utilizar esos campos o valores donde se ingresa la información. El ejemplo más claro se puede ver con un formulario, que permita crear una cuenta en que el usuario sea modificado a gusto de la persona que ingresa los datos personales, de esa manera la cuenta se puede utilizar como un usuario válido.

aaa@aa.com"><script>alert(document.cookie)</script>

Se puede decir entonces que, si la página web tiene dicha vulnerabilidad, una vez que se despliegue el usuario en la página, se mostrarán los valores del cookie del usuario disponibles en la página. Por consecuencia, si un usuario administrador decide generar un reporte de usuarios a través del sistema, en el instante que se muestre el usuario del atacante, se desplegará una ventana emergente con la información del cookie del usuario administrador.

En caso de que el atacante en lugar de desplegar el cookie decida ingresar el siguiente parámetro:

aaa@aa.com"><script src=http://attackersite/hook.js></script>

Entonces se va a ejecutar en la víctima el archivo *hook.js,* el cual contiene código que establece una comunicación con el "*browser explotation framework"* del atacante. Por esa razón el atacante podría obtener acceso de la pantalla en la computadora de la víctima, acceder el cookie, el clipboard y otro tipo de ataques.

Pruebas de Inyección de SQL (OTG-INPVAL-005)

Un ataque de *Inyección de código SQL* consiste en inyectar código SQL en un parámetro o variable que reciba datos del sistema o del cliente y se pueda manipular, con el fin de que la aplicación ejecute el código, de manera tal que lo reciban como si fuera un valor esperado y no código SQL.

¿Cómo detectar una aplicación que tenga inyección de SQL?

El primer paso requerido es detectar los tipos de aplicaciones que tienen interacción con la base de datos, y que procesen información. Algunos ejemplos son:

- Formularios de autenticación
- Motores de búsqueda o campos de búsqueda
- Sitios de compras

El experto deberá realizar una lista de todos los campos que aceptan información del usuario, incluidos campos escondidos o que viajan por POST.

Un método de prueba sencillo y práctico recomendable para utilizar es agregar una comilla sencilla **" ' "** o un punto y coma **";"** en el campo que se desea ejecutar la consulta, esperando obtener un resultado que proporcione una idea al atacante. En caso de que el resultado mostrado en la página sea un "error de sintaxis de SQL", es altamente probable que dicha aplicación tenga la vulnerabilidad descrita anteriormente.

También se pueden utilizar algunos delimitadores como (**-- *or* /* */**, etc), o bien operadores como *AND, OR* para modificar el query.

Otra prueba válida sería ingresando únicamente letras para un campo que se supondría solamente acepte números, el error generado podría suministrar de herramientas a un atacante, brindándole el conocimiento de que existe una alta posibilidad de que exista la vulnerabilidad aquí descrita.

Para estas pruebas el experto puede utilizar pruebas manuales para confirmar si existe la vulnerabilidad, así como la herramienta SQLMap que ejecuta automáticamente las pruebas en el sistema.

Pruebas de inyección de comandos (OTG-INPVAL-013)

La inyección de comandos consiste en ejecutar comandos del sistema operativo a través de la aplicación. De esta manera el usuario podría tener la habilidad de subir archivos con código malicioso o hasta obtener contraseñas.

¿Cómo realizar pruebas de inyección de comandos?

Cuando se ve un archivo en una aplicación web el nombre del archivo normalmente se muestra en el contenido del URL. Perl permite canalizar datos de un proceso hacia una instrucción abierta. El usuario simplemente debe añadir el símbolo pipe "|" en el final del nombre del archivo.

Ejemplo antes del cambio:

http://sensitive/cgi-bin/userData.pl?doc=user1.txt

Ejemplo con el cambio:

http://sensitive/cgi-bin/userData.pl?doc=/bin/ls|

Esto ejecutará el comando /bin/ls

8. Manejo de errores

Pruebas de errores de código (OTG-ERR-001)

A menudo, durante las pruebas en aplicaciones web, se encuentran errores de código generados por la aplicación o por el servidor web. Es posible que estos errores sean desplegados con ciertos parámetros o ciertas solicitudes. Estos errores revelan información sobre bases de datos, errores y otros componentes tecnológicos directamente relacionados con aplicaciones web. Lo anterior es información muy útil para un atacante.

Un error común que se despliega en las pruebas es el HTTP 404 Not Found. A menudo este error proporciona información valiosa sobre el servidor web y componentes asociados.

Por ejemplo:

Not Found

The requested URL /page.html was not found on this server

Apache/2.2.3 (Unix) mod_ssl/2.2.3 OpenSSL/0.9.7g DAV/2 PHP/5.1.2 Server at

localhost Port 80

Los errores de aplicaciones son errores devueltos por la misma aplicación, pueden ser mensajes de código asp o php, errores específicos devueltos por el código de la aplicación. Los errores detallados de la aplicación normalmente proporcionan información de las rutas del servidor, bibliotecas instaladas y versiones de aplicaciones.

Se puede utilizar también errores para enumerar usuarios, por ejemplo, ingresando un usuario en un formulario y que al darle ingresar el error sea “Usuario no existe”.

También para enumerar folders de la aplicación.

Prueba: 403 Forbidden:

Enumeración de directorios usando mensajes de error de acceso denegado:

http://<host>/<dir>

Respuesta:

Directory Listing Denied

This Virtual Directory does not allow contents to be listed.

Pruebas para determinar los rastros de pilas de datos (OTG-ERR-002)

No son vulnerabilidades por sí mismas, pero revelan usualmente información interesante para un atacante. Los atacantes intentan generar rastros de pila de datos mediante la alteración del ingreso a la aplicación web con peticiones HTTP con un formato incorrecto y otros datos de ingreso.

Si la aplicación responde con rastros de pilas de datos que no se manejan, podría revelar información útil a los atacantes. Para estas pruebas, ZAP las realiza de forma automática durante un escaneo de vulnerabilidades normal.

9. Criptografía

Pruebas de SSL/TLS Ciphers débiles, protección de transporte de capas insuficiente (OTG-CRYPST-001)

Los datos sensibles deben ser protegidos cuando se transmiten a través de la red, datos como credenciales, tarjetas de crédito e información del usuario. Por ello, como regla, si los datos se deben proteger cuando se almacenan, también se deben proteger durante la transmisión.

HTTP es un protocolo de texto y normalmente se asegura mediante un túnel SSL/TLS, lo que da como resultado el tráfico HTTPS. Este protocolo HTTPS nos asegura no solo confidencialidad, sino también autenticación por medio de los certificados digitales. Por esto es necesario que:

- Se configuren correctamente los protocolos SSL/TLS.

- Que no se utilicen ciphers débiles o protocolos vulnerables conocidos.
- El certificado a utilizar sea válido.
- Utilizar bandera de seguridad para las cookies de sesión.
- No utilizar contenido mezclado http y https.
- Validar que el certificado generado sea para la página que se está ingresando.

Para realizar estas pruebas se requiere de varias herramientas, como pruebas manuales, utilizar un proxy como ZAP, utilizar un escáner de vulnerabilidades como Nessus para detectar los protocolos utilizados por el servidor web.

Pruebas de envío de información sensible por canales sin encriptar (OTG-CRYPST-003)

Los datos sensibles deben viajar en HTTPS, si la aplicación transmite información sensible a través de canales sin codificar - por ejemplo, HTTP - se considera un riesgo de seguridad. Para realizar estas pruebas es necesario pruebas manuales y utilizar un proxy como ZAP o Fiddler.

10. Pruebas de lógica de negocio

10.1 Prueba de la habilidad para manipular consultas (OTG-BUSLOGIC-002)

Se verifica que la aplicación no permite a los usuarios enviar o modificar los datos de cualquier componente del sistema al que no deben tener acceso, que estén accediendo en ese momento o de esa manera en particular. Para probar esto, el experto puede utilizar un proxy para interceptar los llamados y así poder cambiar el valor de ciertos parámetros ocultos que viajan por POST para tratar de, al modificar su valor, obtener un resultado diferente.

10.2 Prueba de defensa contra el mal uso de la aplicación (OTGBUSLOGIC-007)

En las pruebas de defensa contra el mal uso de la aplicación, se verifica que la aplicación no permite a los usuarios manipular la aplicación de una manera no deseada. Para esto el experto puede manipular los valores del cookie o parámetros del querystring para verificar si existe algún

comportamiento no esperado, por ejemplo, agregando una comilla simple en un parámetro, o duplicando un parámetro y ver el resultado.

10.3 Prueba de la posibilidad de carga de tipos de archivos inesperados (OTG-BUSLOGIC-008)

Se verifica que la aplicación no permite a los usuarios cargar tipos de archivos que el sistema no espera o no requiera de acuerdo con la lógica del negocio. Por ejemplo, que el sistema no soporta subir archivos ejecutables en un formulario de envío de recomendaciones. El experto en este caso podría probar enviando un archivo ejecutable a ver cuál es la respuesta.

11. Prueba del lado del cliente

Para este proyecto de tesis no se puede realizar los controles de esta sección, ya que incluye manipulación a nivel del código de la aplicación para poder obtener los resultados o pruebas necesitadas. Es por eso que los controles de la categoría de “Prueba del lado del cliente” no serán aplicados en el presente proyecto.

CAPÍTULO III

MARCO METODOLÓGICO

3.1 Tipo de investigación

La investigación por desarrollar es evaluativa. Se pretende hacer un estudio de las distintas vulnerabilidades en las páginas web del gobierno central de Costa Rica con el fin de brindar un informe del estado actual; como complemento se pretende utilizar una serie de recomendaciones sobre las medidas a tomar para generar un criterio de experto, ayudar en la toma de decisiones y para asegurar los sistemas.

Se utilizarán los resultados obtenidos para ser compartidos con el MICITT con el fin de que la investigación sirva de herramienta para que los diferentes ministerios del gobierno central de Costa Rica se vean beneficiados, puedan conocer el estado actual de su principal página web y solucionar las vulnerabilidades que sean detectadas tomando como punto de partida el reporte de recomendaciones entregado. Esto con el fin de asegurar el sitio web y, por consiguiente, los usuarios finales estén más seguros.

3.2 Alcance investigativo

Las bases del conocimiento permiten el descubrimiento de ideas nuevas o bien la mejora de las ya existentes sobre un tema. Es necesario por lo tanto llevar a cabo estudios exploratorios para realizar posteriormente un análisis con el fin de describir características y funciones del tema en investigación, haciendo uso de evaluaciones y recolección de datos, y obtener como resultado un proyecto con alcance descriptivo.

3.3 Enfoque

El enfoque de esta investigación es cualitativo, ya que la interpretación de la información es la clave para realizar un adecuado análisis de los resultados obtenidos, y con base en estos resultados poder establecer un criterio de experto para aplicando la metodología de OWASP poder llegar a las conclusiones del estado de las páginas web a analizar. Se pretende hacer uso de

técnicas de triangulación de datos como control y evaluación haciendo uso de una gran variedad de fuentes de información.

3.4 Diseño

El proyecto está basado en realizar una investigación evaluativa, para la búsqueda de vulnerabilidades en las páginas web de los ministerios del Gobierno Central de Costa Rica. Las páginas Web fueron seleccionadas en conjunto con el director del CSIRT basándose en la necesidad del gobierno central de Costa Rica. Las herramientas que se utilizarán han sido seleccionadas basándose en la accesibilidad, experiencia del estudiante en el uso y la confiabilidad de los resultados entregados por estas herramientas.

3.5 Población y Muestreo

Debido al enfoque del proyecto, no es necesario aplicar las técnicas tradicionales de población y muestreo.

3.6 Instrumentos de Recolección de Datos

Los instrumentos de recolección de datos a utilizar en esta investigación son:

Entrevista:

Entrevista al líder del Capítulo de OWASP para Costa Rica con el fin de lograr una comunicación y construcción conjunta de significados sobre el tema.

La entrevista, como una forma de interacción social, consiste “en establecer un diálogo donde una de las partes busca obtener información y la otra sirve de fuente” (Sabino, 1985, p.175).

Ejemplo de la forma de entrevista en los anexos.

3.7 Técnicas de observación

La Observación directa es el “uso sistemático de nuestros sentidos en la búsqueda de los datos que necesitamos para resolver un problema de investigación” (Sabino, 1985, p.155).

3.8 Técnicas de Análisis de la Información

Para poder analizar toda la información obtenida de una mejor manera, se van a aplicar tres técnicas que, además de brindar un análisis acertado, van a brindar una estructura de estos datos de manera organizada y entendible para los lectores, ya sea de manera gráfica o bien en prosa.

- Mapas Conceptuales: Con esta técnica se va a representar gráficamente el contenido del texto, con una estructura jerárquica y seleccionando los conceptos más importantes relevantes para este proyecto.
- Diagramas de Flujo de Datos: Por medio de los diagramas de flujo se pretende entender visualmente el proceso que lleva realizar un análisis de vulnerabilidades en las páginas web.
- Diagramas de red: Por medio de los diagramas de red se pretende entender las diferentes etapas del proceso de detección de vulnerabilidades, así como conceptos que puedan facilitar el entendimiento de la información.

CAPÍTULO IV

DESARROLLO DE LA PROPUESTA DE SOLUCIÓN

4.1 Estructura de los sitios web

Para poder iniciar determinando el alcance de los sitios web, se debió realizar una revisión general de la estructura de estos sitios, determinar la cantidad de páginas y tipo de contenido que poseen estas, la cantidad de componentes e información que procesan.

Para esto se determinó que una herramienta muy valiosa para este propósito es ZAP de OWASP, la cual realiza una revisión de todo el sitio web y crea una especie de estructura tipo árbol, donde despliega todas las carpetas y subcarpetas del sitio web. Además de esto se utilizó el sitemap del sitio web.

4.1.1 ¿Qué es un sitemap?

Un sitemap, o mapa del sitio web, es un archivo en el que los administradores del sitio web enumeran las páginas y rutas del sitio para informar a los motores de búsqueda como Google de la distribución y organización del mismo.

En esta sección solo se estará desplegando la información de 1 sitio web, esto tomando en cuenta que la estructura de los sitios es de muy gran tamaño y cambiante. Es por este motivo que el sitio web elegido es el sitio del ministerio de comercio exterior de Costa Rica.

La **figura 1** muestra en detalle la estructura de uno de los sitios web analizados, únicamente se ejemplifica con un sitio debido a lo extenso de los mismos, esto tomando en cuenta la extensa cantidad de recursos que cada ruta de acceso del sitio contiene, tales como formularios de login, imágenes, archivos de código que ejecutan funciones en la página, etc. La referencia se puede ver en el siguiente enlace: http://www.comex.go.cr/sitemap.xml.

Figura 1. Mapa del sitio web del Comex

Ruta del sitio web	N° Recursos
www.comex.go.cr	38
www.comex.go.cr/acerca_comex/	15
www.comex.go.cr/archivo/	1
www.comex.go.cr/archivo/canada/	2
www.comex.go.cr/archivo/caricom/	1
www.comex.go.cr/archivo/centroamerica/	2
www.comex.go.cr/archivo/comunicados_prensa/	123
www.comex.go.cr/archivo/panama/	1
www.comex.go.cr/archivo/singapur/	2
www.comex.go.cr/banners/bannersportada/	2
www.comex.go.cr/banners/images/	2
www.comex.go.cr/blog/	3
www.comex.go.cr/buscador/	2
www.comex.go.cr/cadenasglobalesdevalor/	2
www.comex.go.cr/cadenasglobalesdevalor/galeria/	3
www.comex.go.cr/cadenasglobalesdevalor/sesion1/	1
www.comex.go.cr/cadenasglobalesdevalor/sesion2/	1
www.comex.go.cr/cadenasglobalesdevalor/sesion3/	1
www.comex.go.cr/cadenasglobalesdevalor/sesion4/	1
www.comex.go.cr/cadenasglobalesdevalor/sesion5/	1
www.comex.go.cr/CAFTADR/	1

www.comex.go.cr/CAFTADR/galeria/	3
www.comex.go.cr/CanalesRSS/	6
www.comex.go.cr/capitulo 7/	1
www.comex.go.cr/chat/	3
www.comex.go.cr/China LAC/galeria/	2
www.comex.go.cr/ckeditor/	1
www.comex.go.cr/Consultas/	4
www.comex.go.cr/css/	2
www.comex.go.cr/downloads/	1
www.comex.go.cr/error_404/	1
www.comex.go.cr/estadisticas/	7
www.comex.go.cr/estadisticas/Analisis/	2
www.comex.go.cr/estadisticas/aranceles/	1
www.comex.go.cr/estadisticas/balanza_comercial/	1
www.comex.go.cr/estadisticas/ied	1
www.comex.go.cr/estadisticas/otras/	1
www.comex.go.cr/estadisticas/SIAC/	22
www.comex.go.cr/eventos/	9
www.comex.go.cr/eventos/AACUE/	1
www.comex.go.cr/eventos/Actividades/	5
www.comex.go.cr/eventos/OCDE/	2
www.comex.go.cr/eventos/TallerEFTA/	1
www.comex.go.cr/hae/	1

www.comex.go.cr/images/	3
www.comex.go.cr/Informe final 20102014/	3
www.comex.go.cr/joinnow/	1
www.comex.go.cr/MIP/	2
www.comex.go.cr/MIP/presentacion/	5
www.comex.go.cr/MIP/video/	7
www.comex.go.cr/newContentsNotificator/	1
www.comex.go.cr/OCDE/	13
www.comex.go.cr/OCDE/actividadesyeventos/	1
www.comex.go.cr/OCDE/ComunicadosdePrensa/	1
www.comex.go.cr/OCDE/Contactos/	1
www.comex.go.cr/OCDE/documentos/	1
www.comex.go.cr/OCDE/Galeria/	1
www.comex.go.cr/OCDE/informe/	1
www.comex.go.cr/OCDE/Lineas Directrices/	1
www.comex.go.cr/OCDE/preguntasfrecuentes/	2
www.comex.go.cr/OCDE/publicaciones/	1
www.comex.go.cr/OCDE/SBS Costa Rica Espanol (PDF interactivo)/	1
www.comex.go.cr/politica_comercio/	7
www.comex.go.cr/politica_comercio/plan_accion/	1
www.comex.go.cr/politica_inversion/	2
www.comex.go.cr/politicas_comercio_inversion/	3

www.comex.go.cr/pruebas/	13
www.comex.go.cr/pruebas/invitacion/	1
www.comex.go.cr/pruebas/noticias/	2
www.comex.go.cr/pruebas/Pages/	1
www.comex.go.cr/pruebas/revista/	1
www.comex.go.cr/sala_prensa/	30
www.comex.go.cr/sala_prensa/boletines/	62
www.comex.go.cr/sala_prensa/comunicados/	1
www.comex.go.cr/sala_prensa/Discursos/	1
www.comex.go.cr/sala_prensa/imagenes/	14
www.comex.go.cr/sala_prensa/presentaciones/	1
www.comex.go.cr/sala_prensa/videos/	1
www.comex.go.cr/Tabs_Index/	1
www.comex.go.cr/tramites_servicios/	10
www.comex.go.cr/tramites_servicios/aacue eur1/	2
www.comex.go.cr/tramites_servicios/Certificadosdeorigen/	1
www.comex.go.cr/tramites_servicios/Comite institucional de servicios/	3
www.comex.go.cr/tramites_servicios/Comite institucional/	3
www.comex.go.cr/tramites_servicios/empresas/	1
www.comex.go.cr/tramites_servicios/Encuesta/	2
www.comex.go.cr/Transparencia/	4
www.comex.go.cr/Transparencia/Bienes y gastos fijos/	3

www.comex.go.cr/Transparencia/Compras y contrataciones/	3
www.comex.go.cr/Transparencia/datos abiertos/	3
www.comex.go.cr/Transparencia/Informes institucionales/	3
www.comex.go.cr/Transparencia/Jerarcas y decisiones/	3
www.comex.go.cr/Transparencia/Participacion ciudadana/	3
www.comex.go.cr/Transparencia/Planes_institucionales/	3
www.comex.go.cr/Transparencia/Presupuestos/	3
www.comex.go.cr/Transparencia/Recursos humanos/	3
www.comex.go.cr/tratados/	1
www.comex.go.cr/tratados/Casos/	2
www.comex.go.cr/tratados/multilaterales/	1
www.comex.go.cr/tratados/proceso_legislativo/	1
www.comex.go.cr/tratados/proceso_negociacion/	3
www.comex.go.cr/tratados/Resoluciones/	3
www.comex.go.cr/tratados/vigentes/	13
Total	**548**

4.2 Identificar la estructura de los sitios web

En las distintas plataformas que se utilizan en los sitios web del gobierno central de Costa Rica, se han identificado diferentes sistemas y herramientas de gestión de contenido web de uso común, entre las que se pueden mencionar:

- Microsoft IIS versiones 7.5 y 8.5
- Apache
- NGINX

- Cloudflare-nginx
- WordPress 3.6 y 4.8
- Joomla 2.5.9 y 3.7.3
- Drupal

La mayoría de los sitios se encuentran alojados en Costa Rica, ya sea dentro de un centro de datos local o bien en las propias oficinas de la institución. El único caso en que el sitio se encuentra alojado fuera del país es el Ministerio de Turismo, específicamente en un servidor en Estados Unidos.

4.3 Identificar a los interesados

Entre los interesados de los distintos departamentos de TICs que soportan las operaciones y el mantenimiento de los sitios web del gobierno central de Costa Rica, se han identificado los siguientes:

- Gerentes de TI
- Encargados de redes
- Gerentes de Seguridad
- Gerentes de infraestructura
- Directores
- Colaboradores de TI

Es importante mencionar que lo anterior responde a los distintos roles encontrados en cada ministerio, a continuación se detallan los interesados de acuerdo con el puesto.

Figura 2. Lista de interesados en cada Ministerio

Puesto	Institución	Encargado
Gerente de TI	Presidencia	Cristian Durán Mora
	MINAE	Macario Ramírez Cascante
	COMEX	Javier Durán Fallas
	MCJ	Catalina Cabezas Bolaños
	MGP	Martha Carvajal Angulo
	MJP	Marianella Granados Saavedra
	MTSS	Daniel Sáenz Hernández
	MIVAH	Lilliana Reyes Pinna
	INAMU	Ingrid Trejos Marín
Encargado de redes	Presidencia	Benjamín Calderón Alvarado
	ICT	Karen Hernández Bonilla
Gerente de Seguridad	MICIT	Johnny Pan Sanabria
	MIDEPLAN	Grettel Mora Solano
	Seguridad Pública	Fanny Redondo
		Eduardo Víquez Sánchez

	MTSS	Leda Hernández Cordero
	MJP	Jonathan Barquero Durán
Gerente de infraestructura	MCJ	Wagner Chavarría Salazar
	MJP	Luis A. Montes Chacón
	INAMU	Jonathan Zúñiga Alvarado
Director	MEP	Iván Rojas Alvarez
	Seguridad Pública	Jaqueline Sequeira Torres
Colaborador de seguridad de TI	Hacienda	Carlos Ferreira Jiménez
		Mauricio Leandro Alpízar
		Walter Acosta Cortés

4.4 Herramientas utilizadas y método de análisis

Para llevar a cabo el estudio y análisis de las vulnerabilidades existentes, es necesario contar con una serie de herramientas que faciliten el proceso, así como también es de mucha importancia que exista un respaldo de conocimiento ya sea práctico o bien académico en relación con el tema. Por lo tanto, es indispensable estudiar y entender las características de las diferentes técnicas y herramientas disponibles, que servirán como punto de referencia para posteriormente escoger las mejores. En la siguiente sección se detallan algunas de las principales y más conocidas actualmente.

4.4.1 Acunetix: La empresa Acunetix fue fundada en 2004 con el fin de combatir el incremento en los ataques en aplicaciones Web. Acunetix es líder del mercado. El escáner de vulnerabilidades Web de Acunetix es una herramienta de seguridad de aplicaciones Web automatizada, la cual es capaz de escanear cualquier sitio Web o aplicación Web que sea accesible a través del protocolo HTTP / HTTPS. Esta herramienta comprueba diferentes tipos de vulnerabilidades como inyección de código SQL, Cross Site Scripting y más de 3000 vulnerabilidades Web.

4.4.2 OpenVas (The Open Vulnerability Assessment System): Es una herramienta escáner de vulnerabilidades de uso libre, compuesta por diferentes servicios y herramientas, que brinda la posibilidad de realizar la identificación y corrección de fallas de seguridad. OpenVas realiza gestión de vulnerabilidades, escaneo de vulnerabilidades y categorización, así como pruebas de vulnerabilidades de red, las cuales comprueban la presencia de un problema de seguridad específico conocido o potencial en los sistemas.

4.4.3 ZAP: Es una herramienta de pruebas de penetración facilitada por la comunidad de OWASP, con el fin de encontrar vulnerabilidades en aplicaciones web. ZAP es de código abierto y gratuito, provee escaneos automáticos, así como una variedad de herramientas que permiten encontrar de forma manual vulnerabilidades. Entre algunas de sus funcionalidades posee un proxy tipo hombre en el medio, escáner de vulnerabilidades, detalles de la conexión de web sockets, así como la posibilidad de modificar y reenviar llamados a sitios web.

4.4.4 Nessus: Es una herramienta de la empresa Tenable que nació en el año 1998 con el objetivo de proveer a la comunidad de internet con un escáner de vulnerabilidades gratis, en el 2005 pasó a ser de paga. Se utiliza como un escáner de vulnerabilidades, que permite realizar escaneo de equipos, revisión de configuración, análisis de actualizaciones faltantes en equipos, vulnerabilidades conocidas y vulnerabilidades del Top 10 de OWASP. Nessus previene ataques al

identificar vulnerabilidades y problemas de configuración utilizados por los atacantes para ingresar a los equipos y redes.

4.4.5 Nexpose Vulnerability Manager: Es una solución de Rapid 7, para realizar gestión de vulnerabilidades, la cual combina la evaluación de vulnerabilidades y controles, así como la planificación de remediación. Maneja estándares de riesgo, vulnerabilidades y mejores prácticas en cuanto a la configuración que debe seguirse para cumplir con estándares de la industria como PCI DSS, NERC CIP, FISMA, entre otros.

4.4.6 FOCA (Fingerprinting Organizations with Collected Archives): Es una herramienta utilizada principalmente para encontrar metadatos e información oculta en los documentos que examina, estos documentos pueden estar en páginas web, resulta entonces que a través de FOCA se pueden descargar y analizar. Tiene la capacidad de analizar una variedad de documentos como archivos de Microsoft Office, Open Office, o ficheros PDF, aunque también analiza ficheros de Adobe InDesign, por ejemplo.

Estos documentos se buscan utilizando tres sitios conocidos como lo son Google, Bing y DuckDuckGo. La suma de los tres buscadores hace que se consiga un gran número de documentos. FOCA une la información, tratando de reconocer qué documentos han sido creados desde el mismo equipo, y qué servidores y clientes se pueden inferir de ellos. Entre sus funcionalidades posee un módulo de descubrimiento de servidores, cuyo objetivo es automatizar el proceso de búsqueda usando técnicas enlazadas de forma recursiva.

Con FOCA es posible también reconocer las tecnologías del sitio web, buscar archivos importantes, encontrar archivos con extensiones que sean utilizadas para respaldo de base de datos, encontrar archivos de configuración que puedan exponer información sensible, búsqueda de directorios abiertos que deben estar con acceso restringido.

4.4.7 SOAPUI: Es una herramienta de código abierto, creada con el objetivo de realizar pruebas de APIs. Ofrece funcionalidad para probar Servicios Web SOAP, cubre también servicios Web WSDL, cuenta con más de 10 años de experiencia y apoyo de una gran comunidad de desarrolladores. Es un método para probar la calidad de los servicios Web, así como para realizar consultas a estos servicios web para probar el funcionamiento de los servicios web y métodos de este servicio, así como para un usuario ejecutar las consultas y consumir la información que devuelve.

4.4.8 Edit This cookie: Es un gestor de cookies gratuito, que sea agrega como una extensión al buscador. Creado por Francesco Capano, con el fin eliminar, editar, buscar, proteger y bloquear cookies. Fue inspirado por la falta de un buen gestor de cookies en el buscador Google Chrome.

Entre sus funcionalidades están:

- Borrar un cookie.
- Editar un cookie.
- Agregar un cookie.
- Buscar un cookie.
- Proteger un cookie (acceso de lectura solamente).
- Bloquear un cookie.
- Limitar la fecha máxima de expiración de una cookie.

4.4.9 NMAP ("Network Mapper"): Creado en 1997 por Gordon Lyon, es un programa gratuito y de código abierto para descubrimiento de redes y auditoría de seguridad. Muchos administradores de redes y sistemas lo encuentran útil también para tareas como inventario de la red, manejar actualizaciones, y monitoreo de equipos. Nmap utiliza funcionalidades para determinar qué equipos están disponibles en la red, qué servicios estos equipos ofrecen, qué

Sistema operativo tienen, qué tipo de filtros/firewalls usa y docenas de otras características. Fue diseñado con el objetivo de rápidamente escanear redes grandes, pero funciona bien sobre computadoras individuales. Nmap se puede ejecutar en la mayoría de los sistemas operativos. Fue nombrado "Producto de seguridad del año" por el Linux Journal, Info World, LinuxQuestions.org.

4.4.10 SQLMAP: Es una herramienta de código abierto cuyo lenguaje de programación utilizado es Python, el objetivo de esta herramienta es detectar y realizar ataques de Inyección de código SQL de forma automática. Su objetivo es también detectar y aprovechar las vulnerabilidades de inyección SQL en aplicaciones web. Una vez que es detectada la vulnerabilidad, el usuario puede elegir entre una variedad de opciones, entre ellas, enlistar los usuarios de la base de datos, obtener los hashes de las contraseñas, verificar los privilegios del usuario en la base de datos o extraer por completo la base de datos.

4.4.11 BURPSUITE: Es una herramienta de pruebas de penetración Web. Es utilizada por muchos profesionales de seguridad de la información para identificar vulnerabilidades y verificar vectores de ataques para aplicaciones web.

Puede ser clasificado como un Proxy de interceptación, donde un usuario configura su navegador de internet para que todo el contenido web de su computadora sea analizado a través de la herramienta Burp Suite antes de salir a internet. Lo anterior le permite a BurpSuite estar en el medio de la comunicación de la computadora del usuario con el internet, actuando de esta manera como un hombre en el medio, capturando y analizando cada información enviada o recibida por el navegador. Gracias a esta aplicación esta información puede ser modificada, se le pueden agregar parámetros, se puede reenviar y realizar análisis para determinar errores o para analizar la seguridad de la página.

4.4.12 Fiddler: Es una herramienta utilizada para analizar el tráfico http-https entre el navegador del usuario y el internet, esto se realiza con el fin de inspeccionar la comunicación y poder analizar tanto lo que se envía desde el navegador como lo que responde la página web que se está accediendo en internet. A través de estos análisis se determinan errores en páginas web, exposición de información, tiempos de respuesta, etc. Funciona como hombre en el medio. Además de esto, posee una funcionalidad que le permite reenviar llamados modificando tanto sus parámetros como el método http (POST, GET, PUT, DELETE).

4.4.13 Google Chrome navegador y Chrome DevTools: Es un navegador web creado por la empresa Google en el año 2008, entre sus funcionalidades posee las herramientas de desarrollador o DevTools que posee diferentes funcionalidades, como por ejemplo el "Network panel" que registra todos los llamados que se realizan a través del browser, así como los tiempos de respuesta de cada llamado, el método utilizado y el tipo de llamado.

4.5 Justificación de las herramientas

Posterior al estudio de las distintas herramientas de trabajo que potencialmente pueden ser bien aceptadas para llevar a cabo el análisis de vulnerabilidades requerido en los sitios web, se destacan las siguientes como las herramientas principales que han sido seleccionadas.

- **ZAP**
- **Acunetix**
- **Nessus**
- **EditThisCookie**
- **FOCA**
- **SQLMAP**
- **NMAP**
- **FIDDLER**

- **Google Chrome/ Chrome DevTools**

Entre los criterios de selección utilizados a continuación se detallan las principales razones que llevaron a esta decisión.

A. **Recomendación de herramienta por parte de la metodología utilizada**

Muchas de estas herramientas son recomendadas en la metodología de OWASP elegida como la metodología de análisis para el desarrollo de la investigación. Como ejemplo se puede mencionar ZAP, la cual es recomendada por la metodología de OWASP en sus diferentes apartados.

El contenido de las guías y estándares es avalado por los actores más importantes de la industria, las revisiones periódicas, generación de nuevos lanzamientos de versiones y los proyectos son un caso de estudio claro del éxito que se puede alcanzar cuando las personas trabajan en colaboración.

Otro aspecto importante es que el contenido (guías, hojas de trucos, herramientas, código fuente, etc.) ha sido desarrollado por personas con muchos años de experiencia lidiando con problemas de Seguridad y que proporcionan un enfoque holístico para remediar estos riesgos.

B. **Recomendación de herramienta por parte del líder de OWASP para Costa Rica**

Otro elemento importante para la toma de decisiones en relación con la escogencia de las herramientas fue la recomendación del experto y líder del Capítulo de OWASP para Costa Rica, Michael Hidalgo, por ejemplo, el uso de Acunetix y ZAP.

El rol que cumple el líder del capítulo local de OWASP Costa Rica, es tener una representación local de la fundación OWASP en el país para crear un ecosistema donde se fomenten las mejores prácticas del desarrollo seguro de software. Algunas de las actividades que se desarrollan son:

- Planear y organizar eventos y conferencias abiertas al público en general donde se comparta conocimiento, mejores prácticas, nuevos vectores de ataques, guías y estándares de la industria, herramientas y código fuente para mejorar la seguridad de las aplicaciones.
- Fomentar los valores de OWASP en la región para crear una cultura de sensibilización sobre los riesgos que afectan nuestra infraestructura.
- Promover en el país y la región Latinoamericana las mejores prácticas del desarrollo seguro para facultar a las empresas y organizaciones a desarrollar aplicaciones que sean resistentes a ataques informáticos actuales.
- Crear un ecosistema donde se pueda compartir conocimiento sobre ataques informáticos y sus métodos de prevención y detección.
- Organizar talleres prácticos donde se pueda contextualizar, desde un enfoque teórico-práctico, los ataques informáticos más difundidos de forma global, y proponer enfoques inteligentes para contrarrestar estas amenazas cibernéticas.
- Promover las bondades que ofrece la comunidad de código abierto (del inglés Open Source), de forma que tanto el sector público como privado se puedan beneficiar con un enfoque educativo.

C. **Conocimiento previo por parte de los estudiantes**

El conocimiento previo que se tiene en relación con las herramientas seleccionadas tiene una gran importancia y peso para la elección de las herramientas, debido a que son utilizadas a diario en los lugares de trabajo, así también como el conocimiento adquirido de estas herramientas durante el proceso de aprendizaje en la maestría. Por lo anterior es evidente que la curva de aprendizaje para el manejo de la herramienta se reduce enormemente, la interpretación y

depuración de resultados es más dinámica, debido a que se maneja de forma profesional en el día a día.

4.6 Interpretación de las pruebas

Aplicada la metodología que ha sido seleccionada, se procede a la interpretación de cada una de las principales vulnerabilidades que han sido encontradas a través de la investigación.

4.6.1 Desarrollo de la metodología para pruebas de penetración de sitios web

Como se mencionó en la sección 2.8, la metodología que se va a utilizar es el Testing Guide 4.0 de OWASP, por medio de pruebas de Caja Negra cuyo conocimiento de la información a evaluar antes de realizar algún descubrimiento por parte del estudiante es ninguna o muy escasa. El método híbrido permite un mayor control de las pruebas debido a que los resultados que se obtienen están basados en pruebas tanto manuales como automáticas. A continuación se muestran los resultados obtenidos al haber aplicado la técnica seleccionada.

Recolección de datos

Recordemos que las pruebas de seguridad para la recolección de datos consisten en dos fases:

La fase pasiva: En esta fase el experto que ejecuta las pruebas intentará entender la lógica de la aplicación, utilizando la aplicación como un usuario normal tratando de entender su funcionamiento. Para esta fase el experto puede utilizar herramientas como un HTTP proxy para interceptar todos los llamados y poder entender el funcionamiento de la aplicación, sus interacciones con otros servidores, los parámetros que envía, los encabezados HTTP.

La fase activa: En esta fase el experto empieza las pruebas usando la metodología descrita en el capítulo anterior de acuerdo con las 11 subcategorías ya mencionadas.

4.6.2 Calificación de Riesgo

El procedimiento para calcular el riesgo y niveles de cada vulnerabilidad está basado en el CVSS v3.0 del inglés (Common Vulnerability Scoring System), el cual es un sistema de puntuación diseñado con el fin de proveer un método abierto y estándar que permita ser utilizado para estimar el impacto de una vulnerabilidad. Basándose en esta calificación, se ha realizado una escala para establecer la correspondencia cualitativa a un valor cuantitativo.

Para determinar este valor, es necesario ingresar variables específicas de cada vulnerabilidad, por medio de una calculadora disponible en el sitio web ***https://www.first.org/cvss/calculator/3.0***; se consideran para el cálculo final las siguientes variables:

- (AV) Vector de ataque
- (AC) Complejidad de ataque
- (PR) Privilegios requeridos
- (UI) Interacción de usuario
- (S) Alcance
- (C) Confidencialidad
- (I) Integridad
- (A) Disponibilidad

La puntuación de las métricas de CVSS producen una cadena de vectores, esta es una representación textual de los valores de las métricas utilizadas para calificar la vulnerabilidad.

La cadena de vectores ***v3.0*** comienza con la etiqueta "**CVSS:**" junto a una representación numérica de la versión actual, "**3.0**". La información mostrada posteriormente a la versión es un conjunto de métricas, cada métrica está precedida por una barra inclinada, **"/"**, que actúa como

un delimitador. Los indicadores de las variables van abreviados como se enumeraron anteriormente, seguido de dos puntos "**:**" con el valor de la métrica asignado.

Para este propósito se ha utilizado la calculadora de CVSS, el valor desplegado será la calificación de riesgo de la vulnerabilidad. Esta cadena de vectores ingresados se despliegan como parámetros en el URL (en el querystring), a continuación se muestra un ejemplo de la cadena de vectores.

CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:C/C:H/I:H/A:H

Por lo tanto, si se requiere hacer una validación del puntaje obtenido, es posible hacerlo ingresando los valores de cada vulnerabilidad en la calculadora, o bien es posible también concatenar al URL ***https://www.first.org/cvss/calculator/3.0#*** los valores asignados a cada vulnerabilidad. Un ejemplo es la siguiente forma:

https://www.first.org/cvss/calculator/3.0#CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:C/C:H/I:H/A:H

El valor que se ha obtenido para el análisis de cada vulnerabilidad será incluido en la descripción de cada una, en la sección "**Análisis de la situación al realizar las pruebas de penetración"** descrita en la sección **4.6.3**.

Escala de puntuación de riesgo

La escala de puntuación que ha sido utilizada a lo largo de esta investigación es la que se muestra en la **Figura 3** y que detalla los niveles de riesgo, el rango de puntuación obtenido de **CVSS v3.0**, con su respectivo código de color por nivel de riesgo, mostrado en orden descendente del nivel crítico hasta el informativo.

Figura 3. Escala de puntuación de riesgo

Nivel	Rango de Puntuación (**CVSS v3.0**)	Código de color
Crítico	7.0 hasta 10.0	Crítico
Medio	4.0 hasta 6.9	Medio
Bajo	0.1 hasta 3.9	Bajo
Informativo	0.0	Informativo

4.6.3. Análisis de la situación al realizar las pruebas de penetración

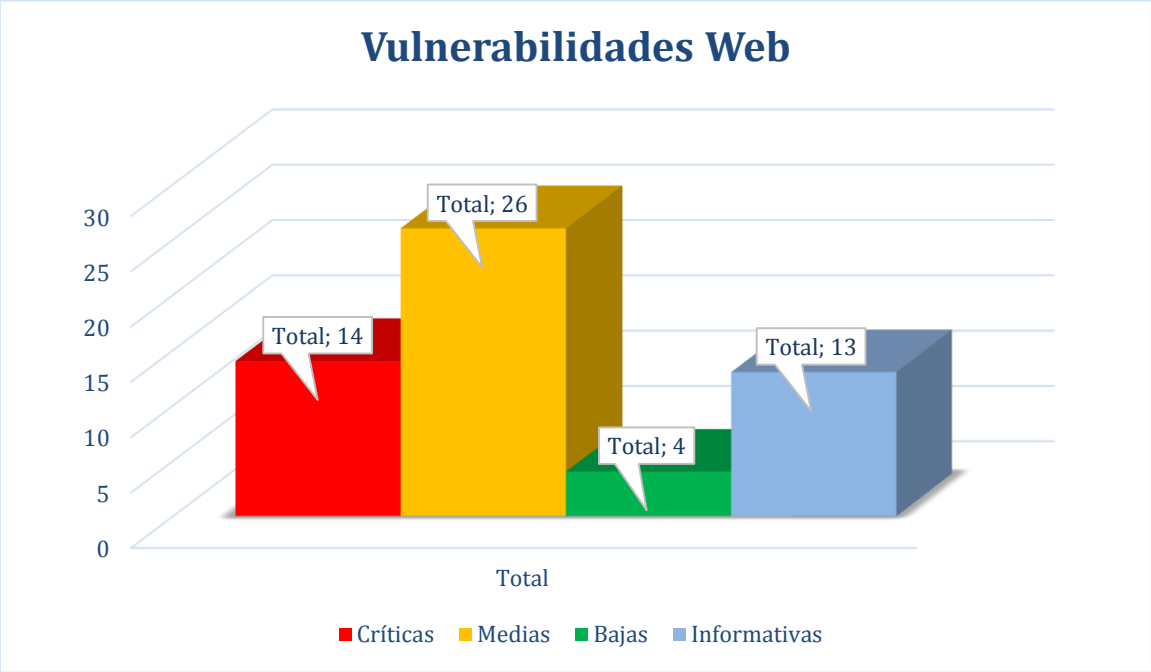
Resumen ejecutivo

El siguiente es un resumen general que detalla cada una de las vulnerabilidades por categoría que han sido encontradas durante esta evaluación, por lo tanto los datos mostrados en la **Figura 4** son datos que consideran únicamente una vez cada vulnerabilidad y representan el total encontrado en los 16 sitios web analizados.

Figura 4. Total de vulnerabilidades por cada categoría

Nivel de riesgo	**CVSS**	**Total**
Crítico	7.0-10	14
Medio	4.0-6.9	27
Bajo	0.1-3.9	4
Informativo	0.0	13
TOTAL GENERAL		**58**

Figura 5. Total de vulnerabilidades por nivel de riesgo



4.6.3.1. Inyección de código SQL

Inyección de código SQL es una vulnerabilidad que permite a un atacante ejecutar código SQL en la base de datos al manipular el código ingresado en un campo de texto. La vulnerabilidad de inyección de código SQL ocurre cuando las aplicaciones web aceptan el texto ingresado por el usuario y lo ejecutan directamente como una sentencia de código SQL, sin filtrar debidamente el código ingresado.

Factor de riesgo: Crítico

Puntuación CVSSv3.0: 10.0

Cadena de vectores: CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:C/C:H/I:H/A:N

Solución: Utilizar un mecanismo de validación en el lado del servidor, así como implementar una lista segura de valores que la aplicación debe aceptar para ese campo específico y descartar los que no debe aceptar; utilizar también un mecanismo de validación del lado del cliente. Filtrar caracteres peligrosos o dañinos como ´ , ¨ ,\,(,),:,;,<,>,[,],+,*,?,-,% . Realizar una verificación del

tamaño del campo y dato ingresado para asegurarse de que el valor enviado es correcto, por ejemplo una fecha puede variar de 6 caracteres (1/1/17) a 10 caracteres (20/12/2017). Realizar también una revisión del tipo de campo esperado, por ejemplo si la aplicación espera solo letras en un campo específico, entonces verifique que solo letras son ingresadas.

4.6.3.2. ISC BIND < 4.9.8 / 8.2.3 Multiple Remote Overflows

El servidor BIND remoto, de acuerdo con su número de versión, se ve afectado por varias vulnerabilidades de desbordamiento de búfer que pueden permitir que un atacante ejecute código arbitrario en el sitio web.

Factor de riesgo: Crítico

Puntuación CVSSv3.0: 10.0

Cadena de vectores: CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:C/C:H/I:H/A:H

Solución: Se debe actualizar a la versión 8.2.3 o 4.9.8 de BIND

4.6.3.3. ISC BIND < 4.9.5 DNS Resolver Functions Remote Overflow(CRITICAL)

El servidor BIND remoto, de acuerdo con su número de versión, es vulnerable a un desbordamiento de búfer remoto dentro de su código de resolución.

Un atacante puede ejecutar código arbitrario haciendo que el servidor DNS remoto realice una solicitud y envíe una respuesta DNS maliciosa con un campo de longitud no válida.

Factor de riesgo: Crítico

Puntuación CVSSv3.0: 10.0

Cadena de vectores: CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:C/C:H/I:H/A:H

Solución: Se debe actualizar a la versión 4.9.5 o más reciente de BIND.

4.6.3.4. Múltiples vulnerabilidades en el servidor de Oracle GlassFish 3.1.2.x < 3.1.2.15

El Oracle GlassFish Server que se ejecuta en el sitio remoto es 3.1.2.x anterior a 3.1.2.15. Por lo tanto, se ve afectado por múltiples vulnerabilidades:

Existe una vulnerabilidad de divulgación de información en la versión empaquetada de libcurl en la función smb_request_state (). Un atacante remoto no autenticado puede explotar esto, a través de un servidor SMB malicioso, para revelar contenidos de memoria arbitrarios (CVE-2015-3237).

Existe una falla no especificada en el subcomponente Web Container que permite que un atacante remoto no autenticado ejecute código arbitrario (CVE-2016-3607).

Factor de riesgo: Crítico

Puntuación CVSSv3: 10.0

Cadena de vectores: CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:C/C:H/I:H/A:H

Solución: Actualice a Oracle GlassFish Server versión 3.1.2.15 o posterior, tal como se menciona en el aviso de actualización de Oracle Critical Patch de julio de 2016.

4.6.3.5. ISC BIND < 4.9.7-REL / 8.2.2-P5 Multiple Remote Vulnerabilities

El servidor BIND remoto, de acuerdo con su número de versión, es vulnerable a varios ataques que podrían permitir a un atacante ejecutar código arbitrario en el sitio web.

Factor de riesgo: Crítico

Puntuación CVSSv3.0: 10.0

Cadena de vectores: CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:C/C:H/I:H/A:H

Solución: Se debe actualizar a la versión 8.2.2-P5 / 4.9.7-REL de BIND.

4.6.3.6. ISC BIND < 4.9.7 / 8.1.2 Inverse-Query Remote Overflow

El servidor BIND remoto, de acuerdo con su número de versión, es vulnerable a un desbordamiento de consulta inversa que podría permitir a un atacante ejecutar código arbitrario en el sitio web.

Factor de riesgo: Crítico

Puntuación CVSSv3.0: 10.0

Cadena de vectores: CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:C/C:H/I:H/A:H)

Solución: Se debe actualizar a la versión 8.1.2 o 4.9.7 de BIND o la más nueva.

4.6.3.7. OpenSSL 'ChangeCipherSpec' vulnerabilidad de MiTM

El servicio OpenSSL es vulnerable a un ataque conocido como Man in the Middle Attack (MiTM), el atacante puede forzar el uso de elementos débiles de generación de claves en clientes y servidores OpenSSL SSL/TLS. De esta forma el atacante puede descifrar y modificar el tráfico desde el cliente y el servidor atacado.

¿Qué versiones de OpenSSL poseen esta vulnerabilidad?

Los clientes de OpenSSL son vulnerables en TODAS las versiones de OpenSSL.

Se sabe que los servidores solo son vulnerables en versiones OpenSSL 1.0.1 y 1.0.2-beta1.

Otra vulnerabilidad que posee es que la función 'ssl3_read_bytes' permite que inyecte datos en otras sesiones o permitir ataques de Denegación de Servicio.

Factor de riesgo: Crítico

Puntuación CVSSv3.0: 9.8

Cadena de vectores: CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H

Solución: A los usuarios de servidores con OpenSSL anterior a 1.0.1 se les recomienda realizar la actualización como medida de precaución.

Usuarios de OpenSSL 0.9.8 SSL/TLS (cliente y/o servidor) deben actualizar a 0.9.8za.

Usuarios de OpenSSL 1.0.0 SSL/TLS (cliente y/o servidor) deben actualizar a 1.0.0m.

Usuarios de OpenSSL 1.0.1 SSL/TLS (cliente y/o servidor) deben actualizar a 1.0.1h

4.6.3.8. Aplicación Web desactualizada

Se detecta que componentes utilizados en la página web se encuentran una o más versiones atrás de la última versión del producto, esto conlleva a que la aplicación no cuente con los

últimos parches de seguridad, los cuales solucionan problemas de seguridad en la aplicación, por lo que posee múltiples vulnerabilidades. Entre algunas de estas aplicaciones se pueden mencionar a WordPress, Joomla y Drupal.

Factor de riesgo: Crítico

Puntuación CVSSv3.0: 9.8

Cadena de vectores: CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H

Solución: Se debe actualizar la aplicación a la versión web más reciente del respectivo sistema, se debe referir al sitio del fabricante.

4.6.3.9. Sitio Web hackeado

Generalmente el hackeo de sitios web se da por diversión, por razones políticas que los llevan a cometer un acto como este o bien por los conocidos como "script kiddies", personas con pocas habilidades técnicas que ejecutan programas desarrollados por hackers para hacer daño a sistemas, sin saber bien su funcionamiento o resultado final. Esto con el fin de impresionar a personas o a una comunidad en internet.

En este sitio web, al ingresar una referencia a un objeto o ruta inexistente, se realiza una redirección a un componente de la página que fue vulnerado y posee información de los atacantes. Lo que se espera obtener de un sitio no encontrado es un error 404, o una página que indique que el recurso solicitado no existe, pero en contraposición esta página devuelve un resultado, el cual es el componente de la página que ha sido hackeada, con el fin de desprestigiar a un gobierno y dejarle saber al mundo que han intervenido un sistema de un gobierno.

Factor de riesgo: Crítico

Puntuación CVSSv3.0: 9.8

Cadena de vectores: CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H

Solución: Es importante revisar todos los componentes a nivel de configuración del sitio web con el fin de remover las referencias a este ataque que se dio en el pasado, pero aún está infectada la página. Este tipo de ataques se da por vulnerabilidades en los gestores de contenido o componentes de la página web, es por esto que es muy importante mantener todos los componentes actualizados.

4.6.3.10. Divulgación de información en OpenSSL Heartbeat

La vulnerabilidad es un fallo en la librería de software OpenSSL que es una implementación del protocolo SSL/TLS, el cual se utiliza para proteger la privacidad de las comunicaciones. Esta vulnerabilidad permite a un atacante leer la memoria del sistema, con lo que podría facilitar el acceso a información sensible como contraseñas, claves privadas y otros datos confidenciales.

Factor de riesgo: Crítico

Puntuación CVSSv3.0: 9.4

Cadena de vectores: CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:L

Solución: Actualizar a la versión OpenSSL 1.0.1g o posterior.

4.6.3.11. Listado de directorios (Directory Listing)

Esta configuración permite que el servidor web pueda mostrar la lista de archivos contenidos en ese directorio y navegar a través de ellos. Esto no es recomendable debido a que el directorio puede contener archivos que normalmente no están expuestos a través de enlaces en el sitio web, como archivos de configuración. Se expone la confidencialidad de la información contenida en ellos, así como la exposición de datos almacenados en esos directorios, además es posible la exposición de código no ofuscado y que pueda ser vulnerado.

Factor de riesgo: Crítico

Puntuación CVSSv3.0: 7.5

Cadena de vectores: CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:N/A:N

Solución: Se recomienda deshabilitar la configuración de Listado de Directorios a nivel del servidor web, en caso de no ser posible, asegurarse de que la información contenida en los directorios expuestos no es información sensible.

4.6.3.12. Métodos inseguros habilitados

Método TRACE habilitado

Permite recibir información de vuelta hacia el cliente, monitoreando los mensajes que se dan entre el cliente y el servidor que indican al cliente de cualquier "string" que se haya enviado hacia el servidor. Su principal función es la de diagnosticar fallas o bien para verificar si existen conexiones intermedias, como por ejemplo servidores en el proceso de la comunicación. Originalmente se creía que el método era inofensivo, pero ahora se sabe que puede ser utilizado para montar un ataque conocido como "Cross Site Tracing XST", el cual fue descubierto por Jeremiah Grossman.

Método PUT habilitado

Permite al cliente subir archivos al servidor web, por lo tanto, un atacante podría explotar este método subiendo archivos maliciosos.

Método DELETE habilitado

Permite borrar archivos de un servidor web y comprometer su funcionalidad.

Factor de riesgo: Crítico

Puntuación CVSSv3.0: 7.5

Cadena de vectores: CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:N/I:H/A:N

Solución: Para poder revisar si los métodos se encuentran habilitados, se puede utilizar la aplicación FOCA, así como si se requiere comprobar que el método funciona se puede probar

utilizando la aplicación Fiddler. Si la aplicación necesita uno o más de estos métodos, es importante revisar que el uso está debidamente limitado a usuarios de confianza o usuarios aprobados. En caso de no ser necesarios, se recomienda deshabilitarlos.

4.6.3.13. Ataque de denegación de servicio XML cuadrático

Algunas versiones de WordPress y Drupal se han visto afectadas por una vulnerabilidad DDoS (denegación de servicio) en el analizador XML PHP utilizado por sus implementaciones XMLRPC. El problema radica en el analizador de expansión de entidades XML que puede causar el agotamiento del CPU y memoria, además de que la base de datos del sitio alcance la cantidad máxima de conexiones abiertas. La vulnerabilidad se denomina ataque de expansión cuadrática XML.

Factor de riesgo: Crítico

Puntuación CVSSv3.0: 7.5

Cadena de vectores: CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:N/I:N/A:H

Solución: Actualizar la versión de WordPress/Drupal a la última versión conocida o bien restringir el acceso al archivo *xmlrpc*.

4.6.3.14. SSL Protocolo versión 2 y 3 detectado

El servicio remoto acepta conexiones cifradas usando el protocolo SSL 2.0 y/o SSL 3.0. Estas versiones de SSL están afectadas por varios defectos criptográficos. Un atacante podría explotar estos defectos para llevar a cabo ataques tipo Man in the Middle (MiTM) descifrando las comunicaciones entre el servicio afectado y los clientes.

Aunque SSL/TLS tiene un medio seguro para elegir la versión más compatible del protocolo, muchos navegadores web implementan esto de una manera insegura, lo que permite al atacante degradar una conexión (como en POODLE). Por lo tanto, se recomienda que estos protocolos se deshabiliten por completo.

El NIST ha determinado que SSL 3.0 ya no es aceptable para comunicaciones seguras.

Factor de riesgo: Crítico

Puntuación CVSSv3.0: 7.5

Cadena de vectores: CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:N/A:N

Solución: Se debe deshabilitar SSL 2.0 y 3.0. Se recomienda utilizar TLS 1.1 o TLS 1.2 en conjunto con estándares de cifrado aprobado, preferiblemente.

4.6.3.15. Uso de HTTP para autenticación

Entre los protocolos más conocidos para navegar en internet tenemos el protocolo HTTP, por sus siglas en inglés "HyperText Transfer Protocol"; es el método predeterminado y más usado en internet, además utiliza el puerto 80. Cuando se navega por un sitio web se dan una serie de solicitudes y respuestas entre el navegador del cliente y el servidor donde se aloja el sitio, por lo tanto, durante ese intercambio de información es posible ver en texto plano dicha comunicación. Es por eso que no es recomendable utilizar HTTP en sitios donde se busca ingresar por medio de un usuario y contraseña, a datos privados o descarga de archivos, ya que la información podría ser fácilmente comprometida.

Factor de riesgo: Media

Puntuación CVSSv3.0: 6.5

Cadena de vectores: CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:L/I:L/A:N

Solución: Es recomendable utilizar un cifrado en la página web mediante un certificado SSL, con el fin de garantizar al usuario la seguridad de que está navegando en el sitio correcto, así como para poder proveer un cifrado entre el usuario y el servidor para que la información no pueda ser leída si es interceptada. Existen soluciones como “Let’s Encrypt”, que provee la posibilidad de cifrar un sitio web mediante un certificado SSL gratuito.

4.6.3.16. Puerto RDP 3389 abierto al público

El servicio de escritorio remoto (RDP) para servidores permite a un usuario de Windows obtener de forma remota un inicio de sesión gráfico, por lo tanto, se podría actuar como un usuario conectado localmente desde un equipo remoto. Si un atacante obtiene un nombre de usuario y contraseña válidos, este servicio podría usarse para obtener acceso adicional en el host remoto. Un atacante también puede usar este servicio para montar un ataque de diccionario contra el host remoto para intentar iniciar sesión de forma remota.

Es necesario tener en cuenta que RDP (el protocolo de escritorio remoto) es vulnerable a los ataques Man-in-the-Middle, lo que facilita a los atacantes robar las credenciales de usuarios legítimos al suplantar al servidor de Windows.

Factor de riesgo: Medio

Puntuación CVSSv3.0: 6.5

Cadena de vectores: CVSS:3.0/AV:N/AC:L/PR:H/UI:N/S:U/C:H/I:N/A:H

Solución: Se debe en la medida de lo posible deshabilitar el servicio de escritorio remoto si no está siendo utilizado, y no permitir que el servicio esté disponible a través de internet sin tener una lista de control de acceso, donde se pueda filtrar por dirección IP para otorgar acceso solo a los que lo necesitan.

4.6.3.17. Archivo inseguro crossdomain.xml

El modelo de seguridad del navegador normalmente impide que el contenido web de un dominio acceda a datos de otro dominio. Esto se conoce comúnmente como la "política del mismo origen". Los archivos de políticas de URL otorgan permisos entre dominios para lectura de datos. Estos permiten operaciones que no son permitidas por defecto. El archivo de política de

URL se encuentra, de manera predeterminada, en el directorio raíz del servidor de destino, con el nombre crossdomain.xml (por ejemplo, en www.ejemplo.com/crossdomain.xml).

Cuando se especifica un dominio en el archivo crossdomain.xml, el sitio declara que está dispuesto a permitir a que los operadores de cualquier servidor en ese dominio obtengan cualquier documento en el servidor donde reside el archivo de política. El archivo crossdomain.xml, que es implementado en un sitio web, puede estar abierto a todos los dominios con el simple uso de un asterisco "*", soportando lo que se conoce como "wildcard"

<cross-domain-policy>
<allow-access-from domain=""/>*
</cross-domain-policy>

Esta práctica es adecuada para servidores públicos, pero no debe usarse para sitios ubicados detrás de un firewall porque podría permitir el acceso a áreas protegidas. No debe usarse para sitios que requieren autenticación en forma de contraseñas o cookies. Para los sitios que utilizan de forma común la autenticación basada en cookies para acceder a datos privados o específicos del usuario, es necesario tener especial cuidado al usar archivos de políticas entre dominios.

Factor de riesgo: Medio

Puntuación CVSSv3.0: 6.5

Cadena de vectores: CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:L/I:L/A:N

Solución: Es necesario evaluar cuidadosamente qué sitios podrán hacer llamadas entre dominios. Se debe considerar la topología de red y cualquier mecanismo de autenticación que se verá afectado por la configuración o implementación de la política de dominios cruzados.

4.6.3.18. Posible exposición de información a través de WSDL

La arquitectura de servicios web puede requerir la exposición de un servicio web que contenga información sobre los servicios de acceso público y cómo los llamados de estos

servicios deberían interactuar con ellos (por ejemplo, qué parámetros esperan y qué tipos devuelven).

1. El servicio web WSDL está disponible para un público más amplio de lo previsto.
2. El servicio web WSDL contiene información sobre los métodos / servicios a los que no se debe acceder públicamente, o información sobre métodos obsoletos. Este problema se hace más probable debido a que el WSDL a menudo se genera automáticamente a partir del código.
3. La información en el servicio web WSDL ayuda a suponer nombres / ubicaciones de métodos / recursos que no deben ser de acceso público.

Factor de riesgo: Medio

Puntuación CVSSv3.0: 6.5

Cadena de vectores: CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:L/I:L/A:N

Solución: Analizar la información que expone este servicio web, en caso de que esta información no deba ser pública, evaluar la posibilidad de implementar autenticación, ya sea mediante un certificado de cliente o mediante autenticación por un identificador tipo token.

4.6.3.19. Biblioteca de JavaScript vulnerable

Se está utilizando una biblioteca JavaScript vulnerable. Se han detectado una o más vulnerabilidades para esta versión de la biblioteca JavaScript. Se ha detectado "JavaScript library jQuery versión 1.10.2".

Factor de riesgo: Media

Puntuación CVSSv3.0: 6.5

Cadena de vectores: CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:L/I:L/A:N

Solución: Se debe actualizar a la versión más reciente.

4.6.3.20. SSL Anonymous Cipher Suites Supported

El sitio web admite el uso de cifrados SSL anónimos. Si bien esto permite que un administrador configure un servicio que encripta el tráfico sin tener que generar y configurar certificados SSL, no ofrece ninguna forma de verificar la identidad del host remoto y hace que el servicio sea vulnerable a un ataque tipo Man-in-the-Middle. Si el atacante está en la misma red es mucho más fácil de explotar

Factor de riesgo: Media

Puntuación CVSSv3.0: 5.9

Cadena de vectores: CVSS:3.0/AV:N/AC:H/PR:N/UI:N/S:U/C:H/I:N/A:N

Solución: Configure nuevamente la aplicación afectada si es posible para evitar el uso de cifrados débiles.

4.6.3.21. DNS Server Spoofed Request Amplification DDoS

El servidor DNS remoto responde a cualquier solicitud. Es posible hacer consultas a los servidores de nombres (NS) de la zona raíz ('.'), y obtener una respuesta que es más grande que la solicitud original. Al suplantar la dirección IP origen, un atacante remoto puede aprovechar esta 'amplificación' para lanzar un ataque de denegación de servicio contra un servidor de terceros utilizando el servidor DNS remoto.

Factor de riesgo: Medio

Puntuación CVSSv3.0: 5.8

Cadena de vectores: CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:C/C:N/I:N/A:L

Solución: Se debe restringir el acceso al servidor DNS desde la red pública o se debe volver a configurar para que rechace dichas consultas.

4.6.3.22. DNS Server Recursive Query Cache Poisoning Weakness

Es posible consultar el servidor DNS de forma remota para buscar nombres de terceros. Si este es el servidor de DNS interno, entonces el vector de ataque puede estar limitado a empleados o invitados, si es permitido. Si se está probando un servidor de DNS remoto, entonces permite que cualquiera lo use para resolver nombres de terceros. Por lo anterior, se permite a los atacantes realizar ataques de envenenamiento de caché contra este servidor DNS. Si se permiten consultas recursivas a través de UDP, se podría usar para ataques de DDoS contra otras redes o sistemas.

Factor de riesgo: Medio

Puntuación CVSSv3.0: 5.8

Cadena de vectores: CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:C/C:N/I:N/A:L

Solución: Se debe restringir las consultas recursivas que puedan utilizar este servidor DNS.

Si usa bind 8, puede hacerlo utilizando la instrucción 'allow-recursion' en la sección 'opciones' de named.conf. Si está utilizando bind 9, puede definir una agrupación de direcciones internas utilizando el comando 'acl'. Luego dentro del bloque de opciones, se puede declarar de forma explícita:

'allow-recursion { hosts_defined_in_acl }'

4.6.3.23. Microsoft Windows Remote Desktop Protocol Server Man-in-the-Middle Weakness

La versión del Servidor de Escritorio Remoto es vulnerable a un ataque de hombre en el medio (MiTM). El cliente RDP no fuerza a realizar una validación de identidad del servidor al configurar el cifrado. Un atacante con la capacidad de interceptar el tráfico del servidor RDP puede establecer el cifrado con el cliente y el servidor sin ser detectado. Un ataque MiTM de esta naturaleza permitiría al atacante obtener cualquier información confidencial transmitida, incluidas las credenciales de autenticación. Este error existe porque el servidor RDP almacena una clave

privada RSA codificada en la biblioteca mstlsapi.dll. Cualquier usuario local con acceso a este archivo (en cualquier sistema de Windows) puede recuperar la clave y usarla para este ataque.

Factor de riesgo: Medio

Puntuación CVSSv3.0: 5.6

Cadena de vectores: CVSS:3.0/AV:N/AC:H/PR:N/UI:N/S:U/C:L/I:L/A:L

Solución:

- Forzar el uso de SSL como capa de transporte para este servicio, si es compatible.
- Seleccionar la configuración 'Permitir conexiones solo desde computadoras que ejecuten escritorio remoto con autenticación de nivel de red', si está disponible.

4.6.3.24. HTTPS no configurado

Es recomendable que el sitio web se configure para utilizar un certificado digital que le permita asegurar a los usuarios que está en la página correcta, así como la posibilidad de cifrar la comunicación de la información enviada o recibida, con el fin de evitar que sea interceptada y manipulada por un atacante.

Factor de riesgo: Media

Puntuación CVSSv3.0: 5.3

Cadena de vectores: CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:L/I:N/A:N

Solución: Es recomendable utilizar un cifrado en la página web mediante un certificado SSL, con el fin de garantizar al usuario la seguridad de que está navegando en el sitio correcto, así como para poder proveer un cifrado entre el usuario y el servidor para que la información no pueda ser leída si es interceptada. Existen soluciones como Let’s Encrypt que proveen la posibilidad de cifrar un sitio web mediante un certificado SSL gratuito.

4.6.3.25. Posibilidad de listar usuarios

Posibilidad de listar usuarios al intentar ingresar, para verificar su validez, es un tipo de ataque en el que el script de validación de la página web muestra al atacante si el nombre de usuario proporcionado es correcto o no. Explotar esta vulnerabilidad motiva al atacante a poder generar una lista de usuarios válidos en la aplicación, como prueba / prueba, administrador / administrador, invitado / invitado, etc., para así poder determinar los que son válidos, y recibe como ayuda los diferentes mensajes de error. Una vez obtenida esta lista, poder experimentar con diferentes contraseñas fáciles de adivinar

Las cuentas de uso común a menudo son creadas por los desarrolladores para fines de prueba y muchas veces las cuentas nunca se desactivan o el desarrollador se olvida de cambiar la contraseña (Symantec, 2017). Se ha descubierto en ocasiones estas cuentas no solo son comunes y tienen contraseñas fáciles de adivinar, sino que a veces también contienen información confidencial de la empresa, asociada a la cuenta de prueba, por lo que si un atacante obtiene la contraseña de esta cuenta, podrá tener acceso a esta información confidencial asociada.

Factor de riesgo: Medio

Puntuación CVSSv3.0: 5.3

Cadena de vectores: CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:L/I:N/A:N

Solución: Evitar desplegar mensajes que infieran que el usuario digitado no existe. Se recomienda revisar todas las diferentes partes de la aplicación web donde las cuentas de usuarios son validadas por el servidor, para verificar las respuestas de este, con el fin de validar que sea una respuesta estándar a la hora de no encontrar un usuario o de ingresar una contraseña incorrecta. Una respuesta diferente puede ser tan obvia como un mensaje de error o la cantidad de tiempo que el servidor toma para responder, o una diferencia mínima como una línea extra de

código en la respuesta. Además de esto se recomienda la implementación de un segundo factor de autenticación para los sitios que contengan información sensible.

4.6.3.26. Gestión de archivos

El uso de archivos de información dentro de los sitios web podría significar un alto nivel de gestión, algunas de las extensiones de archivos como excel, word, ppt, pdf, son de fácil acceso y se pueden descargar, lo que puede representar una seria amenaza si contienen información confidencial.

Para este punto es importante señalar que es probable que los archivos sean de acceso al público por la naturaleza informativa del sitio, sin embargo, es importante una gestión diligente que evite la indebida fuga de información.

Factor de riesgo: Media

Puntuación CVSS: 5.3

Cadena de vectores: CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:L/I:N/A:N

Solución: Es importante que se evalúe el contenido de los archivos que se encuentran expuestos para ser descargados por los usuarios de la página web. La herramienta FOCA puede ser de mucha ayuda para esta depuración, con el fin de enlistar todos los archivos expuestos por la página y poder depurarlos. Los siguientes links sirven de referencia para mejores prácticas según OWASP:

https://www.owasp.org/index.php/Input_Validation_Cheat_Sheet

https://www.owasp.org/index.php/Unrestricted_File_Upload

4.6.3.27. Mensaje de error en aplicación

Los mensajes de error o advertencia de la aplicación pueden exponer información sensible sobre el funcionamiento interno de una aplicación a un atacante. Esta vulnerabilidad fue

encontrada en distintos sitios, fue necesario realizar una búsqueda en distintas secciones de los sitios para confirmar su existencia. Los mensajes de error pueden divulgar información sensible que se puede usar para escalar ataques.

Factor de riesgo: Medio

Puntuación CVSSv3.0: 5.3

Cadena de vectores: CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:L/I:N/A:N

Solución: Es necesario verificar que la página esté revelando mensajes detallados de error o advertencia y de ser así, se recomienda configurar correctamente la aplicación para que registre los errores en un archivo del sistema, en lugar de mostrar el error al usuario.

4.6.3.28. Páginas de pruebas encontradas en sitios de producción

Se encontró, en algunos de los sitios web, páginas de prueba activas en producción, esta práctica no es recomendable para un sitio productivo debido a que puede llevar a que se cometan errores por parte de los desarrolladores, o bien que se exponga información confidencial que no debería ser expuesta al público, como por ejemplo formularios de prueba que aún no están listos para ser utilizados por los usuarios.

Factor de riesgo: Medio

Puntuación CVSSv3.0: 5.3

Cadena de vectores: CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:L/I:N/A:N

Solución: Remover las páginas de prueba o sitios de prueba de la página productiva, se recomienda también implementar una validación a la hora de incluir este tipo de cambios en ambientes productivos.

4.6.3.29. Divulgación de la versión ASP.NET

Las respuestas de HTTP que son devueltas por esta aplicación web incluyen un encabezado llamado X-AspNet-Version. El valor de este encabezado lo utiliza Visual Studio para determinar qué versión de ASP.NET está siendo usada. Para los sitios de producción no es necesario, y lo ideal es que sea desactivado.

Factor de riesgo: Medio

Puntuación CVSSv3.0: 5.3

Cadena de vectores: CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:L/I:N/A:N

Solución: Se debe aplicar los siguientes cambios al archivo web.config para evitar la divulgación de la versión ASP.NET:

```
<System.Web>
 <httpRuntime enableVersionHeader="false" />
</System.Web>
```

4.6.3.30. Exposición de XAMPP

Se ha encontrado referencias a sitios externos que brindan información sensible, como el servidor de software libre XAMPP y la ruta donde se encuentra instalado. Por ello, los mensajes desplegados al cargar la página proveen información de la tecnología utilizada en el servidor web, así como la versión del servidor. Un atacante podría utilizar está información para intentar realizar ataques con vulnerabilidades específicas para la versión de XAMPP utilizada, así como detectar páginas de configuración de XAMPP que puedan estar públicas.

Factor de riesgo: Medio

Puntuación CVSSv3.0: 5.3

Cadena de vectores: CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:L/I:N/A:N

Solución: Es recomendable asegurarse de que si se implementan aplicaciones que consumen servicios de otras páginas o servidores, que sean revisados antes de implementarse para no revelar información que pueda ser considerada sensible. Se recomienda remover la información desplegada.

4.6.3.31. Ataque remoto de DDoS por la versión del sistema operativo

Se ha encontrado una vulnerabilidad de denegación de servicio en la forma en que el servidor Apache HTTP maneja los múltiples rangos de superposición:

http://seclists.org/fulldisclosure/2011/Aug/175

Se puede realizar de forma remota un ataque mediante una herramienta que se encuentra en internet y podría causar un consumo elevado de CPU del servidor, y provocar así una pérdida del servicio y la inhabilitación de la página web.

Las versiones de Apache afectadas son 1.3.x, 2.0.x hasta 2.0.64, y 2.2.x hasta 2.2.19.

Factor de riesgo: Media

Puntuación CVSSv3.0: 5.3

Cadena de vectores: CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:N/I:N/A:L

Solución: Es necesario actualizar a la última versión del servidor HTTP de Apache (2.2.20 o posterior).

4.6.3.32. El análisis en algunos hilos particulares de PHP permite DDoS

Se ha determinado que cuando PHP analiza algunas cadenas como un número de punto flotante '2.2250738585072011e-308' se congela o se detiene, lo que causa una denegación de servicio. Las versiones de PHP que se han visto afectadas son la 5.3 hasta la versión 5.3.5 y la 5.2 hasta la versión 5.2.17

Factor de riesgo: Medio

Puntuación CVSSv3.0: 5.3

Cadena de vectores: CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:N/I:N/A:L

Solución: Es necesario actualizar a la versión de PHP más reciente disponible.

4.6.3.33. Archivo DS_Store en el servidor web

Los archivos de DS_Store son creados automáticamente por el Finder de Mac OS X en los directorios que se exploraron. Estos archivos incluyen en algunas ocasiones información sobre la configuración del sistema. Al ser un servidor web, los archivos podrán ser descargados por los usuarios de la página, por lo que la información expuesta estará en manos de todo usuario de la página web. Si desea más información, se puede consultar las actualizaciones de seguridad de Apple.

Factor de riesgo: Medio

Puntuación CVSSv3.0: 5.3

Cadena de vectores: CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:L/I:N/A:N

Solución: Remover estos archivos del servidor web.

4.6.3.34. Posible exposición de directorios sensibles

Se ha encontrado un posible directorio sensible. Este directorio no está directamente vinculado desde el sitio web. Lo que busca esta comprobación son recursos sensibles de uso común, como directorios de respaldo, volcados de bases de datos, páginas de administración, directorios temporales. Cada uno de estos directorios podría ayudar a un atacante a obtener más información sobre su objetivo.

Factor de riesgo: Medio

Puntuación CVSSv3.0: 5.3

Cadena de vectores:CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:L/I:N/A:N

Solución: Se debe restringir el acceso a esos directorios o quitarlos por completo del sitio web.

4.6.3.35. Exposición de información sensible

A través de la página web es posible descargar una lista de los empleados con su respectivo nombre completo, departamento, teléfono, correo electrónico, puesto donde trabajan y extensión. Esta información puede ser utilizada por un atacante para suplantar la identidad de un funcionario con el fin de enviar una campaña de Spear phishing al personal, buscando infectar a la mayor cantidad posible.

Factor de riesgo: Medio

Puntuación CVSSv3.0: 5.3

Cadena de vectores: CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:L/I:N/A:N

Solución: Analizar si es necesario exponer este tipo de información, en caso de confirmarse que no es necesario se requiere remover la información.

4.6.3.36. SSL Medium Strength Cipher Suites Supported

El sitio web admite el uso de cifrado SSL que ofrece un cifrado de intensidad media. La intensidad media es considerada como cualquier encriptación que utiliza longitudes de clave de al menos 64 bits y menos de 112 bits, o bien que utiliza el conjunto de cifrado 3DES. Es más fácil eludir el cifrado de intensidad media si el atacante se encuentra en la misma red física.

Factor de riesgo: Medio

Puntuación CVSSv3.0: 5.3

Cadena de vectores: CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:L/I:N/A:N

Solución: Vuelva a configurar la aplicación si es posible para evitar el uso de cifrados de intensidad media.

4.6.3.37. SSL 64-bit Block Size Cipher Suites Supported (SWEET32)

El sitio web admite el uso de un cifrado de bloques con bloques de 64 bits en uno o más conjuntos de cifrado. Por lo tanto, se ve afectado por una vulnerabilidad, conocida como SWEET32. Un atacante tipo Man-in-the-Middle con suficientes recursos puede explotar esta vulnerabilidad, a través de un ataque conocido como "birthday", para poder permitirle divulgar el texto secreto, como las cookies HTTPS seguras, y posiblemente resulta en la apropiación de una sesión autenticada. Las pruebas de concepto han demostrado que los atacantes pueden recuperar las cookies de autenticación de una sesión HTTPS en tan solo 30 horas. Es importante entender que el envío de una gran cantidad de solicitudes por medio de la misma conexión TLS entre cliente y servidor es un requisito para llevar a cabo este tipo de ataque. Por tanto, si el número de solicitudes permitidas para una conexión única fuera limitado, esto mitigaría la vulnerabilidad.

Factor de riesgo: Medio

Puntuación CVSSv3.0: 5.3

Cadena de vectores: CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:L/I:N/A:N

Solución: De ser posible, configurar la aplicación para evitar el uso de sistemas de cifrado de 64 bits.

4.6.3.38. Divulgación de información de transferencia de zona del servidor DNS (AXFR)

El servidor de DNS permite realizar transferencias de zona DNS. Una transferencia de zona permite que un atacante remoto llene instantáneamente una lista de posibles objetivos de ataque. Además, las empresas suelen utilizar una convención de nomenclatura que puede dar pistas sobre una aplicación principal de servidores (por ejemplo, proxy.example.com, nómina.example.com, b2b.example.com, etc.).

Como tal, esta información es de gran utilidad para un atacante, que puede usarla para obtener información sobre la topología de la red y detectar nuevos objetivos.

Factor de riesgo: Medio

Puntuación CVSSv3.0: 5.3

Cadena de vectores: CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:L/I:N/A:N

Solución: Se debe limitar las transferencias de zona DNS únicamente a los servidores que necesitan la información.

4.6.3.39. Internet Key Exchange (IKE) Aggressive Mode with Pre-Shared Key

El servicio de Internet Key Exchange (IKE) versión 1, tiene soporte para el modo agresivo con autenticación de clave previamente compartida (PSK). Tal configuración podría permitir a un atacante capturar y crackear el PSK de una puerta de enlace VPN y obtener acceso no autorizado a redes privadas.

Factor de riesgo: Medio

Puntuación CVSSv3.0: 5.3

Cadena de vectores: CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:L/I:N/A:N

Solución:

- Desactivar el modo agresivo si es compatible.
- No usar la clave previamente compartida para autenticación si es posible.
- Si no se puede evitar el uso de la clave previamente compartida, use combinaciones muy fuertes.
- Si es posible, no permita conexiones VPN desde ninguna dirección IP.

Se debe tomar en cuenta que este complemento no se ejecuta en IPv6.

4.6.3.40. No se detecta Web Application Firewall (WAF)

El principal objetivo de utilizar un WAF es tratar de proteger la aplicación web de ataques, como por ejemplo Cross Site Scripting (XSS), Inyección de código SQL, manipulación en los parámetros de cabeceras, que podrían dejar desprotegidos un sistema de detección de intrusos.

Los firewalls de aplicaciones web (WAF) proporcionan una solución efectiva para detectar amenazas, examinando las solicitudes HTTP entrantes, incluso antes de que lleguen al servidor, para detectar y bloquear los ataques maliciosos.

Se puede implementar como un dispositivo de hardware, como un complemento del servidor ejecutándose directamente en el servidor o en línea. El WAF intercepta todas las solicitudes y las analiza, como por ejemplo las solicitudes GET y POST, con el objetivo de identificar y filtrar el tráfico ilegítimo.

Figura 6. Diagrama básico de Web Application Firewall



Recuperado de: https://pentestlab.blog/tag/web-application-firewall/

Factor de riesgo: Media

Puntuación CVSSv3.0: 5.3

Cadena de vectores: CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:L/I:N/A:N

Solución: Se recomienda la implementación de una capa extra de seguridad para la capa de aplicación, la implementación de un WAF es recomendable para toda aplicación web, especialmente cuando la información que se protege es información a nivel de gobierno.

4.6.3.41. Caché del servidor DNS divulga información de forma remota

El servidor DNS responde a las consultas de dominios de terceros que no tienen establecido el bit de recursividad. Esto puede permitir que un atacante remoto determine qué dominios se han resuelto recientemente a través de este servidor de DNS y, por lo tanto, qué hosts se han visitado recientemente.

Por ejemplo, si un atacante está interesado en saber si su compañía utiliza los servicios en línea de una institución financiera en particular, podría usar este ataque para construir un modelo estadístico con respecto al uso de la compañía de esa institución financiera. Por supuesto, el ataque también se puede usar para encontrar socios B2B, patrones de navegación web, servidores de correo externo y más.

Factor de riesgo: Medio

Puntuación CVSSv3.0: 5.2

Cadena de vectores: CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:L/I:N/A:N

Solución: Se debe contactar con el proveedor del software DNS para obtener una solución.

4.6.3.42. Nivel de cifrado de Terminal Services medio o bajo

El servicio de Terminal Services remoto no está configurado para usar un cifrado fuerte. El uso de un cifrado débil con este servicio puede permitir que un atacante escuche las comunicaciones más fácilmente y obtenga capturas de pantalla y/o pulsaciones de teclas.

Factor de riesgo: Medio

Puntuación CVSSv3.0: 3.7

Cadena de vectores: CVSS:3.0/AV:N/AC:H/PR:N/UI:N/S:U/C:L/I:N/A:N

Solución: Cambiar el nivel de cifrado de los servicios de terminal a:

1- Nivel Alto

2- FIPS Compliant

Para cambiar a FIPS compliant es necesario ir al Editor de Políticas Locales (Local Group Policy Editor) y habilitar la política “System Cryptography : Use FIPS compliant algorithms for encryption”, que se encuentra en \Policies\Windows Settings\Security Settings\Local Policies\Security Options

4.6.3.43. Same site scripting

El hacker de sombrero blanco, Tavis Ormandy informó de la existencia de una configuración incorrecta de DNS que puede dar lugar a un pequeño problema de seguridad con las aplicaciones web. Es una práctica común y natural instalar registros del formulario "localhost. IN A 127.0.0.1" en configuraciones de servidores de nombres, pero extrañamente, los administradores suelen erróneamente olvidar el punto final, presentando una variación interesante de Cross-Site Scripting (XSS) llamado Same-Site Scripting. El punto que falta indica que el registro no es un dominio completamente calificado, y por lo tanto las consultas del formulario "localhost.example.com" se pueden resolver. Aunque superficialmente esto puede parecer inofensivo, de hecho, permite a un atacante engañar al RFC2109 (Mecanismo de Gestión del Estado HTTP) de las mismas restricciones de origen y, por lo tanto, secuestrar los datos de gestión del estado.

Factor de riesgo: Bajo

Puntuación CVSSv3.0: 3.7

Cadena de vectores: CVSS:3.0/AV:N/AC:H/PR:N/UI:N/S:U/C:N/I:L/A:N

Solución: Se recomienda que las entradas de localhost que son non-FQ se eliminen de las configuraciones del servidor de nombres para dominios que hospedan sitios web que dependen de HTTP para la gestión del estado.

4.6.3.44. Soporte de SSL RC4 Cipher Suites

El host remoto admite el uso de RC4 en uno o más conjuntos de cifrado. El cifrado RC4 tiene fallas en la generación de una secuencia pseudoaleatoria de bytes, de modo que se introduce una amplia variedad de pequeños sesgos en la secuencia, disminuyendo su aleatoriedad.

Si el texto sin cifrar se encripta repetidamente (por ejemplo, cookies HTTP) y un atacante puede obtener muchos (por ejemplo, decenas de millones) de textos cifrados, el atacante puede derivar el texto sin cifrar.

Factor de riesgo: Bajo

Puntuación CVSSv3.0: 3.7

Cadena de vectores: CVSS:3.0/AV:N/AC:H/PR:N/UI:N/S:U/C:L/I:N/A:N

Solución: De ser posible, se debe configurar la aplicación, para evitar el uso de cifrado RC4. Sería importante considerar el uso de TLS 1.2 con suites AES-GCM sujetas a soporte de navegador y servidor web.

4.6.3.45. Inicios de sesión de POP3 sin cifrar permitidos

El servidor remoto ejecuta un Daemon POP3 que permite inicios de sesión en texto sin cifrar sobre conexiones no encriptadas. Un atacante puede descubrir nombres de usuario y contraseñas haciendo sniffing de tráfico al Daemon POP3 si se usa un mecanismo de autenticación menos seguro (por ejemplo, el comando USER, AUTH PLAIN, AUTH LOGIN).

Factor de riesgo: Bajo

Puntuación CVSSv3.0: 3.7

Cadena de vectores: CVSS:3.0/AV:N/AC:H/PR:N/UI:N/S:U/C:L/I:N/A:N

Solución: Póngase en contacto con su proveedor para obtener una solución que pueda encriptar el tráfico con SSL/TLS utilizando stunnel.

4.6.3.46. Carga de archivos sin validación

El sitio permite a los visitantes subir archivos al servidor sin ser analizados previamente. Varias aplicaciones web permiten a los usuarios cargar archivos (como fotos, imágenes, sonidos, entre otros). Los archivos que son cargados pueden representar un riesgo significativo si no se manejan correctamente o si no se validan en busca de malware o extensiones que no deberían ser aceptadas. Por ejemplo, si lo que se busca es que el usuario cargue una foto a la página, no debería aceptar extensiones de archivos ejecutables, porque estos archivos podrían ser archivos infectados, los cuales se alojarían en el servidor web y podrían ser ejecutados por algún usuario o podrían contener código que ejecute alguna acción, y causar una afectación al equipo.

Factor de riesgo: Informativo

Puntuación CVSSv3.0: 0.0

Cadena de vectores: CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:N/I:N/A:N

Solución: Se deben restringir los tipos de archivos que son aceptados para subirse al servidor, se debe verificar la extensión de los archivos y permitir solamente la carga de algunos tipos de archivos. Se debe hacer un enfoque de carga de archivos por medio de listas blancas para reducir la posibilidad de que se suban archivos no esperados por la aplicación. Comprobar que no existen tipos de extensiones duplicadas como .php.png. Comprobar si existen archivos sin un nombre de archivo como .htaccess (en ASP.NET, compruebe si hay archivos de configuración como web.config). Cambiar los permisos en el directorio donde se guarda el archivo una vez subido al servidor, para que los archivos que contiene el directorio no puedan ser ejecutados. Si es posible, se debe cambiar el nombre de los archivos que se cargan.

4.6.3.47. Referencias a un sitio web en lista negra

Se ha encontrado dentro del sitio web una referencia hacia un sitio que aparece en una lista negra de la página Virus Total www.virustotal.com.

Factor de riesgo: Informativo

Puntuación CVSS: 0

Cadena de vectores: CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:N/I:N/A:N

Solución: Para este tipo de eventos lo recomendable es que se contacte a la persona encargada del sitio para que, en caso de ser un falso positivo, proceda a contactar a los administradores de la lista donde se encuentra para que lo remuevan. En caso de no ser falso positivo, proceder a arreglar el sitio removiendo el posible malware reportado.

4.6.3.48. Versión de Microsoft IIS expuesta

Las respuestas HTTP devueltas por esta aplicación web incluyen un encabezado denominado servidor. El valor de este encabezado incluye la versión del servidor Microsoft IIS. El encabezado HTTP puede revelar información confidencial. Esta información puede ser utilizada para lanzar más ataques a futuro.

Factor de riesgo: Informativo

Puntuación CVSSv3.0: 0.0

Cadena de vectores: CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:N/I:N/A:N

Solución: Microsoft IIS debe configurarse para eliminar encabezados de respuesta HTTP no deseados en la respuesta al cliente.

4.6.3.49. Envío de contraseñas en texto plano al correo electrónico

Para los nuevos usuarios al registrarse, el sistema envía la contraseña de acceso en texto plano al correo del usuario. El usuario ingresa con esta contraseña y el sistema no le solicita cambiarla. Esta práctica supone un riesgo de seguridad, porque por más segura que se establezca una contraseña, siguiendo las mejores prácticas, si un servicio las almacena tal cual, y las envía por email en texto plano, están al alcance de cualquier ciberdelincuente, al igual que si el correo electrónico es vulnerado, tendrán acceso a la contraseña para ingresar a este segundo sistema.

Factor de riesgo: Informativo

Puntuación CVSSv3.0: 0.0

Cadena de vectores: CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:N/I:N/A:N

Solución: Implementar un sistema de envío y confirmación de cuentas en la que no se envíe en texto plano la contraseña al mail de los usuarios.

4.6.3.50. Servidor FTP Detectado

Es posible obtener el mensaje de bienvenida que entrega el servidor FTP, al conectarse de forma remota al puerto donde este servicio está configurado. Los servidores FTP son susceptibles a ataques de fuerza bruta para tratar de adivinar las credenciales de acceso.

Factor de riesgo: Informativo

Puntuación CVSSv3.0: 0.0

Cadena de vectores: CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:N/I:N/A:N

Solución: Se recomienda verificar si este servidor FTP debería estar configurado para que pueda ser detectado de forma externa y remota. En caso de ser necesario utilizar un sistema de este tipo, se recomienda utilizar SFTP y revisar la información que se almacena.

4.6.3.51. VERITAS NetBackup Agent Detection

Se ha encontrado que el sitio web ejecuta el servicio VERITAS NetBackup Java Console. Este servicio es utilizado por la interfaz gráfica de usuario de la Consola Java de NetBackup para administrar el servidor de respaldo. Un usuario autorizado para conectarse a este servicio, puede usarlo como un shell remoto con privilegios del sistema enviando mensajes 'command_EXEC_LIST'.

Factor de riesgo: Informativo.

Puntuación CVSSv3.0: 0.0

Cadena de vectores: CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:N/I:N/A:N

Solución: Como parte de las mejores prácticas, Veritas recomienda restringir el acceso remoto, de ser requerido que sea solo a usuarios autorizados y mantener los sistemas operativos y las aplicaciones actualizadas con los últimos parches de los vendedores.

4.6.3.52. Servidor SMTP Detectado

El sitio web está ejecutando un servidor de correo (SMTP) en el puerto 25. Dado que los servidores SMTP son el objetivo de los spammers, se recomienda que los desactive si no los usa.

Factor de riesgo: Informativo

Puntuación CVSSv3.0: 0.0

Cadena de vectores: CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:N/I:N/A:N

Solución: Se debe deshabilitar este servicio si no se usa, o filtrar el tráfico entrante a este puerto.

4.6.3.53. Microsoft SQL Server TCP/IP Listener Detection

El servidor remoto ejecuta MSSQL, un servidor de base de datos de Microsoft. Es posible extraer el número de versión del servidor remoto, obtenido de la respuesta que se envía preinicio de sesión del servidor. Esta información puede ser utilizada en ataques futuros.

Factor de riesgo: Informativo

Puntuación CVSSv3.0: 0.0

Cadena de vectores: CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:N/I:N/A:N

Solución: Se debe limitar el acceso a la base de datos a direcciones IP permitidas únicamente.

4.6.3.54. Microsoft SQL Server UDP Query Remote Version Disclosure

El servidor Microsoft SQL tiene una función en la que los usuarios remotos pueden consultar la versión que se está ejecutando. La consulta tiene lugar en el mismo puerto UDP que maneja la asignación de múltiples instancias de SQL Server en la misma máquina. Es importante observar que, después de la Versión 8.00.194, Microsoft decidió no actualizar esta función. Esto significa

que los datos devueltos por el ping de SQL son inexactos para las versiones más recientes de SQL Server.

Factor de riesgo: Informativo

Puntuación CVSSv3.0: 0.0

Cadena de vectores: CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:N/I:N/A:N

Solución: Si solo hay una única instancia de SQL instalada en el host remoto, considere filtrar el tráfico entrante a este puerto.

4.6.3.55. Detección de servidor PostgreSQL

El servicio remoto es un servidor de base de datos PostgreSQL, o un derivado como EnterpriseDB. Este tipo de información puede ser utilizada por un atacante para lanzar ataques a futuro cuando se liberen vulnerabilidades específicas para este servicio.

Factor de riesgo: Informativo

Puntuación CVSSv3.0: 0.0

Cadena de vectores: CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:N/I:N/A:N

Solución: Se recomienda limitar el tráfico entrante a este puerto únicamente.

4.6.3.56. Detección del servidor POP

En el host remoto se ejecuta un servidor que comprende el protocolo Post Office (POP), que utilizan los clientes de correo electrónico para recuperar/extraer mensajes de un servidor, posiblemente a través de un enlace de red. Dependiendo de la configuración que posea este servidor, puede ser inseguro para los usuarios.

Factor de riesgo: Informativo

Puntuación CVSSv3.0: 0.0

Cadena de vectores: CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:N/I:N/A:N

Solución: Deshabilitar el servicio si no está siendo utilizado. En caso de estar siendo utilizado configurar SSL/TLS y autenticación.

4.6.3.57. MySQL Server Detection

En el sitio web ejecuta MySQL, un servidor de base de datos de código abierto. Dependiendo de la configuración y prácticas de seguridad aplicadas en este servidor, podría ser vulnerado.

Factor de riesgo: Informativo

Puntuación CVSSv3.0: 0.0

Cadena de vectores: CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:N/I:N/A:N

Solución: Deshabilitar el servicio si no está siendo utilizado. En caso de estar siendo utilizado, configurar listas de acceso para solamente las direcciones IP que necesitan accederlo.

4.6.3.58. Archivos predeterminados configurados en el sistema

La página web posee los archivos de configuración de Drupal por defecto visibles, archivos como install.php, CHANGELOG.txt,INSTALL.txt , INSTALL.mysql.txt e INSTALL.pgsql.txt, revelan información que se puede considerar como sensible ya que provee información de los componentes y versiones del sistema existentes.

Factor de riesgo: Informativo

Puntuación CVSSv3.0: 0.0

Cadena de vectores: CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:N/I:N/A:N

Solución: Algunas soluciones como Drupal permiten cambiar los permisos de los archivos, mover los archivos o eliminarlos para que no sean visibles.

4.6.4 Vulnerabilidades detectadas en todos los sitios web analizados

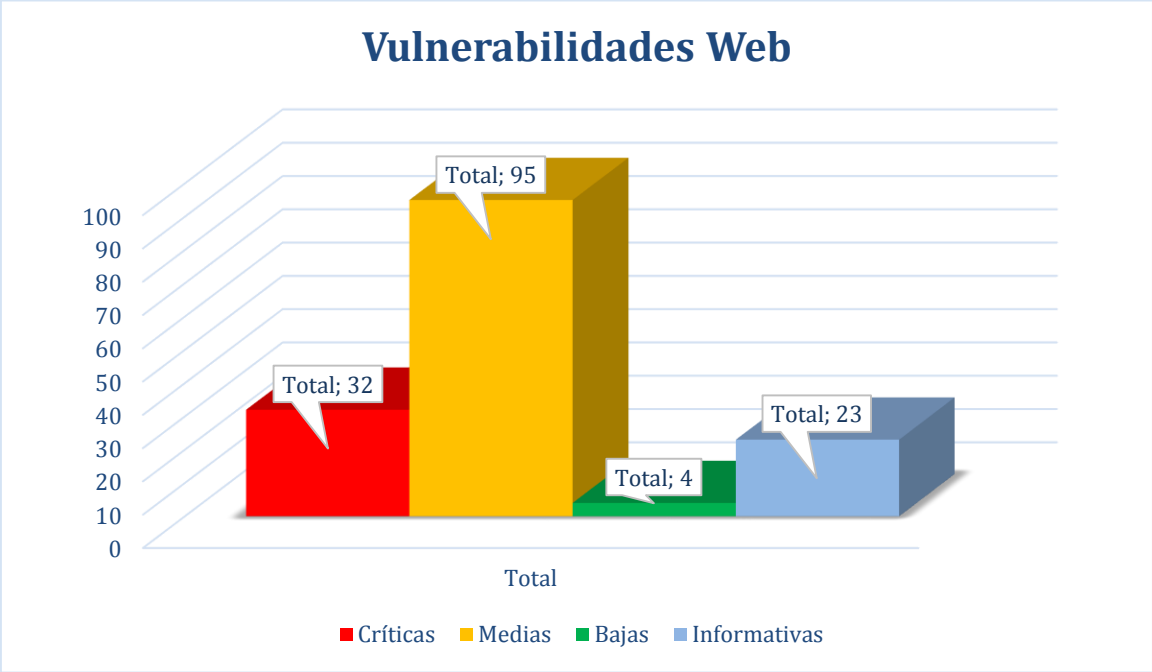
Después de realizadas las pruebas en cada uno de los 16 sitios web, se determina que poseen un total de 154 vulnerabilidades, entre las cuales se enlistan desde las críticas hasta las informativas. El siguiente resumen muestra el detalle por nivel de riesgo.

Resumen ejecutivo

Figura 7. Total de vulnerabilidades detectadas por nivel de riesgo en todos los sitios web

Nivel de riesgo	CVSS	Total
Crítico	7.0-10	32
Medio	4.0-6.9	95
Bajo	0.1-3.9	4
Informativo	0.0	23
TOTAL GENERAL		**154**

Figura 8 Total de vulnerabilidades detectadas por nivel de riesgo en todos los sitios web



En la **Figura 9** se muestra, en detalle, la totalidad de vulnerabilidades que se han detectado, enlistadas por nivel de riesgo y por cada ministerio analizado.

Figura 9. Vulnerabilidades por categoría en cada Ministerio

Ministerio	Crítico	Medio	Bajo	Info
Instituto Mixto de Ayuda Social (IMAS)	3	5	0	3
Instituto Costarricense de Turismo	1	9	1	5
Ministerio de Trabajo y Seguridad Social	2	9	0	3
Ministerio de Ambiente y Energía	2	6	0	2
Ministerio de Hacienda	3	7	1	0
Ministerio de Ciencia, Tecnología y	1	4	0	0

Telecomunicaciones				
Ministerio de Seguridad Pública	**5**	**5**	**0**	**1**
Ministerio de Gobernación y Policía	**6**	**4**	**0**	**3**
Ministerio de Justicia y Paz	**2**	**5**	**0**	**1**
Ministerio de Cultura y Juventud	**1**	**4**	**0**	**1**
Ministerio de Comercio Exterior	**1**	**9**	**1**	**1**
Instituto Nacional de las mujeres (INAMU)	**1**	**9**	**0**	**1**
Ministerio de Vivienda y Asentamientos Humanos (MIVAH)	**1**	**2**	**1**	**0**
MIDEPLAN	**1**	**5**	**0**	**0**
Presidencia de la República	**2**	**6**	**0**	**1**
Ministerio de Educación Pública	**0**	**6**	**0**	**1**
TOTAL	**32**	**95**	**4**	**23**

4.7 Análisis de la situación al momento de realizar las pruebas por cada Ministerio

Los elementos que han sido considerados importantes de señalar, ya sea por el riesgo que representan o bien por lo que se considera no es una buena práctica de seguridad, serán detallados en esta sección. Se han considerado fundamentalmente las vulnerabilidades críticas y medias encontradas, en algunos casos se han incluido vulnerabilidades de severidad baja o informativa,

que potencialmente pueden convertirse en medias o altas, para que sean consideras como una oportunidad de mejora y se puedan solucionar de manera inmediata.

4.7.1 Instituto Mixto de Ayuda Social (IMAS)

Nivel de riesgo	CVSS	Total Vulnerabilidades
Crítico	7.0-10	3
Medio	4.0-6.9	5
Bajo	0.1-3.9	0
Informativo	0.0	3

ID	VULNERABILIDAD	CVSS	NIVEL
4.6.3.1	Inyección de código SQL	**10.0**	Crítico
4.6.3.8	Aplicación web desactualizada	**9.8**	Crítico
4.6.3.12	Métodos inseguros habilitados	**7.5**	Crítico
4.6.3.15	Uso de HTTP para autenticación	**6.5**	Medio
4.6.3.24	HTTPS no configurado	**5.3**	Medio
4.6.3.26	Gestión de archivos	**5.3**	Medio
4.6.3.35	Exposición de información sensible	**5.3**	Medio
4.6.3.40	No se detecta Web Application Firewall (WAF)	**5.3**	Medio

4.6.3.53	Microsoft SQL Server TCP/IP Listener Detection	-	Informativo
4.6.3.54	Microsoft SQL Server UDP Query Remote Version Disclosure	-	Informativo
4.6.3.55	Detección de servidor PostgreSQL	-	Informativo

Resumen de las vulnerabilidades encontradas

Vulnerabilidad descrita en la sección 4.6.3.1. Inyección de código SQL esta es una vulnerabilidad que permite a un atacante ejecutar código SQL en la base de datos al manipular el código ingresado en un campo de texto. El formulario de acceso es vulnerable, y se comprueba la vulnerabilidad ingresando en el campo de usuario y contraseña:

Usuario**: admin'--**

Contraseña**: admin'--**

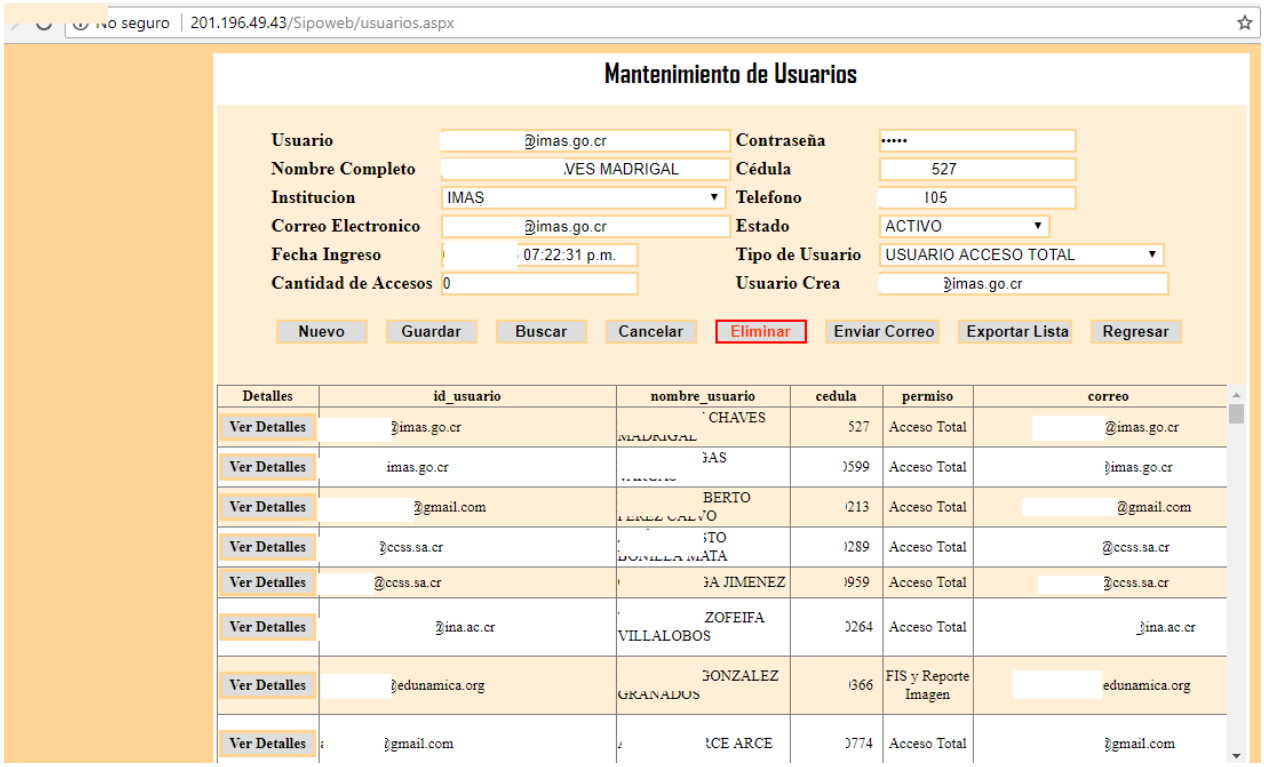
- http://201.196.49.43/Sipoweb/login.aspx

Figura 10. Posibilidad de ingreso al sistema con SQL injection en IMAS



Debido a lo anterior es posible ingresar al mantenimiento de usuarios, con la posibilidad de modificar usuarios y contraseñas, elevar o quitar privilegios.

Figura 11. Revelación de información sensible en IMAS



Refiérase a la sección 4.6.3.8. La versión de Drupal está desactualizada, se ha detectado que se encuentra en una versión inferior a 8.1.3/7.44

http://www.imas.go.cr//CHANGELOG.txt

Figura 12. Versión de drupal

```
Drupal 7.41, 2015-10-21
-----------------------
- Fixed security issues (open redirect). See SA-CORE-2015-004.

Drupal 7.40, 2015-10-14
-----------------------
```

Refiérase a la sección 4.6.3.12 para mayor información del uso de métodos inseguros habilitados, esto porque el método **TRACE** se encuentra habilitado, a continuación se muestran algunos vínculos de referencia.

Insecure methods found (trace) on http://www.imas.go.cr/

Insecure methods found (trace) on http://www.imas.go.cr/sites/

Insecure methods found (trace) on http://www.imas.go.cr/sites/default/

Insecure methods found (trace) on http://www.imas.go.cr/sites/default/files/

Refiérase a la sección 4.6.3.15 para mayor información del uso de HTTP para autenticación.

Figura 13. Ingreso al sitio en HTTP IMAS





Refiérase a la sección 4.6.3.24 para mayores detalles del uso de certificados digitales para sitios web.

Refiérase a la sección 4.6.3.26 para mayor información relacionada a la gestión de archivos en sitios web.

Figura 14. Lista de archivos expuesto del IMAS

Id	Type	URL
406	pdf	http://www.imas.go.cr/sites/default/files/docs/acta-053-09_27-julio_0.pdf
407	pdf	http://www.imas.go.cr/sites/default/files/docs/acta-001e-08_25-setiembre.pdf
408	pdf	http://www.imas.go.cr/sites/default/files/docs/acta-001-08_10-enero.pdf
409	pdf	http://www.imas.go.cr/sites/default/files/docs/acta-069-10-14_23-octubre_0.pdf
410	pdf	http://www.imas.go.cr/sites/default/files/docs/acta-085-11-2015_30-noviembre_0.pdf
411	pdf	http://www.imas.go.cr/sites/default/files/docs/aud053-_2016.pdf
412	pdf	http://www.imas.go.cr/sites/default/files/docs/acta-064-09-2015-_07-setiembre_0.pdf
413	pdf	http://www.imas.go.cr/sites/default/files/files/106%20-%20Anexo%206%20Informe%20de%20Gestion%20en%20la%20Planificacion.pdf
414	pdf	http://www.imas.go.cr/sites/default/files/docs/acta-53-08-2017_24-agosto.pdf
415	pdf	http://www.imas.go.cr/sites/default/files/docs/acta-70-12-2016_08-diciembre.pdf
416	pdf	http://www.imas.go.cr/sites/default/files/docs/acta-045-07_11-junio.pdf
417	pdf	http://www.imas.go.cr/sites/default/files/docs/acta-064-10-14_06-octubre_0.pdf
418	pdf	http://www.imas.go.cr/sites/default/files/docs/acta-054-07_09-julio.pdf
419	pdf	http://www.imas.go.cr/sites/default/files/docs/acta-40-06-2017_29-junio.pdf
420	pdf	http://www.imas.go.cr/sites/default/files/docs/acta-54-08-2017_28-agosto.pdf

Vulnerabilidad descrita en la sección 4.6.3.35. A través de la página web y a raíz de la vulnerabilidad de SQL injection es posible descargar las listas de usuarios, para este caso se pudo obtener la información de 991 usuarios.

Figura 15. Lista parcial de usuarios del IMAS



	B	C	D	E	F	G	H	I	J
975	YOVANA ANDREA MARTINEZ SANCHEZ	701580037	Acceso Total	ymartisa@ccss.sa.cr	85580584	CCSS, AREA DE SALUD LIMON	18/10/2012	Activo	88
976	YAIROL MANRIQUE MATA PICADO	207220566	Acceso Total	ymata@imas.go.cr		IMAS	16/05/2017	Activo	45
977	YASBEK CASTILLO CAMPOS	1-1503-0530	Acceso Total	ymcasti@ccss.sa.cr		CCSS	30/08/2017	Activo	172
978	YANCY MORA	111340042	FIS y Reporte Imagen	ymora@dhr.go.cr	27832638	IMAS	5/6/2012	Activo	206
979	YARIELA VINDAS MEJICANO	601071468	Acceso Total	ymvindas@ccss.sa.cr	27718500	CCSS	18/07/2017	Activo	137
980	YENDRY NATALIA ARIAS ULLOA	304740364	Acceso Total	ynariasu@ccss.sa.cr	22247293	CCSS	25/10/2017	Activo	26
981	YESSICA NUÑEZ ALFARO	206860300	Acceso Total	ynuneza@imas.go.cr	24945337	IMAS	17/05/2017	Activo	10
982	YENDRY PAOLA CALVO VENEGAS	603930384	Acceso Total	ypcalvov@ccss.sa.cr	27751429	CCSS	3/11/2015	Activo	161
983	YURY PATRICIA GUTIÉRREZ MORA	603900264	Acceso Total	ypgutier@ccss.sa.cr		CCSS	8/9/2017	Activo	0
984	YELDA PIÑAS CHAVES	107790863	Acceso Total	ypinarchaves@ina.ac.cr	22106934	INSTITUTO NACIONAL DE APRENDIZAJE (INA)	20/12/2016	Activo	0
985	YIRLENE MONTOYA VALVERDE	113450337	Acceso Total	ypmontoya@ccss.sa.cr		CCSS	28/05/2014	Activo	748
986	YESENIA GERARDA QUESADA ARAYA	110120365	Acceso Total	yquesada@imas.go.cr	22193670	IMAS	11/7/2017	Activo	0
987	YESSENIA UGALDE ARAYA	110830189	Acceso Total	yugalde@imas.go.cr	24601105	IMAS	9/5/2016	Activo	0
988	YOISER VARGAS VIDAURRE	603150918	Acceso Total	YVargasV@imas.go.cr		IMAS	14/12/2016	Activo	1
989	YULIANA MARÍA ZAMORA VILLALOBOS	113890777	Acceso Total	yzamora@imas.go.cr	22569583	IMAS	7/6/2017	Activo	2
990	ZAYLIN BONILLA MORALES	503430998	Acceso Total	zaylin.gs@gmail.com	84919024	MUNICIPALIDAD DE LA CRUZ	24/03/2015	Activo	72
991	ZAIDA MONTIEL		FIS y Reporte Imagen	zmontie@ccss.sa.cr		CCSS	20/06/2012	Activo	5

Refiérase a la sección 4.6.3.40 para obtener información acerca de lo que es un WAF y la importancia de implementarlo.

Refiérase a la sección 4.6.3.53 para información de MSSQL.

Refiérase a la sección 4.6.3.54 para información de la exposición de SQL.

Refiérase a la sección 4.6.3.55 para información de PostgreSQL.

4.7.2 Instituto Costarricense de Turismo (ICT)

Nivel de riesgo	CVSS	Total Vulnerabilidades
Crítico	7.0-10	1
Medio	4.0-6.9	9
Bajo	0.1-3.9	1
Informativo	0.0	5

ID	VULNERABILIDAD	CVSS	NIVEL

4.6.3.12	Métodos inseguros habilitados	**7.5**	Crítico
4.6.3.15	Uso de HTTP para autenticación	**6.5**	Medio
4.6.3.21	DNS Server Spoofed Request Amplification DDoS	**5.8**	Medio
4.6.3.24	HTTPS no configurado	**5.3**	Medio
4.6.3.26	Gestión de archivos	**5.3**	Medio
4.6.3.27	Mensaje de error en aplicación	**5.3**	Medio
4.6.3.28	Páginas de pruebas encontradas en sitios de producción	**5.3**	Medio
4.6.3.36	SSL Medium Strength Cipher Suites Supported	**5.3**	Medio
4.6.3.40	No se detecta Web Application Firewall (WAF)	**5.3**	Medio
4.6.3.41	Caché del servidor DNS divulga información de forma remota	**5.2**	Medio
4.6.3.45	Inicios de sesión de POP3 sin cifrar permitidos	**3.7**	Bajo
4.6.3.46	Carga de archivos sin validación	-	Informativo
4.6.3.50	Servidor FTP Detectado	-	Informativo
4.6.3.52	Servidor SMTP Detectado	-	Informativo
4.6.3.56	Detección del servidor POP	-	Informativo

4.6.3.57	MySQL Server Detection	-	Informativo

Resumen de las vulnerabilidades encontradas

Vulnerabilidad descrita en la sección 4.6.3.12 se encuentra el método **TRACE,** cuya función principal es la de diagnosticar fallas o bien para verificar si existen conexiones intermedias, puede ser utilizado para montar un ataque conocido como "Cross Site Tracing XST".

El método **PUT** habilitado permite al cliente subir archivos al servidor web, por lo tanto, un atacante podría explotar este método subiendo archivos maliciosos. Por otro lado, también tiene el método **DELETE** habilitado, lo cual permite borrar archivos de un servidor web que puede comprometer su funcionalidad. Se puede ver a continuación un ejemplo de dónde se detectó los métodos habilitados.

Figura 16. Direcciones encontradas de métodos inseguros

Folder	Methods
http://www.ict.go.cr/en/documents/	PUT
http://www.ict.go.cr/en/documents/	DELETE

Los métodos inseguros han sido encontrados en los siguientes URLs.

PUT

- http://www.ict.go.cr/es/documentos-institucionales/
- http://www.ict.go.cr/es/documentos-institucionales/gerencia/ley-9398/
- http://www.ict.go.cr/es/documentos-institucionales/auditoría

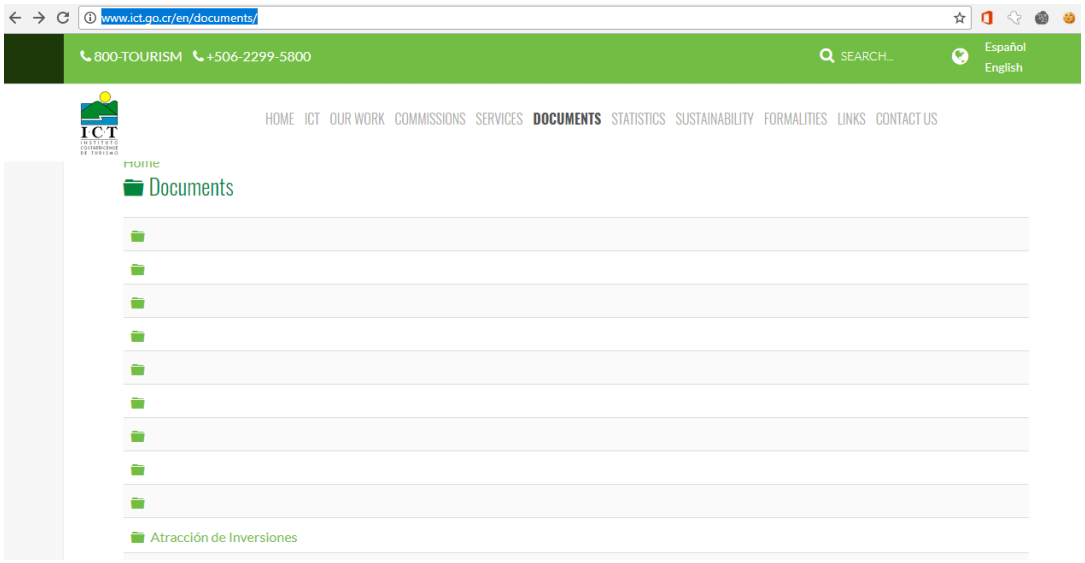
DELETE

- http://www.ict.go.cr/es/documentos-institucionales/
- http://www.ict.go.cr/es/documentos-institucionales/gerencia/ley-9398/
- http://www.ict.go.cr/es/documentos-institucionales/auditoría

También se encuentran en las subcarpetas asociadas, algunos ejemplos se muestran a continuación.

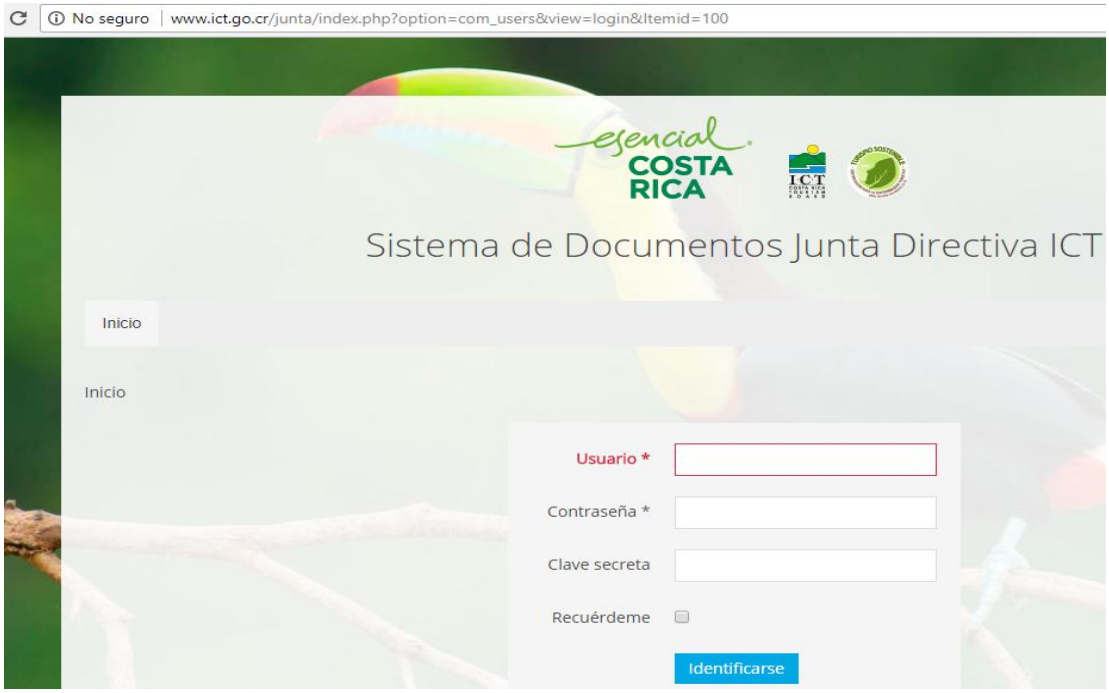
Prueba de creación de carpetas, aprovechándose del método PUT.

Figura 17. Creación de carpetas en el sitio del ICT



Vulnerabilidad descrita en la sección 4.6.3.15. No se recomienda el uso de HTTP en sitios donde se busca ingresar, por medio de un usuario y contraseña, a datos privados o descarga de archivos, ya que podría ser fácilmente comprometida.

Figura 18. Formulario de acceso con HTTP en ICT

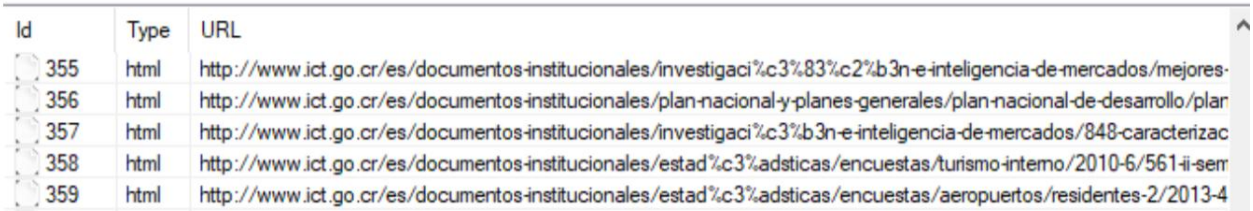


Vulnerabilidad descrita en la sección 4.6.3.21. Con la solicitud de amplificación DNS un atacante remoto puede lanzar un ataque de denegación de servicio contra un servidor.

Refiérase a la sección 4.6.3.24 para mayores detalles del uso de certificados digitales para sitios web.

Refiérase a la sección 4.6.3.26 para mayor información relacionada a la gestión de archivos en sitios web.

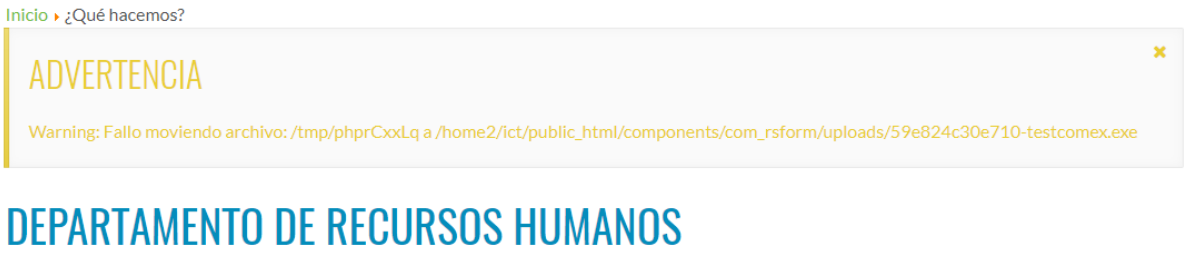
Figura 19. Archivos expuestos del ICT



Id	Type	URL
355	html	http://www.ict.go.cr/es/documentos-institucionales/investigaci%c3%83%c2%b3n-e-inteligencia-de-mercados/mejores-
356	html	http://www.ict.go.cr/es/documentos-institucionales/plan-nacional-y-planes-generales/plan-nacional-de-desarrollo/plar
357	html	http://www.ict.go.cr/es/documentos-institucionales/investigaci%c3%b3n-e-inteligencia-de-mercados/848-caracterizac
358	html	http://www.ict.go.cr/es/documentos-institucionales/estad%c3%adsticas/encuestas/turismo-interno/2010-6/561-ii-sem
359	html	http://www.ict.go.cr/es/documentos-institucionales/estad%c3%adsticas/encuestas/aeropuertos/residentes-2/2013-4

Refiérase a la sección 4.6.3.27 para conocer de los mensajes de error o advertencia de la aplicación.

Figura 20. Mensaje de error en ICT

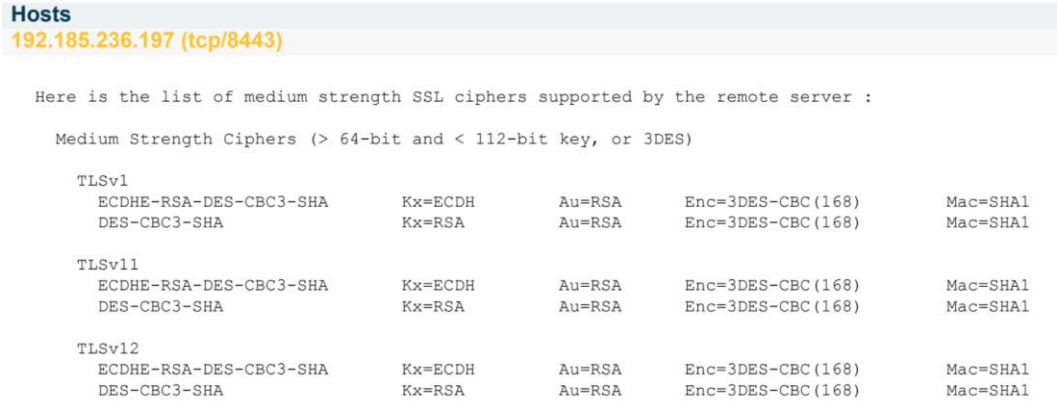


Refiérase a la sección 4.6.3.28 para conocer sobre las páginas de prueba en sitios de producción. La siguiente dirección es utilizada como prueba en el sitio en producción.

http://www.ict.go.cr/es/component/rsform

Vulnerabilidad descrita en la sección 4.6.3.36. El sitio web admite el uso de cifrado SSL que ofrece un cifrado de intensidad media. Se utiliza longitudes de clave de al menos 64 bits y menos de 112 bits o cifrado como 3DES.

Figura 21. Cifrado medio en ICT

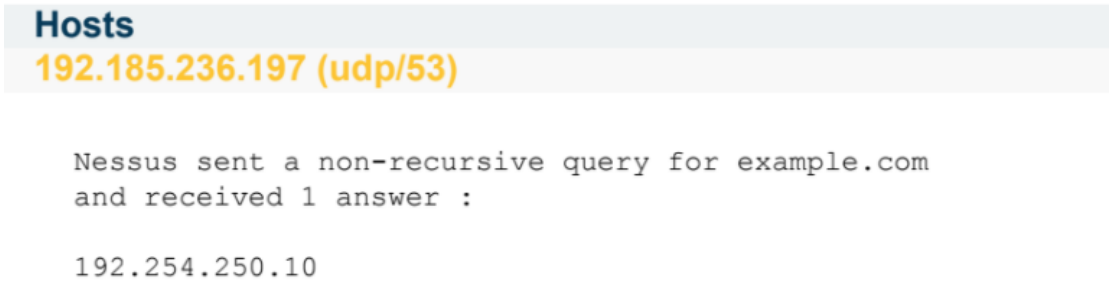


Refiérase a la sección 4.6.3.40 para obtener información acerca de lo que es un WAF y la importancia de implementarlo.

Vulnerabilidad descrita en la sección 4.6.3.41. El servidor DNS responde a las consultas de dominios de terceros que no tienen establecido el bit de recursividad. Esto puede permitir que un

atacante remoto determine qué dominios se han resuelto recientemente a través de este servidor de DNS y, por lo tanto, qué hosts se han visitado recientemente.

Figura 22. Respuesta de DNS



Vulnerabilidad descrita en la sección 4.6.3.45. El servidor remoto ejecuta un Daemon POP3 que permite inicios de sesión en texto sin cifrar sobre conexiones no encriptadas.

Figura 23. POP3 habilitado



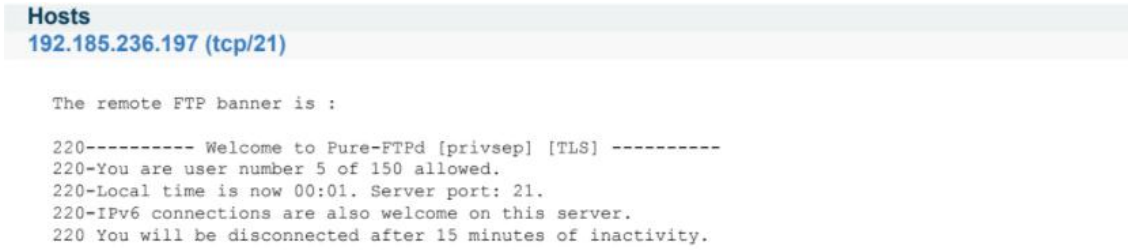
Refiérase a la sección 4.6.3.46 para obtener información relacionada a la carga de archivos. Algunos de los links de la página en donde se pueden subir archivos sin filtrar son:

http://www.ict.go.cr/es/informacion-institucional/recursos-humanos.html

http://www.ict.go.cr/es/2-uncategorised

Vulnerabilidad descrita en la sección 4.6.3.50. Es posible obtener el banner del servidor FTP remoto conectándose a un puerto remoto.

Figura 24. Anuncio de bienvenida del FTP



Vulnerabilidad descrita en la sección 4.6.3.52. El sitio web está ejecutando un servidor de correo (SMTP) en este puerto y puede ser objetivo para los “spammers”.

Vulnerabilidad descrita en la sección 4.6.3.56. En el host remoto se ejecuta un servidor que comprende el protocolo Post Office (POP).

Vulnerabilidad descrita en la sección 4.6.3.57. El sitio ejecuta MySQL.

4.7.3 Ministerio de Trabajo y Seguridad Social

Nivel de riesgo	CVSS	Total Vulnerabilidades
Crítico	7.0-10	2
Medio	4.0-6.9	9
Bajo	0.1-3.9	0
Informativo	0.0	3

ID	VULNERABILIDAD	CVSS	NIVEL
4.6.3.12	Métodos inseguros habilitados	**7.5**	Crítico
4.6.3.14	SSL Protocolo versión 2 y 3 detectado	**7.5**	Crítico
4.6.3.15	Uso de HTTP para autenticación	**6.5**	Medio
4.6.3.16	Puerto RDP 3389 abierto al público	**6.5**	Medio
4.6.3.24	HTTPS no configurado	**5.3**	Medio
4.6.3.25	Posibilidad de listar usuarios	**5.3**	Medio

4.6.3.26	Gestión de archivos	**5.3**	Medio
4.6.3.28	Páginas de pruebas encontradas en sitios de producción	**5.3**	Medio
4.6.3.35	Exposición de información sensible	**5.3**	Medio
4.6.3.36	SSL Medium Strength Cipher Suites Supported	**5.3**	Medio
4.6.3.37	SSL 64-bit Block Size Cipher Suites Supported (SWEET32)	**5.3**	Medio
4.6.3.48	Versión de Microsoft IIS expuesta	-	Informativo
4.6.3.50	Servidor FTP Detectado	-	Informativo
4.6.3.51	VERITAS NetBackup Agent Detection	-	Informativo

Resumen de las vulnerabilidades encontradas

Refiérase a la sección 4.6.3.12 para mayor información del uso de métodos inseguros habilitados, esto porque el método **TRACE** se encuentra habilitado. A continuación se muestran algunos vínculos de referencia.

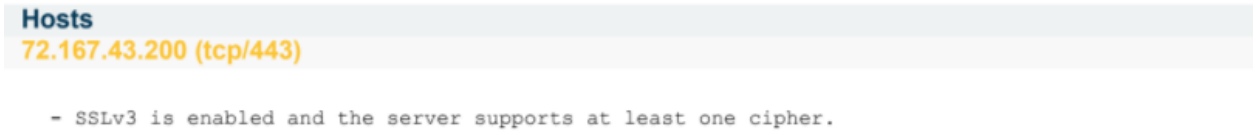
Insecure methods found (trace) on http://www.mtss.go.cr/

Insecure methods found (trace) on http://www.mtss.go.cr/tramites-servicios/formularios/

Insecure methods found (trace) on http://www.mtss.go.cr/tramites-servicios/

Vulnerabilidad descrita en la sección 4.6.3.14. El servidor permite el cifrado de tráfico utilizando un protocolo con debilidades conocidas, SSL 2.0 y 3.0.

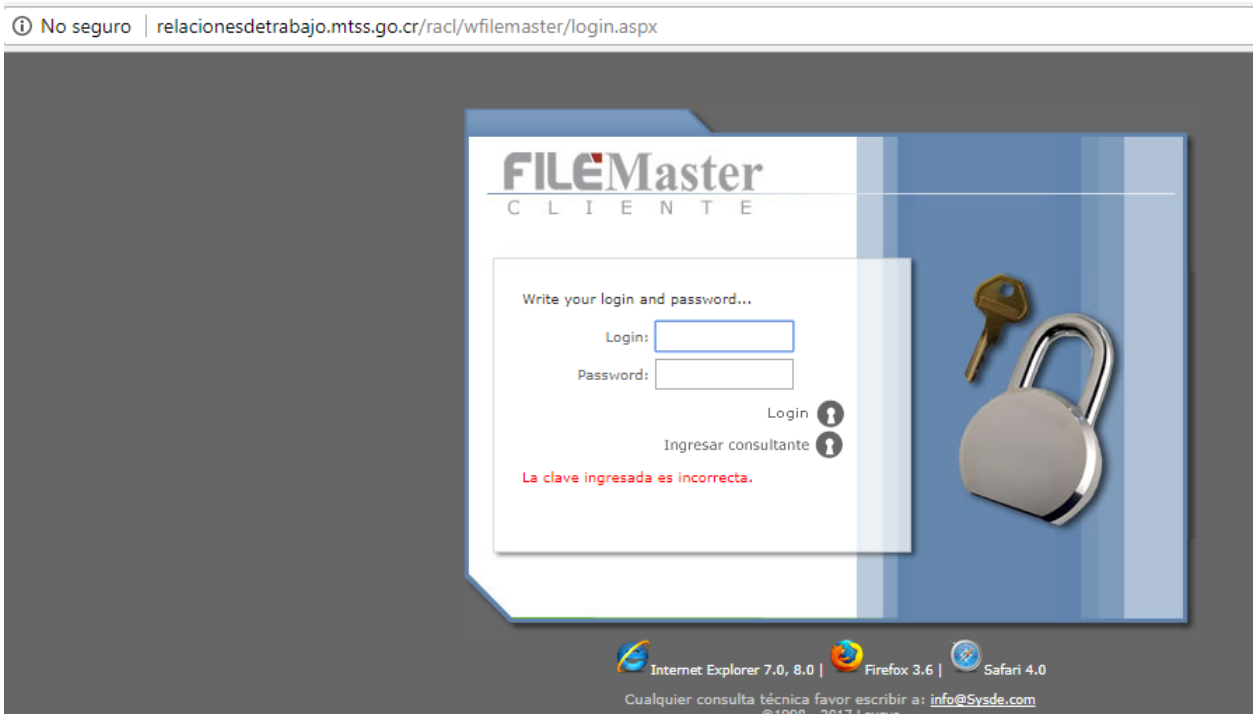
Figura 25. SSLv3 habilitado



Refiérase a la sección 4.6.3.15 para mayor información del uso de HTTP para autenticación.

- http://relacionesdetrabajo.mtss.go.cr/racl/wfilemaster/login.aspx

Figura 26. Ingreso al sistema con texto plano MTSS



Vulnerabilidad descrita en la sección 4.6.3.16. Es permitido acceso remoto para un usuario de Windows de forma gráfica al servidor. Puerto TCP/3389 abierto.

Refiérase a la sección 4.6.3.24 para mayor información del uso de certificados digitales.

Vulnerabilidad descrita en la sección 4.6.3.25. . La posibilidad de listar usuarios al intentar ingresar, para verificar su validez, es un tipo de ataque en que el script de validación de la página web muestra al atacante si el nombre de usuario proporcionado es correcto o no.

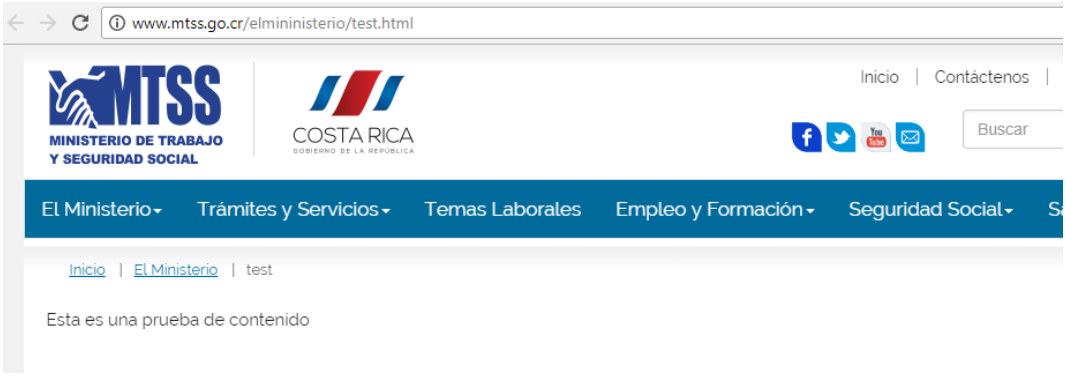
Figura 27. Listado de usuarios permite adivinar contraseñas



Refiérase a la sección 4.6.3.26 para mayor información relacionada a la gestión de archivos en sitios web.

Refiérase a la sección 4.6.3.28 para conocer sobre las páginas de prueba en sitios de producción.

Figura 28. Página de prueba del MTSS en producción



Refiérase a la sección 4.6.3.35 para obtener información acerca de la exposición de información sensible. Al obtener el usuario "admin", en la sección 4.6.3.25 se procede a realizar pruebas con algunas contraseñas de uso común para evitar el uso de fuerza bruta que podría bloquear el usuario. Se determina que la contraseña es "123456", y se da como exitosa la prueba de acceso.

Figura 29. Ingreso al sistema con usuario administrador en MTSS



Figura 30. Conexión válida para modificar información

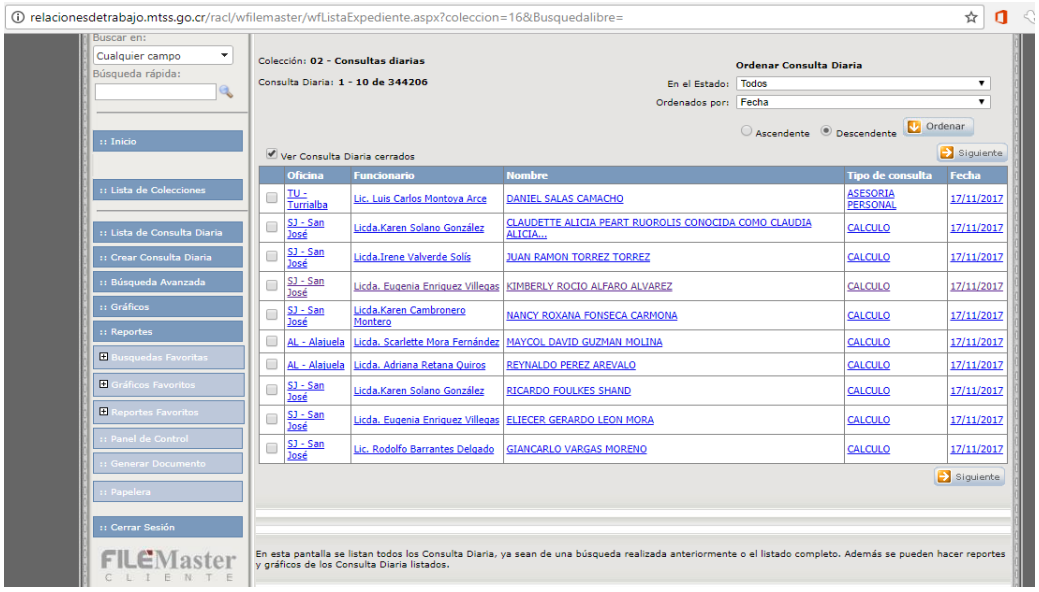
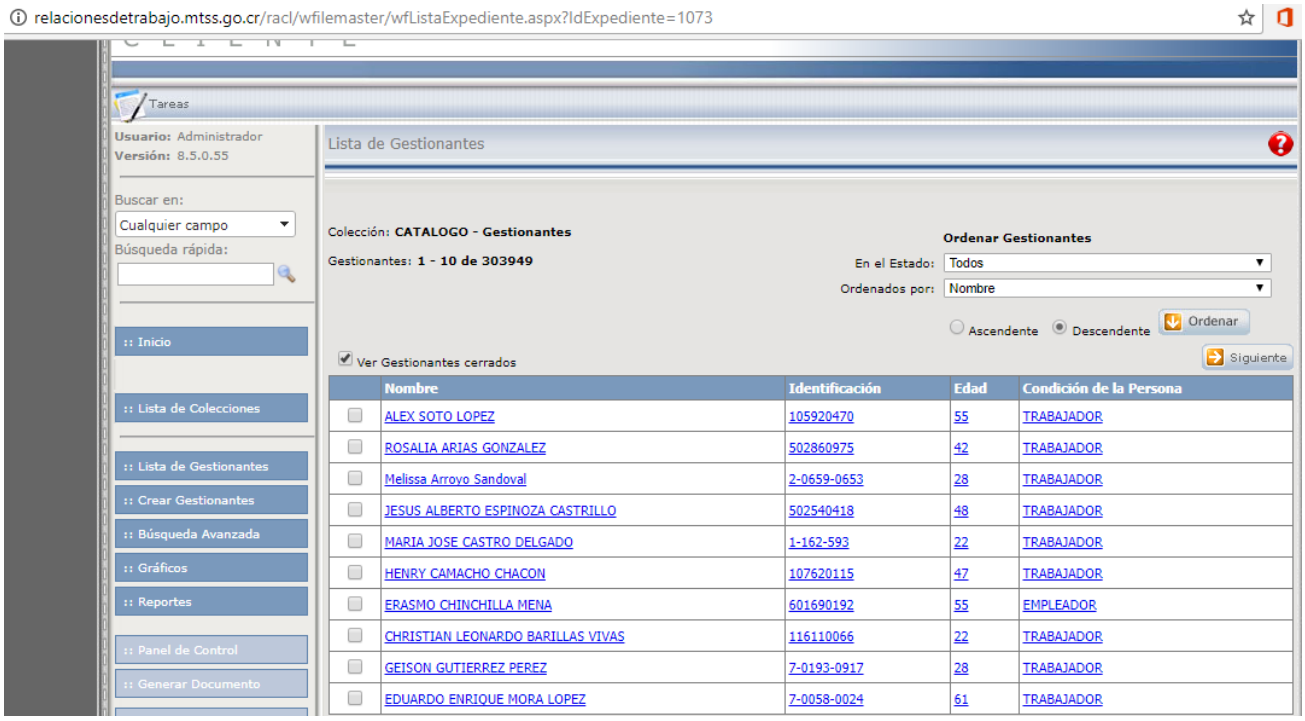


Figura 31. Consulta realizada tipo cálculo



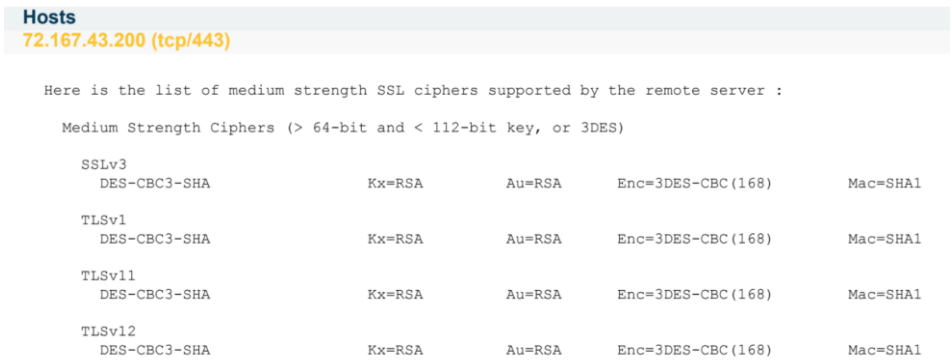
Figura 32. Catálogo de gestiones

Al ingresar como usuario administrador se pueden verificar las gestiones que ingresan los ciudadanos al Ministerio de Trabajo.



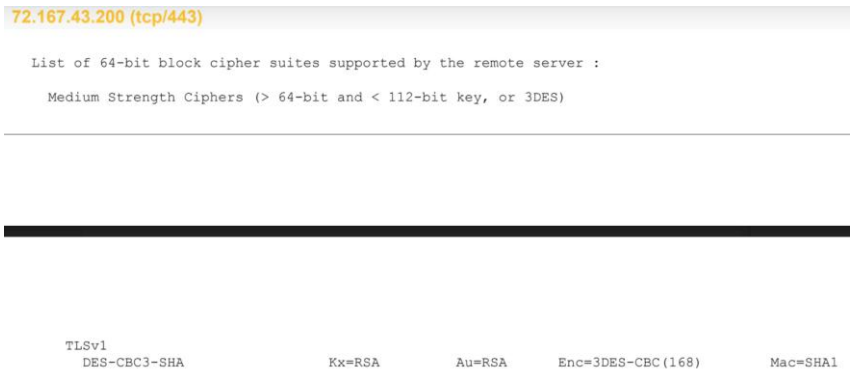
Vulnerabilidad descrita en la sección 4.6.3.36. El sitio web admite el uso de cifrado SSL que ofrece un cifrado de intensidad media. Se utiliza longitudes de clave de al menos 64 bits y menos de 112 bits o cifrado como 3DES.

Figura 33. Cifrado medio



Vulnerabilidad descrita en la sección 4.6.3.37. El sitio web admite el uso de un cifrado de bloques con bloques de 64 bits en uno o más conjuntos de cifrado.

Figura 34. Bloques de cifrado



Refiérase a la sección 4.6.3.48 para obtener información en relación con la versión expuesta.

Figura 35. Versión de Microsoft



Refiérase a la sección 4.6.3.50 para obtener información acerca del FTP.

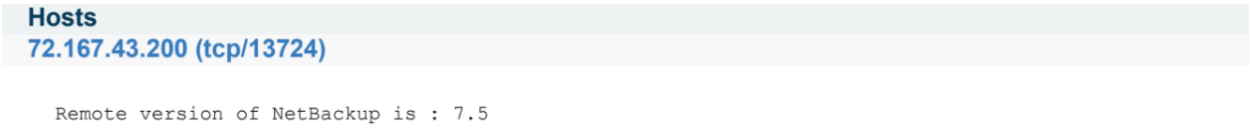
A continuación se muestra el banner de un servidor como ejemplo de esta vulnerabilidad.

Figura 36. Mensaje de bienvenida FTP



Vulnerabilidad descrita en la sección 4.6.3.51. El Agente veritas permite que un usuario autorizado para conectarse a este servicio pueda usarlo como un shell remoto con privilegios del sistema enviando mensajes 'command_EXEC_LIST'.

Figura 37. Revela la versión



4.7.4 Ministerio de Ambiente y Energía (MINAE)

Nivel de riesgo	CVSS	Total Vulnerabilidades
Crítico	7.0-10	2
Medio	4.0-6.9	6
Bajo	0.1-3.9	0
Informativo	0.0	2

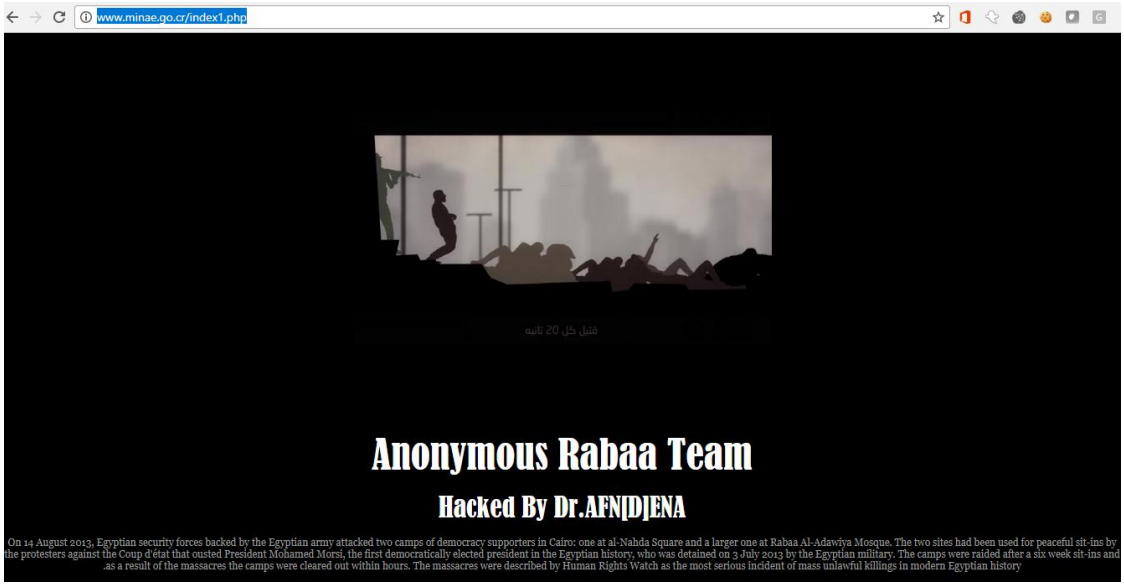
ID	VULNERABILIDAD	CVSS	NIVEL
4.6.3.9	Sitio Web hackeado	**9.8**	Crítico
4.6.3.12	Métodos inseguros habilitados	**7.5**	Crítico
4.6.3.15	Uso de HTTP para autenticación	**6.5**	Medio
4.6.3.21	DNS Server Spoofed Request Amplification DDoS	**5.8**	Medio
4.6.3.24	HTTPS no configurado	**5.3**	Medio
4.6.3.26	Gestión de archivos	**5.3**	Medio

4.6.3.27	Mensaje de error en aplicación	**5.3**	Medio
4.6.3.40	No se detecta Web Application Firewall (WAF)	**5.3**	Medio
4.6.3.47	Referencias a un sitio web en lista negra	-	Informativo
4.6.3.50	Servidor FTP Detectado	-	Informativo

Resumen de las vulnerabilidades encontradas

Vulnerabilidad descrita en la sección 4.6.3.9. En este sitio web al ingresar una referencia a un objeto o ruta inexistente, se realiza una redirección a un componente de la página que fue vulnerado y posee información de los atacantes. Cualquier referencia no encontrada en vez de devolver el error "404 Not Found" devuelve lo siguiente:

Figura 38. Sitio del MINAE Hackeado



Algunos URLs de referencia:

- http://www.minae.go.cr/admin
- http://www.minae.go.cr/test

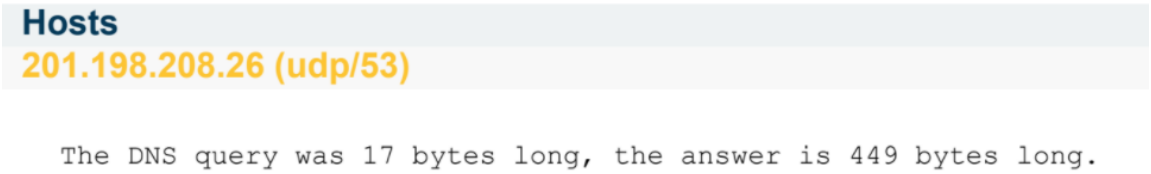
Refiérase a la sección 4.6.3.12 para mayor información del uso de métodos inseguros habilitados, esto porque el método **TRACE** se encuentra habilitado.

Refiérase a la sección 4.6.3.15 para mayor información del uso de HTTP para autenticación.

- http://www.minae.go.cr/denuncias/login.aspx

Vulnerabilidad descrita en la sección 4.6.3.21. Con la solicitud de amplificación DNS un atacante remoto puede lanzar un ataque de denegación de servicio contra un servidor.

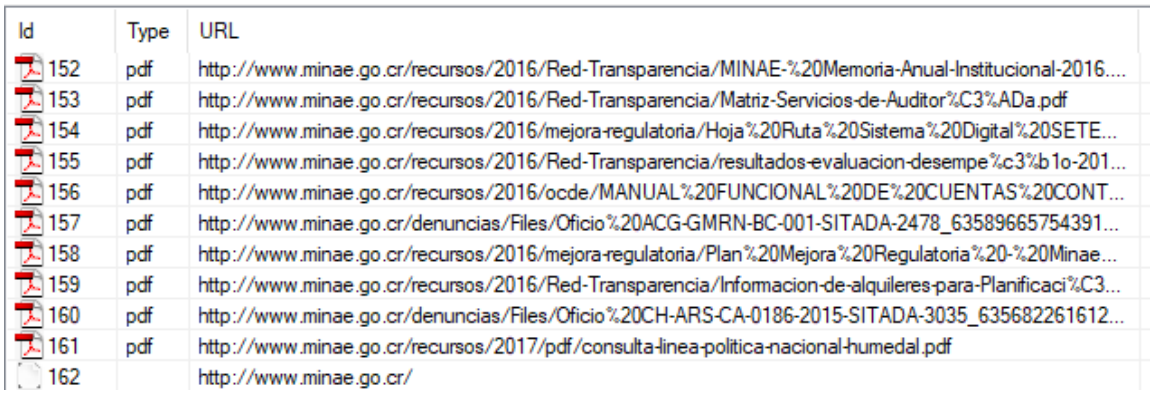
Figura 39. Amplificación de DNS



Refiérase a la sección 4.6.3.24 para mayores detalles del uso de certificados digitales para sitios web.

Refiérase a la sección 4.6.3.26 para mayor información relacionada a la gestión de archivos en sitios web, tomada con la herramienta FOCA.

Figura 40. Lista de archivos expuestos



Id	Type	URL
152	pdf	http://www.minae.go.cr/recursos/2016/Red-Transparencia/MINAE-%20Memoria-Anual-Institucional-2016....
153	pdf	http://www.minae.go.cr/recursos/2016/Red-Transparencia/Matriz-Servicios-de-Auditor%C3%ADa.pdf
154	pdf	http://www.minae.go.cr/recursos/2016/mejora-regulatoria/Hoja%20Ruta%20Sistema%20Digital%20SETE...
155	pdf	http://www.minae.go.cr/recursos/2016/Red-Transparencia/resultados-evaluacion-desempe%c3%b1o-201...
156	pdf	http://www.minae.go.cr/recursos/2016/ocde/MANUAL%20FUNCIONAL%20DE%20CUENTAS%20CONT...
157	pdf	http://www.minae.go.cr/denuncias/Files/Oficio%20ACG-GMRN-BC-001-SITADA-2478_63589665754391...
158	pdf	http://www.minae.go.cr/recursos/2016/mejora-regulatoria/Plan%20Mejora%20Regulatoria%20-%20Minae...
159	pdf	http://www.minae.go.cr/recursos/2016/Red-Transparencia/Informacion-de-alquileres-para-Planificaci%C3...
160	pdf	http://www.minae.go.cr/denuncias/Files/Oficio%20CH-ARS-CA-0186-2015-SITADA-3035_635682261612...
161	pdf	http://www.minae.go.cr/recursos/2017/pdf/consulta-linea-politica-nacional-humedal.pdf
162		http://www.minae.go.cr/

Refiérase a la sección 4.6.3.27 para conocer de los mensajes de error o advertencia de la aplicación.

Refiérase a la sección 4.6.3.40 para obtener información acerca de lo que es un WAF y la importancia de implementarlo.

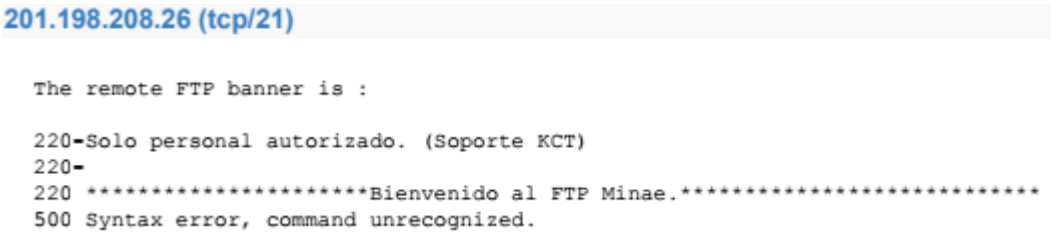
Vulnerabilidad descrita en la sección 4.6.3.47. Se ha encontrado dentro del sitio web la referencia hacia un sitio que aparece en una lista negra de Virus Total.

- http://cambioclimaticocr.com/

Refiérase a la sección 4.6.3.50 para obtener información acerca del FTP.

A continuación, se muestra el banner de un servidor como ejemplo de esta vulnerabilidad.

Figura 41. Mensaje de bienvenida del FTP



4.7.5 Ministerio de Hacienda

Nivel de riesgo	CVSS	Total Vulnerabilidades
Crítico	7.0-10	3
Medio	4.0-6.9	7
Bajo	0.1-3.9	1
Informativo	0.0	0

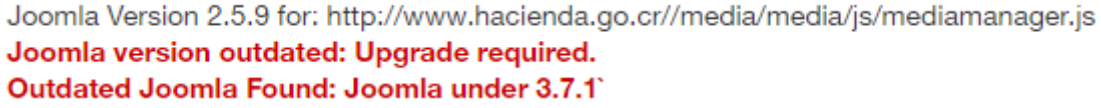
ID	VULNERABILIDAD	CVSS	NIVEL
4.6.3.8	Aplicación web desactualizada	**9.8**	Crítico

4.6.3.12	Métodos inseguros habilitados	**7.5**	Crítico
4.6.3.14	SSL Protocolo versión 2 y 3 detectado	**7.5**	Crítico
4.6.3.15	Uso de HTTP para autenticación	**6.5**	Medio
4.6.3.24	HTTPS no configurado	**5.3**	Medio
4.6.3.30	Exposición de XAMPP	**5.3**	Medio
4.6.3.31	Ataque remoto de DDoS por la versión del sistema operativo	**5.3**	Medio
4.6.3.32	El análisis en algunos hilos particulares de PHP permite DDoS	**5.3**	Medio
4.6.3.36	SSL Medium Strength Cipher Suites Supported	**5.3**	Medio
4.6.3.40	No se detecta Web Application Firewall (WAF)	**5.3**	Medio
4.6.3.44	Soporta SSL RC4 Cipher Suites	**3.7**	Informativo

Resumen de las vulnerabilidades encontradas

Vulnerabilidad descrita en la sección 4.6.3.8. Se ha detectado que la aplicación web está desactualizada una o más versiones atrás de la última versión del producto, esto conlleva a que la aplicación no cuente con los últimos parches de seguridad solucionados en la aplicación.

Figura 42. Versión de Joomla



El siguiente enlace muestra las vulnerabilidades de la versión 2.5.9. https://www.cvedetails.com/vulnerability-list/vendor_id-3496/product_id-16499/version_id-145393/Joomla-Joomla--2.5.9.html

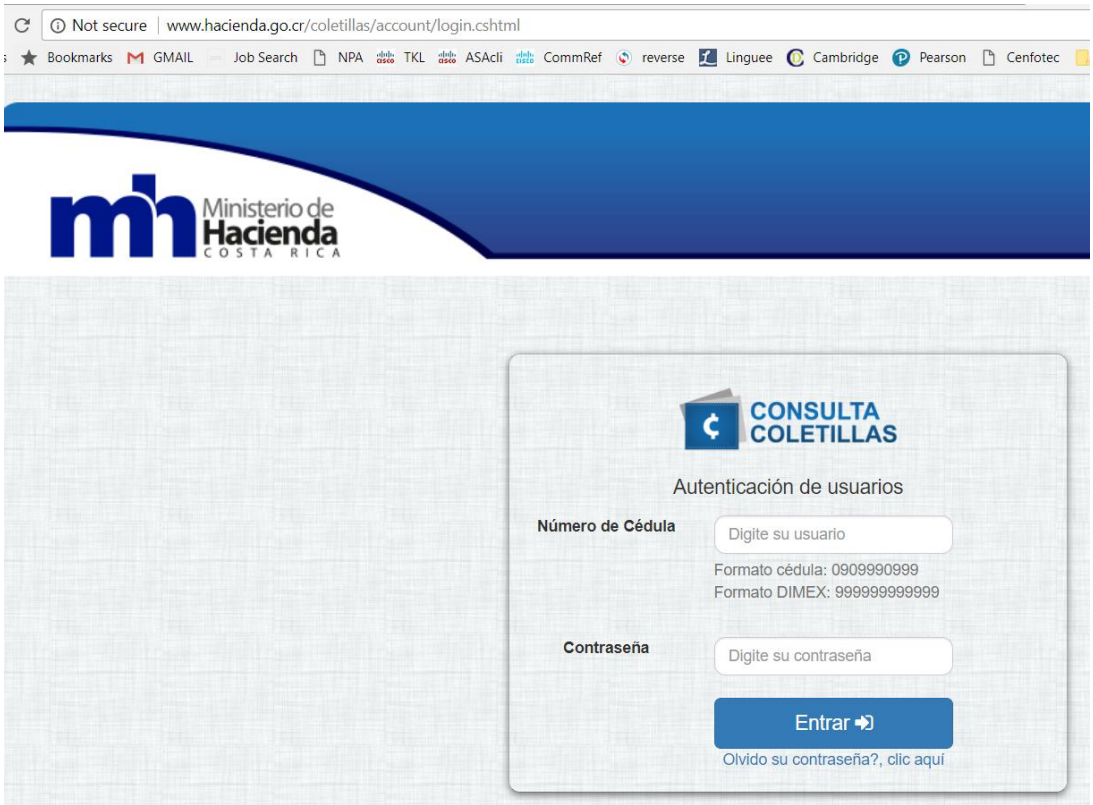
Refiérase a la sección 4.6.3.12 para mayor información del uso de métodos inseguros habilitados, esto porque el método **TRACE** se encuentra habilitado.

Vulnerabilidad descrita en la sección 4.6.3.14. El servidor permite el cifrado de tráfico utilizando un protocolo con debilidades conocidas, SSL 2.0 y 3.0.

Refiérase a la sección 4.6.3.15 para mayor información del uso de HTTP para autenticación.

- http://www.hacienda.go.cr/coletillas/account/login.cshtml
- http://www.hacienda.go.cr/mesaserviciotic/
- http://www.hacienda.go.cr/TICA/web/hCnConsVariasE.aspx

Figura 43. Formulario de acceso con HTTP en Hacienda



Refiérase a la sección 4.6.3.24 para mayores detalles del uso de certificados digitales para sitios web.

Vulnerabilidad descrita en la sección 4.6.3.30. Se expone información sensible que revela algunos detalles importantes de la aplicación.

Figura 44. XAMPP expuesto en Hacienda



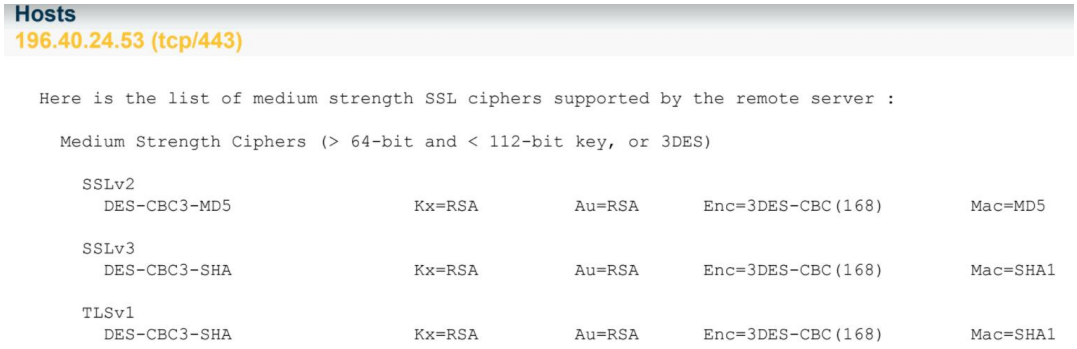
Vulnerabilidad descrita en la sección 4.6.3.31. Se ha encontrado una vulnerabilidad de denegación de servicio en Apache, que podría causar un consumo elevado de CPU y degradar el servicio.

Las versiones de Apache afectadas son 1.3.x, 2.0.x hasta 2.0.64, y 2.2.x hasta 2.2.19.

Vulnerabilidad descrita en la sección 4.6.3.32. Cuando PHP analiza algunas cadenas se congela, lo que causa una denegación de servicio. Las versiones de PHP que se han visto afectadas son la 5.3 hasta la versión 5.3.5 y la 5.2 hasta la versión 5.2.17.

Vulnerabilidad descrita en la sección 4.6.3.36. El sitio web admite el uso de cifrado SSL que ofrece un cifrado de intensidad media. Se utiliza longitudes de clave de al menos 64 bits y menos de 112 bits o cifrado como 3DES.

Figura 45. Cifrado medio



Refiérase a la sección 4.6.3.40 para obtener información acerca de lo que es un WAF y la importancia de implementarlo.

Vulnerabilidad descrita en la sección 4.6.3.44. Al admitir SSL RC4, si el texto sin cifrar se encripta repetidamente (por ejemplo, cookies HTTP) y un atacante puede obtener muchos (por ejemplo, decenas de millones) de textos cifrados, el atacante puede derivar el texto sin cifrar.

4.7.6 Ministerio de Ciencia, Tecnología y Telecomunicaciones (MICITT)

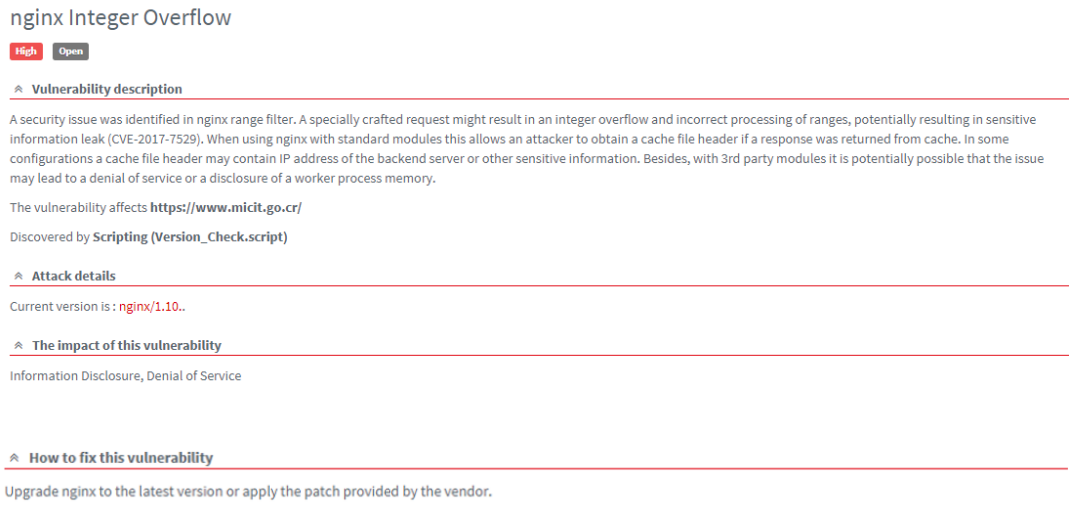
Nivel de riesgo	CVSS	Total Vulnerabilidades
Crítico	7.0-10	1
Medio	4.0-6.9	4
Bajo	0.1-3.9	0
Informativo	0.0	0

ID	VULNERABILIDAD	CVSS	NIVEL
4.6.3.8	Aplicación web desactualizada	**9.8**	Crítico
4.6.3.17	Archivo inseguro crossdomain.xml	**6.5**	Medio
4.6.3.26	Gestión de archivos	**5.3**	Medio
4.6.3.31	Ataque remoto de DDoS por la versión del sistema operativo	**5.3**	Medio
4.6.3.40	No se detecta Web Application Firewall (WAF)	**5.3**	Medio

Resumen de las vulnerabilidades encontradas

Vulnerabilidad descrita en la sección 4.6.3.8. Se ha detectado que la aplicación web está desactualizada, la versión actual es nginx 1.10, se recomienda actualizar a la última versión o aplicar el parche recomendado.

Figura 46. Aplicación web desactualizada en MICITT



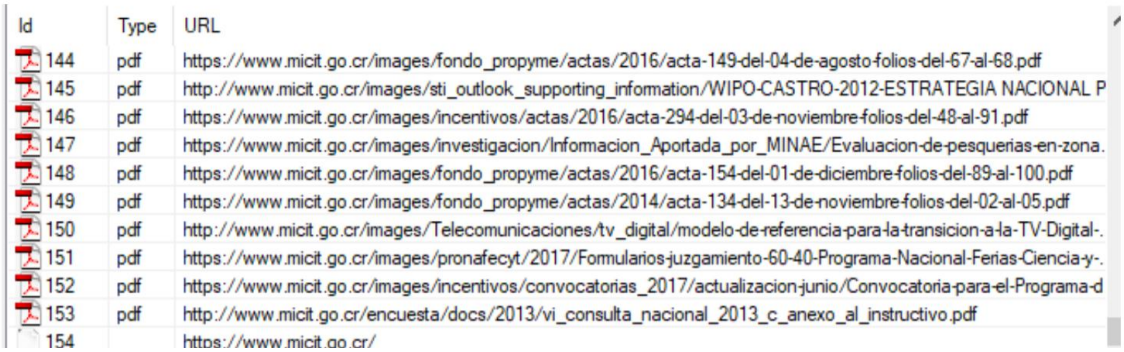
Vulnerabilidad descrita en la sección 4.6.3.17. El archivo crossdomain.xml que es implementado en un sitio web puede estar abierto a todos los dominios con el simple uso de un asterisco "*", soporta así lo que se conoce como "wildcard".

<cross-domain-policy>
<allow-access-from domain="" />*
</cross-domain-policy>

Afecta directamente al URL: https://micitt.go.cr/

Refiérase a la sección 4.6.3.26 para mayor información relacionada a la gestión de archivos en sitios web.

Figura 47. Archivos expuestos



Id	Type	URL
144	pdf	https://www.micit.go.cr/images/fondo_propyme/actas/2016/acta-149-del-04-de-agosto-folios-del-67-al-68.pdf
145	pdf	http://www.micit.go.cr/images/sti_outlook_supporting_information/WIPO-CASTRO-2012-ESTRATEGIA NACIONAL P
146	pdf	https://www.micit.go.cr/images/incentivos/actas/2016/acta-294-del-03-de-noviembre-folios-del-48-al-91.pdf
147	pdf	https://www.micit.go.cr/images/investigacion/Informacion_Aportada_por_MINAE/Evaluacion-de-pesquerias-en-zona.
148	pdf	https://www.micit.go.cr/images/fondo_propyme/actas/2016/acta-154-del-01-de-diciembre-folios-del-89-al-100.pdf
149	pdf	https://www.micit.go.cr/images/fondo_propyme/actas/2014/acta-134-del-13-de-noviembre-folios-del-02-al-05.pdf
150	pdf	http://www.micit.go.cr/images/Telecomunicaciones/tv_digital/modelo-de-referencia-para-la-transicion-a-la-TV-Digital-.
151	pdf	https://www.micit.go.cr/images/pronafecyt/2017/Formularios-juzgamiento-60-40-Programa-Nacional-Ferias-Ciencia-y-.
152	pdf	https://www.micit.go.cr/images/incentivos/convocatorias_2017/actualizacion-junio/Convocatoria-para-el-Programa-d
153	pdf	http://www.micit.go.cr/encuesta/docs/2013/vi_consulta_nacional_2013_c_anexo_al_instructivo.pdf
154		https://www.micit.go.cr/

Refiérase a la sección 4.6.3.40 para obtener información acerca de lo que es un WAF y la importancia de implementarlo.

4.7.7 Ministerio de Seguridad Pública

Nivel de riesgo	CVSS	Total Vulnerabilidades
Crítico	7.0-10	5
Medio	4.0-6.9	5
Bajo	0.1-3.9	0

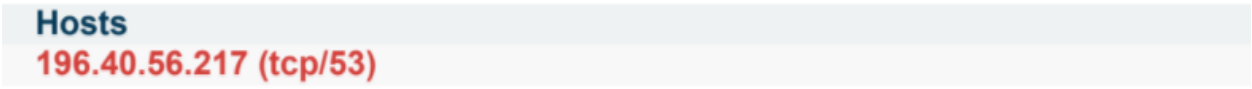
Informativo	0.0	1

ID	VULNERABILIDAD	CVSS	NIVEL
4.6.3.2	ISC BIND < 4.9.8 / 8.2.3 Multiple Remote Overflows	**10.0**	Crítico
4.6.3.3	ISC BIND < 4.9.5 DNS Resolver Functions Remote Overflow	**10.0**	Crítico
4.6.3.5	ISC BIND < 4.9.7-REL / 8.2.2-P5 Multiple Remote Vulnerabilities	**10.0**	Crítico
4.6.3.6	ISC BIND < 4.9.7 / 8.1.2 Inverse-Query Remote Overflow	**10.0**	Crítico
4.6.3.12	Métodos inseguros habilitados	**7.5**	Crítico
4.6.3.15	Uso de HTTP para autenticación	**6.5**	Medio
4.6.3.24	HTTPS no configurado	**5.3**	Medio
4.6.3.26	Gestión de archivos	**5.3**	Medio
4.6.3.35	Exposición de información sensible	**5.3**	Medio
4.6.3.40	No se detecta Web Application Firewall (WAF)	**5.3**	Medio
4.6.3.48	Versión de Microsoft IIS expuesta	**-**	Informativo

Resumen de las vulnerabilidades encontradas

Las vulnerabilidades descritas desde la sección 4.6.3.2 hasta la 4.6.3.6 se ven afectadas por la versión del servidor, y por varias vulnerabilidades de desbordamiento de búfer que pueden permitir que un atacante ejecute código arbitrario en el sitio web.

Figura 48. DNS expuesto debido a la versión del servidor



Refiérase a la sección 4.6.3.12 para mayor información del uso de métodos inseguros habilitados, esto porque el método **TRACE** se encuentra habilitado. A continuación se muestran algunos vínculos de referencia.

Insecure methods found (trace) on

http://www.seguridadpublica.go.cr/direccion/armamento/servicios/ArmasyExplosivos/

Insecure methods found (trace) on

http://www.seguridadpublica.go.cr/sala_prensa/comunicados/2012/marzo/

Insecure methods found (trace) on

http://www.seguridadpublica.go.cr/sala_prensa/revista/2010/

Refiérase a la sección 4.6.3.15 para mayor información del uso de HTTP para autenticación.

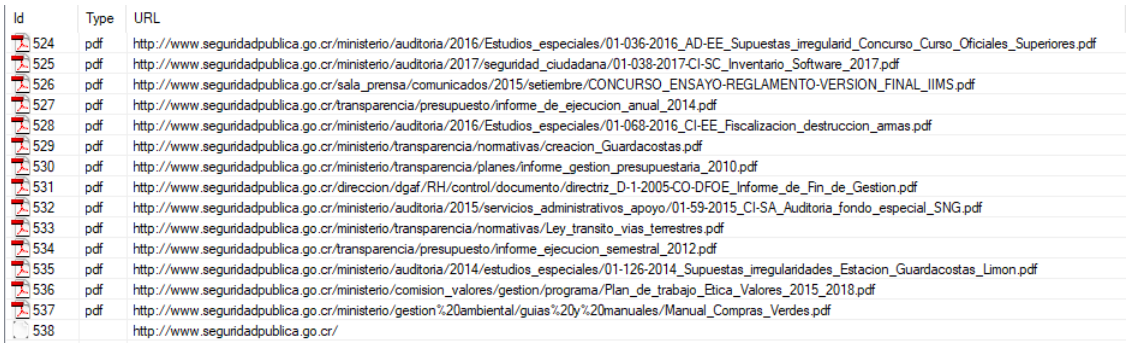
Figura 49. Formulario de acceso en HTTP MSP



Refiérase a la sección 4.6.3.24 para mayor información del uso de certificados digitales.

Refiérase a la sección 4.6.3.26 para mayor información relacionada a la gestión de archivos en sitios web.

Figura 50. Archivos expuestos



Id	Type	URL
524	pdf	http://www.seguridadpublica.go.cr/ministerio/auditoria/2016/Estudios_especiales/01-036-2016_AD-EE_Supuestas_irregularid_Concurso_Curso_Oficiales_Superiores.pdf
525	pdf	http://www.seguridadpublica.go.cr/ministerio/auditoria/2017/seguridad_ciudadana/01-038-2017-CI-SC_Inventario_Software_2017.pdf
526	pdf	http://www.seguridadpublica.go.cr/sala_prensa/comunicados/2015/setiembre/CONCURSO_ENSAYO-REGLAMENTO-VERSION_FINAL_IIMS.pdf
527	pdf	http://www.seguridadpublica.go.cr/transparencia/presupuesto/informe_de_ejecucion_anual_2014.pdf
528	pdf	http://www.seguridadpublica.go.cr/ministerio/auditoria/2016/Estudios_especiales/01-068-2016_CI-EE_Fiscalizacion_destruccion_armas.pdf
529	pdf	http://www.seguridadpublica.go.cr/ministerio/transparencia/normativas/creacion_Guardacostas.pdf
530	pdf	http://www.seguridadpublica.go.cr/ministerio/transparencia/planes/informe_gestion_presupuestaria_2010.pdf
531	pdf	http://www.seguridadpublica.go.cr/direccion/dgaf/RH/control/documento/directriz_D-1-2005-CO-DFOE_Informe_de_Fin_de_Gestion.pdf
532	pdf	http://www.seguridadpublica.go.cr/ministerio/auditoria/2015/servicios_administrativos_apoyo/01-59-2015_CI-SA_Auditoria_fondo_especial_SNG.pdf
533	pdf	http://www.seguridadpublica.go.cr/ministerio/transparencia/normativas/Ley_transito_vias_terrestres.pdf
534	pdf	http://www.seguridadpublica.go.cr/transparencia/presupuesto/informe_ejecucion_semestral_2012.pdf
535	pdf	http://www.seguridadpublica.go.cr/ministerio/auditoria/2014/estudios_especiales/01-126-2014_Supuestas_irregularidades_Estacion_Guardacostas_Limon.pdf
536	pdf	http://www.seguridadpublica.go.cr/ministerio/comision_valores/gestion/programa/Plan_de_trabajo_Etica_Valores_2015_2018.pdf
537	pdf	http://www.seguridadpublica.go.cr/ministerio/gestion%20ambiental/guias%20y%20manuales/Manual_Compras_Verdes.pdf
538		http://www.seguridadpublica.go.cr/

Refiérase a la sección 4.6.3.35 para obtener información acerca exposición de información sensible.

Archivos que pueden revelar información sensible:

- http://www.seguridadpublica.go.cr/transparencia/rh/resultado_evaluacion_desempeno_2016,_al_28-09-2017.xlsx

Informes de auditoría donde se mencionan expedientes no encontrados:

- http://www.seguridadpublica.go.cr/ministerio/auditoria/2017/seguridad_ciudadana/01-038-2017-CI-SC_Inventario_Software_2017.pdf
- http://www.seguridadpublica.go.cr/ministerio/auditoria/2016/Estudios_especiales/01-036-2016_AD-EE_Supuestas_irregularid_Concurso_Curso_Oficiales_Superiores.pdf

Refiérase a la sección 4.6.3.40 para obtener información acerca de lo que es un WAF y la importancia de implementarlo.

Refiérase a la sección 4.6.3.48 para obtener información en relación con la versión expuesta.

Figura 51. Versión de Microsoft



4.7.8 Ministerio de Gobernación y policía

Nivel de riesgo	CVSS	Total Vulnerabilidades
Crítico	7.0-10	6
Medio	4.0-6.9	4
Bajo	0.1-3.9	0
Informativo	0.0	3

ID	VULNERABILIDAD	CVSS	NIVEL
4.6.3.7	OpenSSL 'ChangeCipherSpec' MiTM Vulnerability	**9.8**	Crítico
4.6.3.8	Aplicación web desactualizada	**9.8**	Crítico
4.6.3.10	Divulgación de información en OpenSSL Heartbeat	**9.4**	Crítico
4.6.3.12	Métodos inseguros habilitados	**7.5**	Crítico
4.6.3.13	Ataque de denegación de servicio XML cuadrático	**7.5**	Crítico
4.6.3.14	SSL Protocolo versión 2 y 3 detectado	**7.5**	Crítico

4.6.3.15	Uso de HTTP para autenticación	**6.5**	Medio
4.6.3.24	HTTPS no configurado	**5.3**	Medio
4.6.3.35	Exposición de información sensible	**5.3**	Medio
4.6.3.40	No se detecta Web Application Firewall (WAF)	**5.3**	Medio
4.6.3.53	Microsoft SQL Server TCP/IP Listener Detection	-	Informativo
4.6.3.54	Microsoft SQL Server UDP Query Remote Version Disclosure	-	Informativo
4.6.3.55	Detección de servidor PostgreSQL	-	Informativo

Resumen de las vulnerabilidades encontradas

Vulnerabilidad descrita en la sección 4.6.3.7. El servicio OpenSSL es vulnerable a un ataque conocido como Man in the Middle attack (MiTM). El servicio remoto en el puerto 8445 aceptó un mensaje ChangeCipherSpec anterior, lo que provocó que la MAC y las claves de cifrado se derivaran por completo de la información pública. Se completó todo el protocolo de enlace SSL, y el servidor aceptó y produjo mensajes cifrados y autentificados usando estos métodos débiles.

Dirección IP 190.10.68.139 (tcp/8445)

Vulnerabilidad descrita en la sección 4.6.3.8. La aplicación web utilizada en el sitio está desactualizada, usa la versión 3.6 de WordPress, la cual fue publicada para su uso en agosto del 2013, y es vulnerable a SQL injection. Se recomienda actualizar a la última versión.

Figura 52. Versión de WordPress desactualizada



Vulnerabilidad descrita en la sección 4.6.3.10. Esta vulnerabilidad permite a un atacante leer la memoria del sistema, con lo que podría facilitar el acceso a información sensible como contraseñas y otros datos confidenciales.

Refiérase a la sección 4.6.3.12 para mayor información del uso de métodos inseguros habilitados, esto porque el método **TRACE** se encuentra habilitado, a continuación se muestran algunos vínculos de referencia.

Insecure methods found (trace) on http://www.mgp.go.cr/

Insecure methods found (trace) on http://www.mgp.go.cr/wordpress/wp-content/

Insecure methods found (trace) on http://www.mgp.go.cr/wordpress/wp-content/uploads/2016/

Insecure methods found (trace) on http://www.mgp.go.cr/wordpress/

Insecure methods found (trace) on http://www.mgp.go.cr/wordpress/wp-content/uploads/

Vulnerabilidad descrita en la sección 4.6.3.13. El problema del ataque XML cuadrático radica en el analizador de expansión de entidades XML que puede causar el agotamiento del CPU y memoria, además de que la base de datos del sitio alcance la cantidad máxima de conexiones abiertas.

- http://www.mgp.go.cr/wordpress/xmlrpc.php

Vulnerabilidad descrita en la sección 4.6.3.14. El servidor permite el cifrado de tráfico utilizando un protocolo con debilidades conocidas, SSL 2.0 y 3.0.

Figura 53. Soporta cifrado SSLv3



Refiérase a la sección 4.6.3.15 para mayor información del uso de HTTP para autenticación.

Refiérase a la sección 4.6.3.24 para mayores detalles del uso de certificados digitales para sitios web. En este caso no se debería redireccionar a un link que revela la información del gestor de contenido al intentar conectarse usando SSL. Esto también es considerado como exposición de información sensible.

Figura 54. No utiliza HTTPS



Refiérase a la sección 4.6.3.35 para obtener información acerca exposición de información sensible. Al cargar la página **http://www.mgp.go.cr** redirecciona a **http://www.mgp.go.cr/wordpress/** y expone el gestor de contenido inmediatamente.

Refiérase a la sección 4.6.3.40 para obtener información acerca de lo que es un WAF y la importancia de implementarlo.

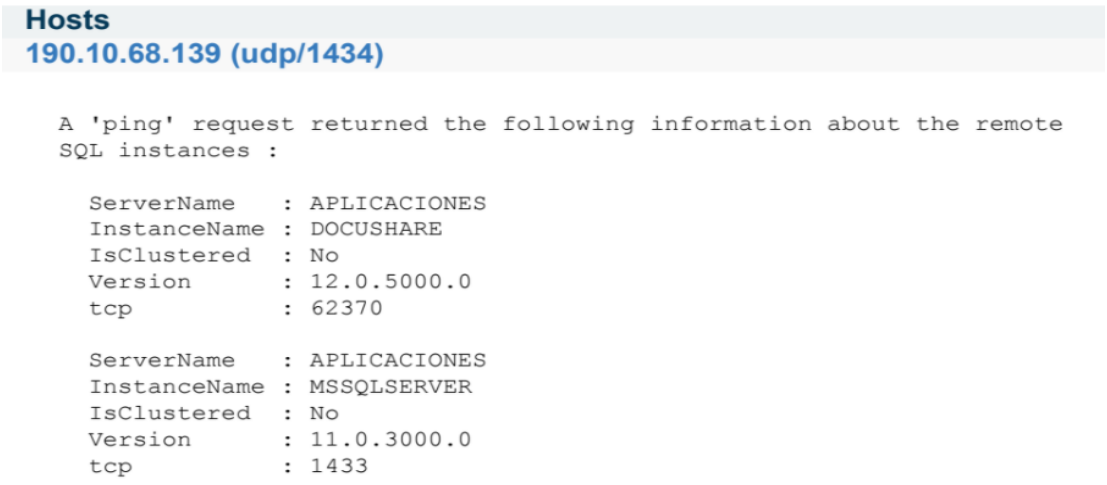
Vulnerabilidad descrita en la sección 4.6.3.53. Se ejecuta MSSQL.

Referencia: XREF OSVDB:112

Hosts 190.10.68.139 (tcp/1433). The remote SQL Server version is 11.0.3156.0. 190.10.68.139 (tcp/62370). The remote SQL Server version is 12.0.5203.0.

Vulnerabilidad descrita en la sección 4.6.3.54. Se determina que existe la posibilidad de que usuarios remotos pueden consultar el servidor de base de datos para la versión que se está ejecutando. La consulta tiene lugar en el mismo puerto UDP que maneja la asignación de múltiples instancias de SQL Server.

Figura 55. Instancias de SQL expuestas



Vulnerabilidad descrita en la sección 4.6.3.55. El servicio remoto es un servidor de base de datos PostgreSQL, o un derivado como EnterpriseDB.

Host: 190.10.68.139 (TCP/5432)

4.7.9 Ministerio de Justicia y Paz

Nivel de riesgo	CVSS	Total Vulnerabilidades
Crítico	7.0-10	2
Medio	4.0-6.9	5
Bajo	0.1-3.9	0
Informativo	0.0	1

ID	VULNERABILIDAD	CVSS	NIVEL
4.6.3.11	Listado de directorios (Directory Listing)	**7.5**	Crítico
4.6.3.12	Métodos inseguros habilitados	**7.5**	Crítico
4.6.3.19	Biblioteca de JavaScript vulnerable	**6.5**	Medio
4.6.3.24	HTTPS no configurado	**5.3**	Medio
4.6.3.26	Gestión de archivos	**5.3**	Medio
4.6.3.34	Posible exposición de directorios sensibles	**5.3**	Medio
4.6.3.40	No se detecta Web Application Firewall (WAF)	**5.3**	Medio
4.6.3.48	Versión de Microsoft IIS expuesta	-	Informativo

Resumen de las vulnerabilidades encontradas

Vulnerabilidad descrita en la sección 4.6.3.11. Esta configuración permite que el servidor web pueda mostrar la lista de archivos contenidos en ese directorio y navegar a través de ellos, poniendo en riesgo información sensible contenida en los archivos. Algunos ejemplos son:

- http://www.mjp.go.cr/servicios/computopenas/
- http://www.mjp.go.cr/servicios/
- http://www.mjp.go.cr/servicios/CCCEP/
- http://www.mjp.go.cr/servicios/AsesoriaJuridica/
- http://www.mjp.go.cr/pdf/
- http://www.mjp.go.cr/content/

Figura 56. Listado de directorios



Refiérase a la sección 4.6.3.12 para mayor información del uso de métodos inseguros habilitados, esto porque el método **TRACE** se encuentra habilitado. A continuación se muestran algunos vínculos de referencia.

Insecure methods found (trace) on http://www.mjp.go.cr/Servicios/

Insecure methods found (trace) on http://www.mjp.go.cr/Servicios/ComputoPenas/

Insecure methods found (trace) on http://www.mjp.go.cr/Servicios/AsesoriaJuridica/

Vulnerabilidad descrita en la sección 4.6.3.19. El ejemplo específico del URL afectado se muestra a continuación:

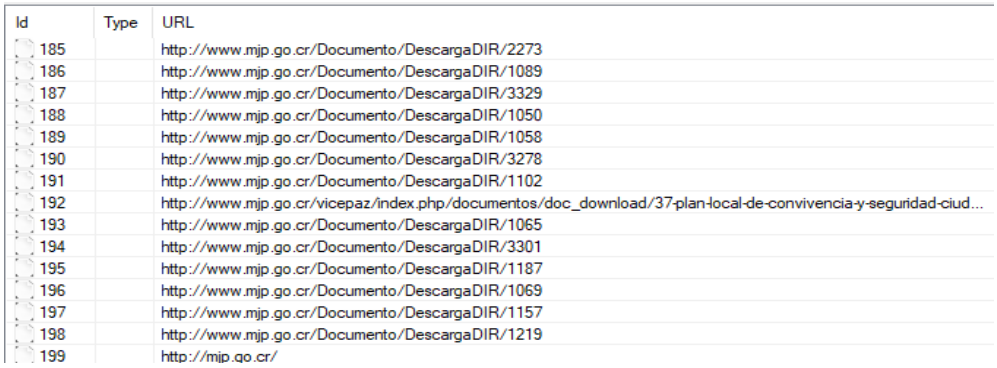
- http://www.mjp.go.cr/scripts/jquery-1.10.2.js

Se ha detectado “JavaScript library jQuery versión 1.10.2”.

Refiérase a la sección 4.6.3.24 para mayores detalles del uso de certificados digitales para sitios web.

Refiérase a la sección 4.6.3.26 para mayor información relacionada a la gestión de archivos en sitios web.

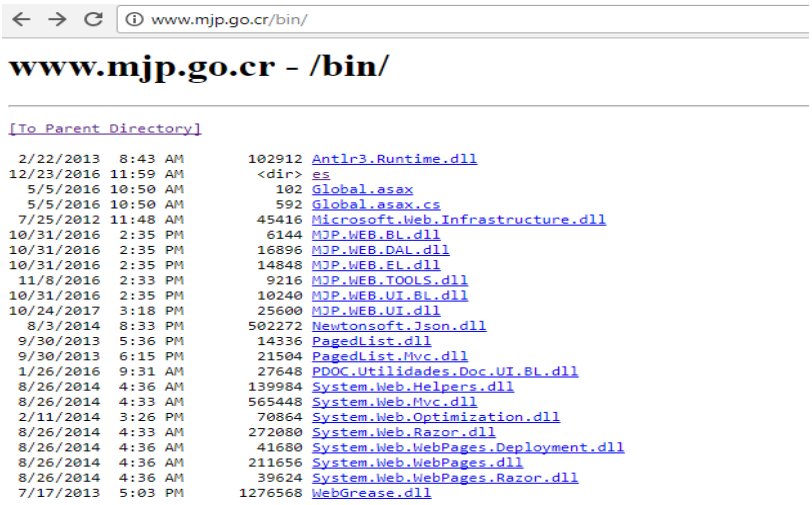
Figura 57. Archivos expuestos



Id	Type	URL
185		http://www.mjp.go.cr/Documento/DescargaDIR/2273
186		http://www.mjp.go.cr/Documento/DescargaDIR/1089
187		http://www.mjp.go.cr/Documento/DescargaDIR/3329
188		http://www.mjp.go.cr/Documento/DescargaDIR/1050
189		http://www.mjp.go.cr/Documento/DescargaDIR/1058
190		http://www.mjp.go.cr/Documento/DescargaDIR/3278
191		http://www.mjp.go.cr/Documento/DescargaDIR/1102
192		http://www.mjp.go.cr/vicepaz/index.php/documentos/doc_download/37-plan-local-de-convivencia-y-seguridad-ciud...
193		http://www.mjp.go.cr/Documento/DescargaDIR/1065
194		http://www.mjp.go.cr/Documento/DescargaDIR/3301
195		http://www.mjp.go.cr/Documento/DescargaDIR/1187
196		http://www.mjp.go.cr/Documento/DescargaDIR/1069
197		http://www.mjp.go.cr/Documento/DescargaDIR/1157
198		http://www.mjp.go.cr/Documento/DescargaDIR/1219
199		http://mjp.go.cr/

Refiérase a la sección 4.6.3.34 para obtener información acerca la posible exposición de directorios sensibles.

Figura 58. Directorio /bin expuesto



Refiérase a la sección 4.6.3.40 para obtener información acerca de lo que es un WAF y la importancia de implementarlo.

Vulnerabilidad descrita en la sección 4.6.3.48. Las respuestas HTTP devueltas por esta aplicación web incluyen un encabezado denominado Servidor, ese encabezado incluye la versión del servidor Microsoft IIS.

Figura 59. Versión de Microsoft



4.7.10 Ministerio de Cultura y Juventud

Nivel de riesgo	CVSS	Total Vulnerabilidades
Crítico	7.0-10	1
Medio	4.0-6.9	4
Bajo	0.1-3.9	0

Informativo	0.0	1

ID	VULNERABILIDAD	CVSS	NIVEL
4.6.3.12	Métodos inseguros habilitados	**7.5**	Crítico
4.6.3.15	Uso de HTTP para autenticación	**6.5**	Medio
4.6.3.24	HTTPS no configurado	**5.3**	Medio
4.6.3.26	Gestión de archivos	**5.3**	Medio
4.6.3.28	Páginas de pruebas encontradas en sitios de producción	**5.3**	Medio
4.6.3.46	Carga de archivos sin validación	**-**	Informativo

Resumen de las vulnerabilidades encontradas

Refiérase a la sección 4.6.3.12 para mayor información del uso de métodos inseguros habilitados, esto porque el método **TRACE** se encuentra habilitado. A continuación una imagen de FOCA sobre los métodos inseguros encontrados.

Figura 60. Método TRACE habilitado

Trace method found

high	Insecure methods found (trace) on http://www.mcj.go.cr/ministerio/organizacion/
high	Insecure methods found (trace) on http://www.mcj.go.cr/ministerio/organizacion/administrativo/
high	Insecure methods found (trace) on http://www.mcj.go.cr/ministerio/organizacion/administrativo/recur...
high	Insecure methods found (trace) on http://www.mcj.go.cr/ministerio/organizacion/administrativo/recur...
high	Insecure methods found (trace) on http://www.mcj.go.cr/ministerio/organizacion/administrativo/recur...
high	Insecure methods found (trace) on http://www.mcj.go.cr/ministerio/organizacion/staff/

Refiérase a la sección 4.6.3.15 para mayor información del uso de HTTP para autenticación.

- http://www.mcj.go.cr/Login.aspx

Refiérase a la sección 4.6.3.24 para mayores detalles del uso de certificados digitales para sitios web.

Refiérase a la sección 4.6.3.26 para mayor información relacionada a la gestión de archivos en sitios web.

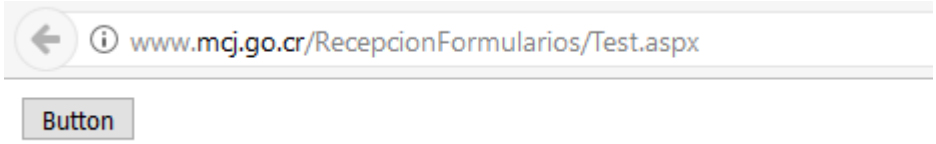
Figura 61. Archivos expuestos

Id	Type	URL
509	pdf	http://www.mcj.go.cr/comisiones/comisionvalores/Manual%20de%20Etica%20y%20Valores.pdf
510	pdf	http://www.mcj.go.cr/contactenos/archivos/Reglamento_39096.pdf
511	pdf	http://www.mcj.go.cr/ministerio/transparencia/informe%20de%20labores/ley%209398/2015-2016/Informe%202016%20%20CPJ.pdf
512	pdf	http://www.mcj.go.cr/ministerio/Transparencia/consulta_publica/DE_REGLAMENTO_TRAMITES%20_PATRIMONIO_03-04-17.pdf
513	pdf	http://www.mcj.go.cr/ministerio/organizacion/administrativo/informatica/politicasti/DI-PO-11-2014%20Politica%20para%20el%20control%20de%20contrasenas.pdf
514	pdf	http://www.mcj.go.cr/ministerio/Transparencia/consulta_publica/Formulario%20Recomendacion%20para%20exoneracion%20de%20impuestos%20de%20salida.pdf
515	pdf	http://www.mcj.go.cr/ministerio/transparencia/informe%20de%20labores/ley%209398/2015-2016/Informe%202016%20MCG.pdf
516	pdf	http://www.mcj.go.cr/ministerio/PND%20final.pdf
517	pdf	http://www.mcj.go.cr/ministerio/organizacion/administrativo/informatica/politicasti/DI-PO-01-2014%20Politica%20de%20Gestion%20de%20Soporte%20Tecnico.pdf
518	pdf	http://www.mcj.go.cr/ministerio/Transparencia/informe%20de%20labores/informeslabores/02.pdf
519	pdf	http://www.mcj.go.cr/ministerio/Transparencia/consulta_publica/REGLAMENTO_EDITORIAL_MHCJS_CORREGIDO.pdf
520	pdf	http://www.mcj.go.cr/ministerio/transparencia/informe%20de%20labores/ley%209398/2015-2016/Informe%202016%20CCPC.pdf
521	pdf	http://www.mcj.go.cr/ministerio/Transparencia/Financiero%20contable/Directriz%20Politica%20Interna%20implementacion%20de%20libros%20contables%20digitales.pdf
522	pdf	http://www.mcj.go.cr/ministerio/juventud/archivos/documentos/JUVENTUD,%20POBREZA%20Y%20EQUIDAD.pdf

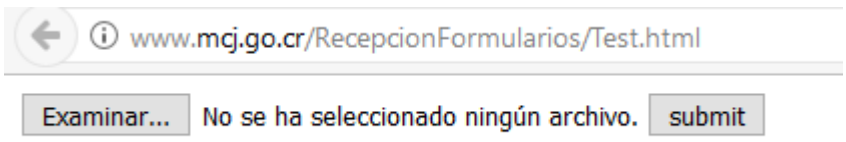
Refiérase a la sección 4.6.3.28 para conocer sobre las páginas de prueba en sitios de producción.

- http://www.mcj.go.cr/RecepcionFormularios/Test.aspx

Figura 62. Página de prueba en Ministerio de Cultura



- http://www.mcj.go.cr/RecepcionFormularios/Test.html



El sitio utiliza HTTP, sin embargo en un intento por verificar si existe una redirección a un sitio seguro, se determina que, usando HTTPS, se hace un redireccionamiento a otro sitio, que en apariencia no pertenece al Ministerio de Cultura.

- https://www.mcj.go.cr

Figura 63. Redirecciona a otro sitio al usar HTTPS



Refiérase a la sección 4.6.3.46 para obtener información relacionada a la carga de archivos.

- http://www.mcj.go.cr/formulariosfd/prestamoinstalacionescenac.aspx

4.7.11 Ministerio de Comercio Exterior (Comex)

Nivel de riesgo	CVSS	Total Vulnerabilidades
Crítico	7.0-10	1
Medio	4.0-6.9	9
Bajo	0.1-3.9	1

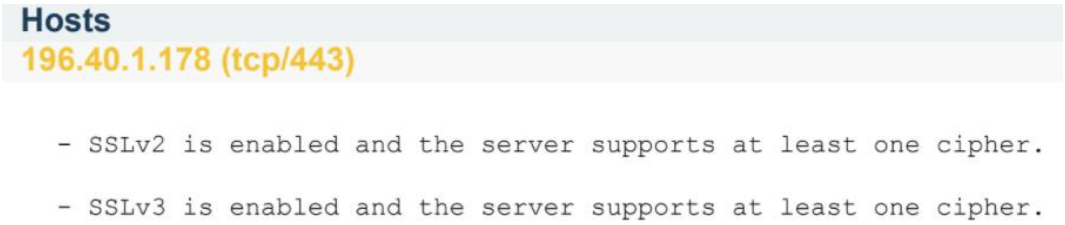
Informativo	0.0	1

ID	VULNERABILIDAD	CVSS	NIVEL
4.6.3.14	SSL Protocolo versión 2 y 3 detectado	**7.5**	Crítico
4.6.3.15	Uso de HTTP para autenticación	**6.5**	Medio
4.6.3.16	Puerto RDP 3389 abierto al público	**6.5**	Medio
4.6.3.23	Microsoft Windows Remote Desktop Protocol Server Man-in-the-Middle Weakness	**5.6**	Medio
4.6.3.24	HTTPS no configurado	**5.3**	Medio
4.6.3.26	Gestión de archivos	**5.3**	Medio
4.6.3.27	Mensaje de error en aplicación	**5.3**	Medio
4.6.3.28	Páginas de pruebas encontradas en sitios de producción	**5.3**	Medio
4.6.3.29	Divulgación de la versión ASP.NET	**5.3**	Medio
4.6.3.40	No se detecta Web Application Firewall (WAF)	**5.3**	Medio
4.6.3.42	Nivel de cifrado de Terminal Services medio o bajo	**3.7**	Bajo
4.6.3.46	Carga de archivos sin validación	**-**	Informativo

Resumen de las vulnerabilidades encontradas

Vulnerabilidad descrita en la sección 4.6.3.14. El servidor permite el cifrado de tráfico utilizando un protocolo con debilidades conocidas, SSL 2.0 y 3.0.

Figura 64. Cifrado SSLv2 y SSLv3



Vulnerabilidad descrita en la sección 4.6.3.15. No se recomienda el uso de HTTP en sitios donde se busca ingresar, por medio de un usuario y contraseña, a datos privados o descarga de archivos, ya que esta información podría ser fácilmente comprometida.

Vulnerabilidad descrita en la sección 4.6.3.16. Es permitido el acceso remoto para un usuario de Windows de forma gráfica al servidor, y por lo tanto actuar como un usuario local. Un atacante podría obtener los credenciales válidos de usuario y contraseña o bien montar un ataque de diccionario contra el servidor.

Figura 65. Acceso remoto al servidor de producción del Comex



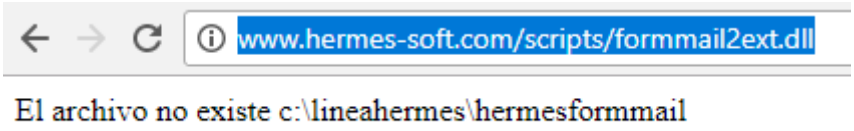
Vulnerabilidad descrita en la sección 4.6.3.23. La versión del Servidor es vulnerable a un ataque de hombre en el medio (MiTM). El cliente RDP no fuerza a realizar una validación de identidad del servidor al configurar el cifrado.

Refiérase a la sección 4.6.3.24 para mayores detalles del uso de certificados digitales para sitios web.

Refiérase a la sección 4.6.3.26 para mayor información relacionada a la gestión de archivos en sitios web.

Vulnerabilidad descrita en la sección 4.6.3.27. Los mensajes de error o advertencia de la aplicación pueden exponer información sensible, al cargar un archivo muy pesado en el formulario http://www.comex.go.cr/consultas/formulario.aspx, revela información de la ruta del servidor de un tercero (www.hermes-soft.com).

Figura 66. Mensaje de error que revela datos sensibles



Vulnerabilidad descrita en la sección 4.6.3.28. Se encontró que el sitio en producción tiene páginas de prueba, lo cual no es recomendable ya que esto podría provocar que se cometan errores no deseados en los desarrolladores, o bien que pueda ser expuesta información no deseada a los usuarios. Los siguientes enlaces son una muestra.

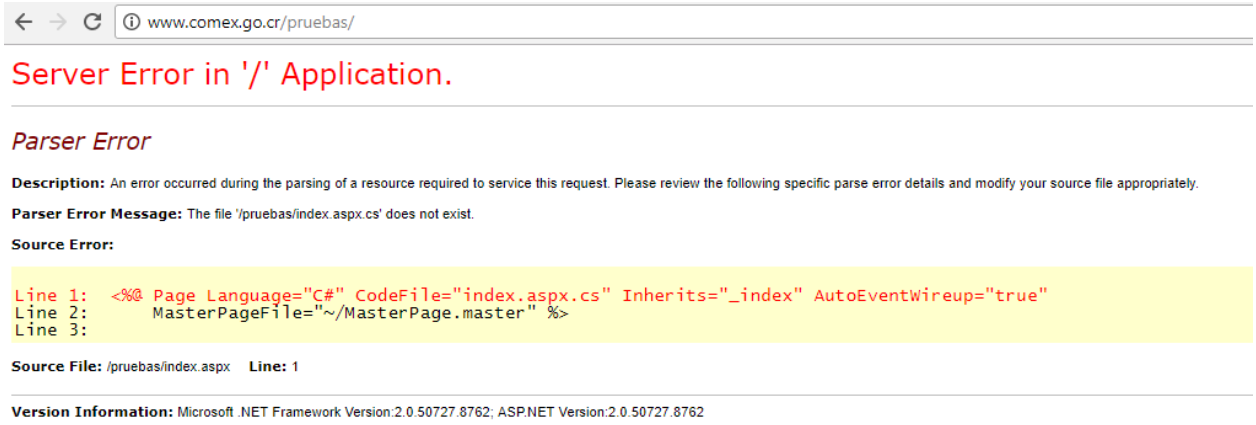
- http://www.comex.go.cr/pruebas/formulario.aspx
- http://www.comex.go.cr/prueba.aspx

Figura 67. Página de prueba del Comex en producción



Vulnerabilidad descrita en la sección 4.6.3.29. Las respuestas de HTTP que son devueltas por esta aplicación web incluyen un encabezado llamado X-AspNet-Version que revelan información de la versión.

Figura 68. ASP.NET expuesto



Refiérase a la sección 4.6.3.40 para obtener información acerca de lo que es un WAF y la importancia de implementarlo.

Vulnerabilidad descrita en la sección 4.6.3.42. El servicio de "Terminal Services remoto" no está configurado para usar una criptografía fuerte.

Vulnerabilidad descrita en la sección 4.6.3.46. Es posible cargar archivos al servidor, lo cual es potencialmente un riesgo si los archivos no son manipulados correctamente, un atacante podría enviar un archivo con el objetivo de ejecutar código malicioso en el servidor.

Figura 69. Carga de archivos sin validación en Comex



4.7.12 Instituto Nacional de las Mujeres (INAMU)

Nivel de riesgo	CVSS	Total Vulnerabilidades
Crítico	7.0-10	1
Medio	4.0-6.9	9
Bajo	0.1-3.9	0
Informativo	0.0	1

ID	VULNERABILIDAD	CVSS	NIVEL
4.6.3.4	Múltiples vulnerabilidades en el servidor de Oracle GlassFish	**10.0**	Crítico
4.6.3.15	Uso de HTTP para autenticación	**6.5**	Medio
4.6.3.21	DNS Server Spoofed Request Amplification DDoS	**5.8**	Medio
4.6.3.22	DNS Server Recursive Query Cache Poisoning Weakness	**5.8**	Medio
4.6.3.24	HTTPS no configurado	**5.3**	Medio
4.6.3.26	Gestión de archivos	**5.3**	Medio
4.6.3.35	Exposición de información sensible	**5.3**	Medio

4.6.3.38	Divulgación de información de transferencia de zona del servidor DNS (AXFR)	**5.3**	Medio
4.6.3.39	Internet Key Exchange (IKE) Aggressive Mode with Pre-Shared Key	**5.3**	Medio
4.6.3.41	Caché del servidor DNS divulga información de forma remota	**5.2**	Medio
4.6.3.49	Envío de contraseñas en texto plano al correo electrónico	-	Informativo

Resumen de las vulnerabilidades encontradas

Vulnerabilidad descrita en la sección 4.6.3.4. El Oracle GlassFish Server que se ejecuta en el sitio remoto es 3.1.2.x anterior a 3.1.2.15, por lo tanto, se ve afectado por múltiples vulnerabilidades.

Hostname: **www.inamu.go.cr**

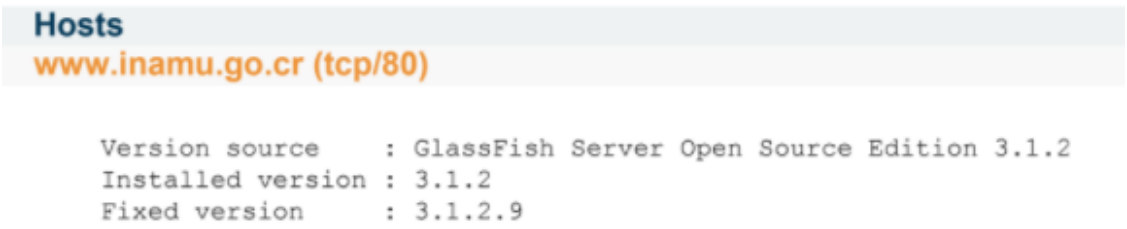
IP address: **181.193.29.34**

System Details:

Running on: GlassFish

Powered by: JSP/2.2

Figura 70. Versión de GlassFish desactualizada



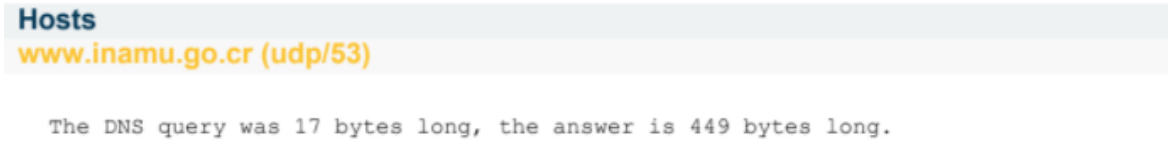
Refiérase a la sección 4.6.3.15 para mayor información del uso de HTTP para autenticación.

Figura 71. Ingreso al sito sin uso de HTTPs



Vulnerabilidad descrita en la sección 4.6.3.21. Con la solicitud de amplificación DNS, un atacante remoto puede lanzar un ataque de denegación de servicio contra un servidor.

Figura 72. Amplificación de DNS



Vulnerabilidad descrita en la sección 4.6.3.22. Si se está probando un servidor de DNS remoto, entonces permite que cualquiera lo use para resolver nombres de terceros.

Refiérase a la sección 4.6.3.24 para mayor información del uso de certificados digitales.

Refiérase a la sección 4.6.3.26 para mayor información relacionada a la gestión de archivos en sitios web.

Refiérase a la sección 4.6.3.35 para obtener información acerca exposición de información sensible. Permite descargar la base de datos de empleados con información sensible, un total de 301 empleados.

- http://www.inamu.go.cr/492

Figura 73. Base de datos de empleados del INAMU

	A	B	C	D	E	F	G	H
7	275	Verny Miguel Alvarado Rugama	valvarado	valvarado@inamu.go.cr			Prof. Ejecutivo	UNIDAD REGIONAL CHOROTEGA
8	276	Verónica Vargas Rodríguez	vvargas	vvargas@inamu.go.cr			Prof. Operativa 2	UNIDAD REGIONAL HUETAR NORTE
9	277	Victor Manuel Ortiz Campos	vortiz	vortiz@inamu.go.cr	8641		Operador Equipo Móvil	SERVICIOS GENERALES Y TRANSORTES
0	278	Virginia Badilla Solorzano	vbadilla	vbadilla@inamu.go.cr			Auxiliar	CEAAM OCCIDENTE
1	279	Vivian Jiménez Capuano	vjimenez	vjimenez@inamu.go.cr	5656		Prof. Especialista	AUDITORIA INTERNA
2	280	Viviana Eugenia Arguello Carvajal	varguello	varguello@inamu.go.cr	924608498		Prof. Ejecutiva	UNIDAD REGIONAL HUETAR NORTE
3	281	Wendolyn Madrigal Bejarano	wmadrigal	wmadrigal@inamu.go.cr			Prof. Operativa 2	UNIDAD REGIONAL HUETAR CARIBE
4	282	Wendy Garita Gomez	wgarita	wgarita@inamu.go.cr			Prof. Operativa 2	UNIDAD REGIONAL CENTRAL
5	283	Wendy Lilliana Lopez Castrillo	wlopez	wlopez@inamu.go.cr	8437		Prof. Ejecutiva	SERVICIOS GENERALES Y TRANSORTES
6	284	Wendy Nuñez Incera	wnunez	wnunez@inamu.go.cr			Técnica Administrativa	DELEGACION DE LA MUJER
7	285	Wendy Ward Bennett	wward	wward@inamu.go.cr	8445		Prof. Especialista	RECURSOS HUMANOS
8	286	Wendy Williams Barnett	wwilliams	wwilliams@inamu.go.cr	8643		Auxiliar	CEAAM HUETAR CARIBE
9	287	Xinia Quintero Arauz	xquintero	xquintero@inamu.go.cr	2758 7031	27587031	Prof. Ejecutiva	UNIDAD REGIONAL HUETAR CARIBE
0	288	Xinia Rodriguez Parra	xrodriguez	xrodriguez@inamu.go.cr	8645		Auxiliar	CEAAM OCCIDENTE
1	289	Xiomara Carballo Briceño	xcarballo	xcarballo@inamu.go.cr			Prof. Operativa 2	UNIDAD REGIONAL CHOROTEGA
2	290	Xiomara Castro Chaves	xcastro	xcastro@inamu.go.cr	8501		Jefa	INVESTIGACION
3	291	Yendry Porras Barrantes	yporras	yporras@inamu.go.cr	8646		Técnica Administrativa	CEAAM OCCIDENTE
4	292	Yensy Herrera Vega	yherrera	yherrera@inamu.go.cr	8474		Coordinadora	CIUDADANIA ACTIVA
5	293	Yolanda Delgado Cascante	ydelgado	ydelgado@inamu.go.cr	8403		Directora	DIRECCION TECNICA
6	294	Yorleny Calvin Zúñiga	ycalvin	ycalvin@inamu.go.cr			Prof. Operativa 2	UNIDAD REGIONAL HUETAR CARIBE
7	295	Yuliana Madrigal Salas	ymadrigal	ymadrigal@inamu.go.cr			Prof. Operativa 2	UNIDAD REGIONAL HUETAR NORTE
8	296	Zaida Barboza Hernandez	zbarboza	zbarboza@inamu.go.cr	8404		Prof. Especialista	DIRECCION ADMINISTRATIVA
9	297	Zaida Estrada Cedeño	zestrada	zestrada@inamu.go.cr	8649		Auxiliar	CEAAM HUETAR CARIBE
0	298	Zaidy Patricia Arrieta Leal	parrieta	parrieta@inamu.go.cr	8626		Prof. Ejecutiva	CEAAM METROPOLITANO
1	299	Zeidy Barrantes Salazar	zbarrantes	zbarrantes@inamu.go.cr	8650		Auxiliar	CEAAM HUETAR CARIBE
2	300	Zeidy Mata Mejía	zmata	zmata@inamu.go.cr	8576		Prof. Operativa 2	UNIDAD REGIONAL CHOROTEGA
3	301	Zelenia Rodriguez Morales	zrodriguez	zrodriguez@inamu.go.cr	8451		Jefa	DOCUMENTACION

Vulnerabilidad descrita en la sección 4.6.3.38. La transferencia de zona permite que un atacante remoto llene instantáneamente una lista de posibles objetivos de ataque. La nomenclatura tradicional puede contribuir a facilitar ese ataque (por ejemplo, proxy.example.com, nómina.example.com, b2b.example.com, etc.).

Figura 74. Lista de direcciones que contiene el sitio de la INAMU



Vulnerabilidad descrita en la sección 4.6.3.39. La versión IKEv1 soporta la autenticación de clave previamente compartida (PSK). Tal configuración podría permitir a un atacante capturar y crackear el PSK de una puerta de enlace VPN y obtener acceso no autorizado a redes privadas.

Vulnerabilidad descrita en la sección 4.6.3.41. El servidor DNS responde a las consultas de dominios de terceros que no tienen establecido el bit de recursividad. Esto puede permitir que un atacante remoto determine qué dominios se han resuelto recientemente a través de este servidor de DNS.

Vulnerabilidad descrita en la sección 4.6.3.49. Al registrarse un usuario, el sistema envía la contraseña de acceso en texto plano al correo del usuario. Al registrarse en SIDOC, el usuario y la contraseña son enviados en texto plano y al ingresar no se solicita el cambio de contraseña de manera obligatoria.

Figura 75. Información de usuario enviando en texto plano



4.7.13 Ministerio de Vivienda y Asentamientos Humanos (MIVAH)

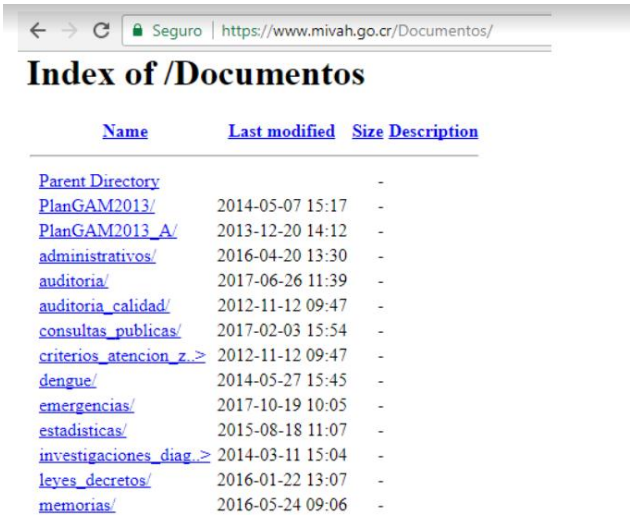
Nivel de riesgo	CVSS	Total Vulnerabilidades
Crítico	7.0-10	1
Medio	4.0-6.9	2
Bajo	0.1-3.9	1
Informativo	0.0	0

ID	VULNERABILIDAD	CVSS	NIVEL
4.6.3.11	Listado de directorios (Directory Listing)	**7.5**	Crítico
4.6.3.24	HTTPS no configurado	**5.3**	Medio
4.6.3.26	Gestión de archivos	**5.3**	Medio
4.6.3.43	Same site scripting	**3.7**	Bajo

Resumen de las vulnerabilidades encontradas

Vulnerabilidad descrita en la sección 4.6.3.11. Esta configuración permite que el servidor web pueda mostrar la lista de archivos contenidos en ese directorio y navegar a través de ellos, y pone en riesgo información sensible contenida en los archivos.

Figura 76. Listado de directorios



Refiérase a la sección 4.6.3.24 para mayores detalles del uso de certificados digitales para sitios web.

Refiérase a la sección 4.6.3.26 para mayor información relacionada a la gestión de archivos en sitios web.

Vulnerabilidad descrita en la sección 4.6.3.43. Una configuración incorrecta de DNS en el "localhost. IN A 127.0.0.1".

El detalle del ataque se puede ver en el host: **localhost.mivah.go.cr** que resuelva a 127.0.0.1.

4.7.14 MIDEPLAN

Nivel de riesgo	CVSS	Total Vulnerabilidades
Crítico	7.0-10	1
Medio	4.0-6.9	5
Bajo	0.1-3.9	0
Informativo	0.0	0

ID	VULNERABILIDAD	CVSS	NIVEL
4.6.3.11	Listado de directorios (Directory Listing)	**7.5**	Crítico
4.6.3.15	Uso de HTTP para autenticación	**6.5**	Medio
4.6.3.18	Posible exposición de información a través de WSDL	**6.5**	Medio
4.6.3.24	HTTPS no configurado	**5.3**	Medio
4.6.3.26	Gestión de archivos	**5.3**	Medio
4.6.3.27	Mensaje de error en aplicación	**5.3**	Medio

Resumen de las vulnerabilidades encontradas

Refiérase a la sección 4.6.3.11 para obtener información

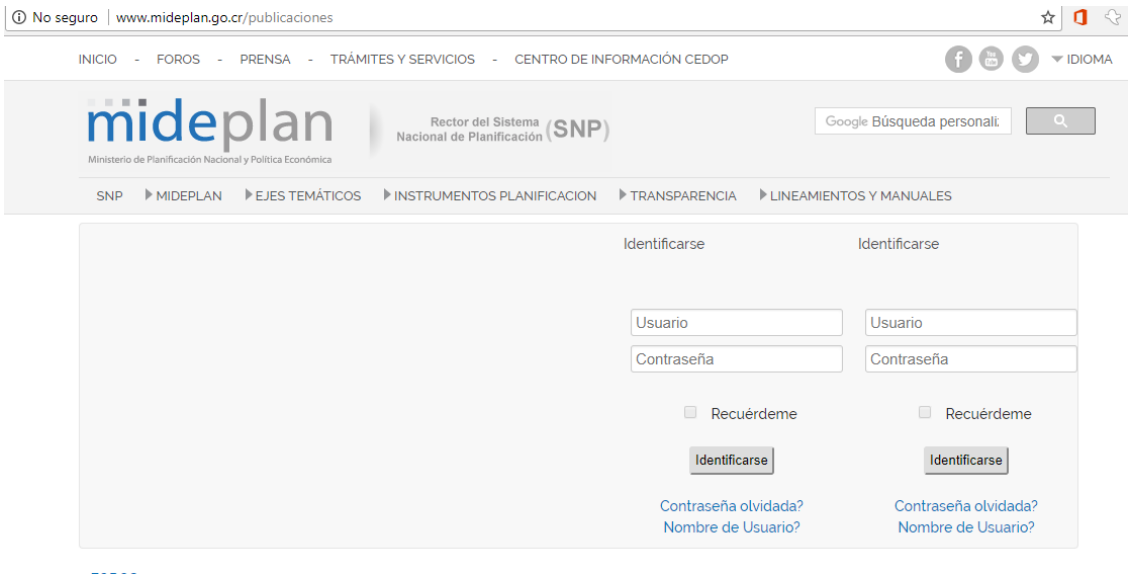
Figura 77. Listado de directorios MidePlan





Refiérase a la sección 4.6.3.15 para mayor información del uso de HTTP para autenticación.

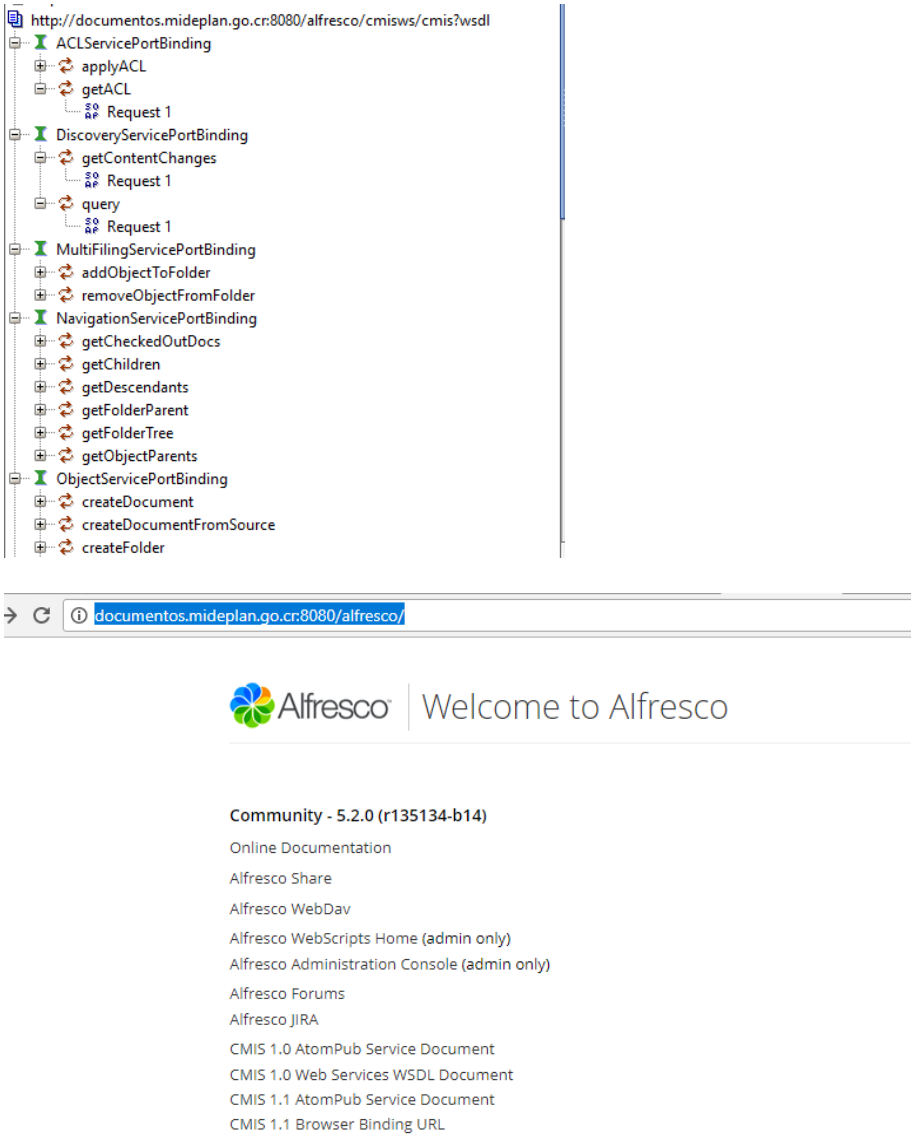
Figura 78. Ingreso al sitio Mideplan con HTTP



Vulnerabilidad descrita en la sección 4.6.3.18. El servicio web WSDL contiene información sobre los métodos / servicios a los que no se debe acceder públicamente o información sobre métodos obsoletos.

- http://documentos.mideplan.go.cr:8080/alfresco/

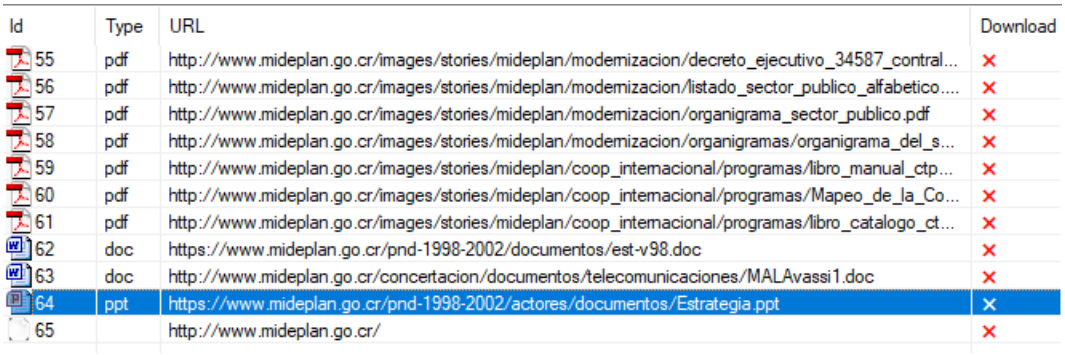
Figura 79. Exposición de WSDL



Refiérase a la sección 4.6.3.24 para mayores detalles del uso de certificados digitales para sitios web.

Refiérase a la sección 4.6.3.26 para mayor información relacionada con la gestión de archivos en sitios web.

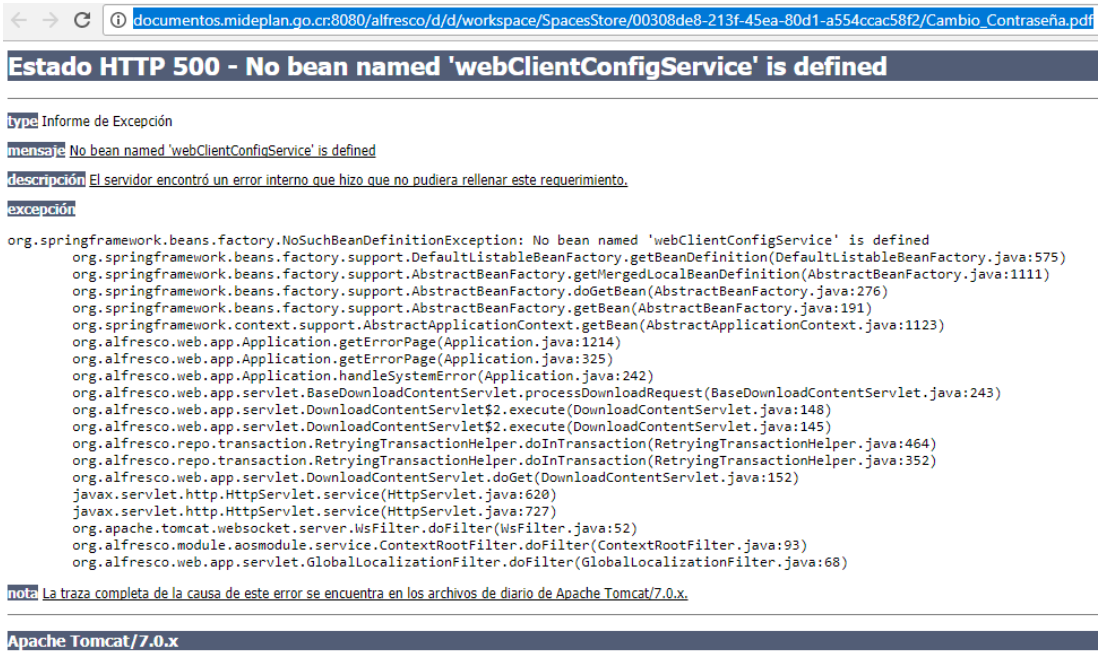
Figura 80. Archivos expuestos



Id	Type	URL	Download
55	pdf	http://www.mideplan.go.cr/images/stories/mideplan/modernizacion/decreto_ejecutivo_34587_contral...	✕
56	pdf	http://www.mideplan.go.cr/images/stories/mideplan/modernizacion/listado_sector_publico_alfabetico....	✕
57	pdf	http://www.mideplan.go.cr/images/stories/mideplan/modernizacion/organigrama_sector_publico.pdf	✕
58	pdf	http://www.mideplan.go.cr/images/stories/mideplan/modernizacion/organigramas/organigrama_del_s...	✕
59	pdf	http://www.mideplan.go.cr/images/stories/mideplan/coop_internacional/programas/libro_manual_ctp...	✕
60	pdf	http://www.mideplan.go.cr/images/stories/mideplan/coop_internacional/programas/Mapeo_de_la_Co...	✕
61	pdf	http://www.mideplan.go.cr/images/stories/mideplan/coop_internacional/programas/libro_catalogo_ct...	✕
62	doc	https://www.mideplan.go.cr/pnd-1998-2002/documentos/est-v98.doc	✕
63	doc	http://www.mideplan.go.cr/concertacion/documentos/telecomunicaciones/MALAvassi1.doc	✕
64	ppt	https://www.mideplan.go.cr/pnd-1998-2002/actores/documentos/Estrategia.ppt	✕
65		http://www.mideplan.go.cr/	✕

Vulnerabilidad descrita en la sección 4.6.3.27. Los mensajes de error o advertencia de la aplicación pueden exponer información sensible sobre el funcionamiento interno de una aplicación a un atacante.

Figura 81. Mensaje de error en la aplicación



documentos.mideplan.go.cr:8080/alfresco/d/d/workspace/SpacesStore/00308de8-213f-45ea-80d1-a554ccac58f2/Cambio_Contraseña.pdf

Estado HTTP 500 - No bean named 'webClientConfigService' is defined

type Informe de Excepción

mensaje No bean named 'webClientConfigService' is defined

descripción El servidor encontró un error interno que hizo que no pudiera rellenar este requerimiento.

excepción

```
org.springframework.beans.factory.NoSuchBeanDefinitionException: No bean named 'webClientConfigService' is defined
        org.springframework.beans.factory.support.DefaultListableBeanFactory.getBeanDefinition(DefaultListableBeanFactory.java:575)
        org.springframework.beans.factory.support.AbstractBeanFactory.getMergedLocalBeanDefinition(AbstractBeanFactory.java:1111)
        org.springframework.beans.factory.support.AbstractBeanFactory.doGetBean(AbstractBeanFactory.java:276)
        org.springframework.beans.factory.support.AbstractBeanFactory.getBean(AbstractBeanFactory.java:191)
        org.springframework.context.support.AbstractApplicationContext.getBean(AbstractApplicationContext.java:1123)
        org.alfresco.web.app.Application.getErrorPage(Application.java:1214)
        org.alfresco.web.app.Application.getErrorPage(Application.java:325)
        org.alfresco.web.app.Application.handleSystemError(Application.java:242)
        org.alfresco.web.app.servlet.BaseDownloadContentServlet.processDownloadRequest(BaseDownloadContentServlet.java:243)
        org.alfresco.web.app.servlet.DownloadContentServlet$2.execute(DownloadContentServlet.java:148)
        org.alfresco.web.app.servlet.DownloadContentServlet$2.execute(DownloadContentServlet.java:145)
        org.alfresco.repo.transaction.RetryingTransactionHelper.doInTransaction(RetryingTransactionHelper.java:464)
        org.alfresco.repo.transaction.RetryingTransactionHelper.doInTransaction(RetryingTransactionHelper.java:352)
        org.alfresco.web.app.servlet.DownloadContentServlet.doGet(DownloadContentServlet.java:152)
        javax.servlet.http.HttpServlet.service(HttpServlet.java:620)
        javax.servlet.http.HttpServlet.service(HttpServlet.java:727)
        org.apache.tomcat.websocket.server.WsFilter.doFilter(WsFilter.java:52)
        org.alfresco.module.aosmodule.service.ContextRootFilter.doFilter(ContextRootFilter.java:93)
        org.alfresco.web.app.servlet.GlobalLocalizationFilter.doFilter(GlobalLocalizationFilter.java:68)
```

note La traza completa de la causa de este error se encuentra en los archivos de diario de Apache Tomcat/7.0.x.

Apache Tomcat/7.0.x

4.7.15 Ministerio de la Presidencia

Nivel de riesgo	CVSS	Total Vulnerabilidades
Crítico	7.0-10	2
Medio	4.0-6.9	6
Bajo	0.1-3.9	0
Informativo	0.0	1

ID	VULNERABILIDAD	CVSS	NIVEL
4.6.3.8	Aplicación web desactualizada	**9.8**	Crítico
4.6.3.11	Listado de directorios (Directory Listing)	**7.5**	Crítico
4.6.3.20	SSL Anonymous Cipher Suites Supported	**5.9**	Medio
4.6.3.24	HTTPS no configurado	**5.3**	Medio
4.6.3.26	Gestión de archivos	**5.3**	Medio
4.6.3.36	SSL Medium Strength Cipher Suites Supported	**5.3**	Medio
4.6.3.37	SSL 64-bit Block Size Cipher Suites Supported (SWEET32)	**5.3**	Medio

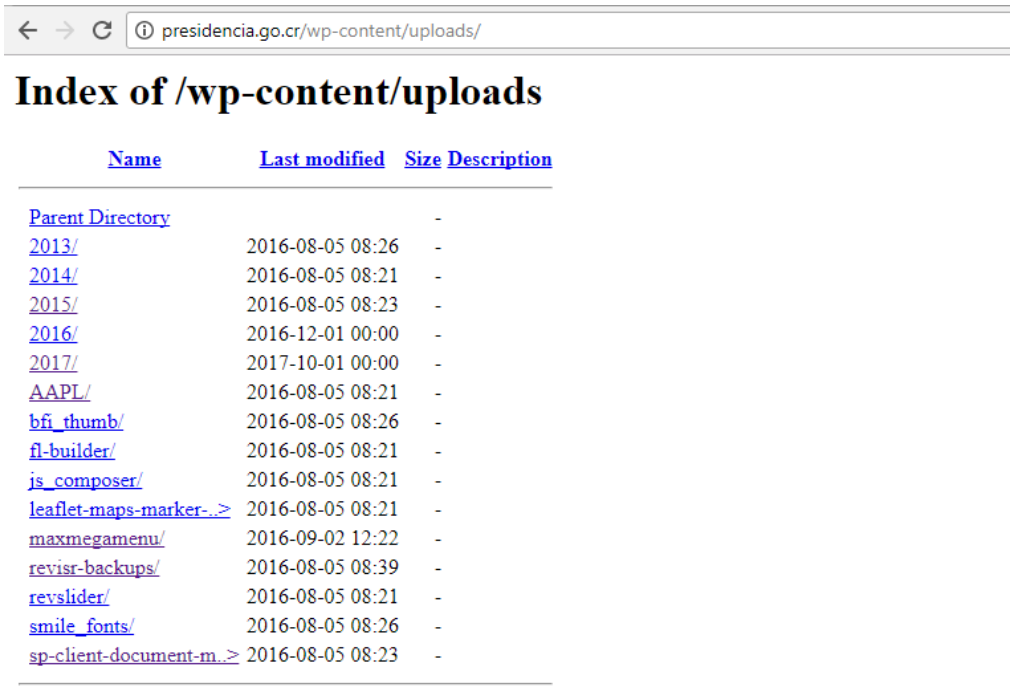
4.6.3.40	No se detecta Web Application Firewall	**5.3**	Medio
4.6.3.52	Servidor SMTP Detectado	-	Informativo

Resumen de las vulnerabilidades encontradas

Vulnerabilidad descrita en la sección 4.6.3.8. La aplicación web utilizada en el sitio está desactualizada, usa la versión 4.8.2 de WordPress que es vulnerable a SQL injection, a partir de la versión 4.8.3 se soluciona este problema.

Vulnerabilidad descrita en la sección 4.6.3.11. Esta configuración permite que el servidor web pueda mostrar la lista de archivos contenidos en ese directorio y navegar a través de ellos, lo cual pone en riesgo información sensible contenida en los archivos.

Figura 82. Listado directorio Presidencia



Vulnerabilidad descrita en la sección 4.6.3.20. El sitio permite el uso de “Anonymous SSL ciphers”.

Figura 83. Cifrados SSL

```
DH-AES128-SHA256            Kx=DH         Au=None     Enc=AES-GCM(128)         Mac=SHA256
DH-AES256-SHA384            Kx=DH         Au=None     Enc=AES-GCM(256)         Mac=SHA384
ADH-AES128-SHA              Kx=DH         Au=None     Enc=AES-CBC(128)         Mac=SHA1
ADH-AES256-SHA              Kx=DH         Au=None     Enc=AES-CBC(256)         Mac=SHA1
ADH-CAMELLIA128-SHA         Kx=DH         Au=None     Enc=Camellia-CBC(128)    Mac=SHA1
ADH-CAMELLIA256-SHA         Kx=DH         Au=None     Enc=Camellia-CBC(256)    Mac=SHA1
AECDH-AES128-SHA            Kx=ECDH       Au=None     Enc=AES-CBC(128)         Mac=SHA1
AECDH-AES256-SHA            Kx=ECDH       Au=None     Enc=AES-CBC(256)         Mac=SHA1
DH-AES128-SHA256            Kx=DH         Au=None     Enc=AES-CBC(128)         Mac=SHA256
DH-AES256-SHA256            Kx=DH         Au=None     Enc=AES-CBC(256)         Mac=SHA256
```

Vulnerabilidad descrita en la sección 4.6.3.24. El sitio web tiene habilitadas las opciones de HTTP y HTTPS, por lo que se recomienda utilizar únicamente SSL que le permita asegurar a los usuarios que se está cifrando la comunicación de la información enviada o recibida, con el fin de evitar que sea interceptada y manipulada por un atacante.

Vulnerabilidad descrita en la sección 4.6.3.26. La gestión de archivos en sitios es compleja, se podría fácilmente exponer información sensible. Debido a la naturaleza informativa y pública del sitio es comprensible, sin embargo se recomienda hacer una gestión diligente de estos archivos en todo momento.

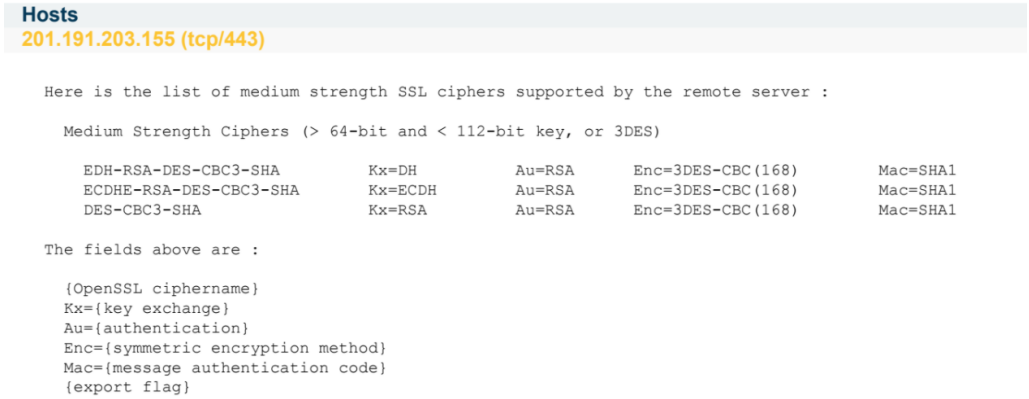
Figura 84. Gestión de archivos



Id	Type	URL
159	pdf	http://presidencia.go.cr/wp-content/uploads/2016/06/convenci_n-colectiva-1980-1982.pdf
160	pdf	http://presidencia.go.cr/wp-content/uploads/2017/01/Empleo-Pu%CC%81blico-Presentacio%CC%81n.pdf
161	pdf	http://presidencia.go.cr/wp-content/uploads/2015/02/CV-Diego-Gonz%c3%a1lez.pdf
162	pdf	http://presidencia.go.cr/wp-content/uploads/2017/03/Programa-Foro-TV-Digital-TV-Morfosis-Costa-Rica-2017.pdf
163	pdf	http://presidencia.go.cr/wp-content/uploads/2015/06/Libertad-de-expresi%c3%b3n.pdf
164	pdf	http://presidencia.go.cr/wp-content/uploads/2017/02/Afectacion-verano-2017.pdf
165	pdf	http://presidencia.go.cr/wp-content/uploads/2017/04/Presentacio%CC%81n-factores-asociados-PISA-2017.pdf

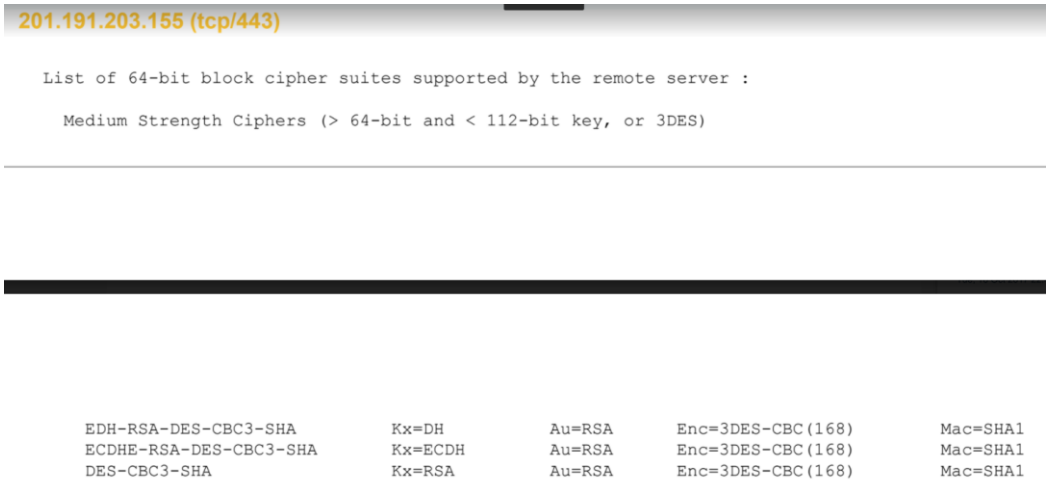
Vulnerabilidad descrita en la sección 4.6.3.36. El sitio web admite el uso de cifrado SSL que ofrece un cifrado de intensidad media. Se utiliza longitudes de clave de al menos 64 bits y menos de 112 bits o cifrado como 3DES.

Figura 85. Cifrado medio



Vulnerabilidad descrita en la sección 4.6.3.37. El sitio web admite el uso de un cifrado de bloques con bloques de 64 bits en uno o más conjuntos de cifrado.

Figura 86. Cifrado medio



Vulnerabilidad descrita en la sección 4.6.3.40. Los firewalls de aplicaciones web (WAF) proporcionan una solución efectiva para detectar amenazas, al examinar las solicitudes HTTP entrantes, incluso antes de que lleguen al servidor, para detectar y bloquear los ataques maliciosos.

Vulnerabilidad descrita en la sección 4.6.3.52. El sitio web está ejecutando un servidor de correo (SMTP) en este puerto y puede ser objetivo para los “spammers”.

4.7.16 Ministerio de Educación Pública (MEP)

Nivel de riesgo	CVSS	Total Vulnerabilidades
Crítico	7.0-10	0
Medio	4.0-6.9	6
Bajo	0.1-3.9	0
Informativo	0.0	1

ID	VULNERABILIDAD	CVSS	NIVEL
4.6.3.15	Uso de HTTP para autenticación	**6.5**	Medio
4.6.3.24	HTTPS no configurado	**5.3**	Medio
4.6.3.25	Posibilidad de listar usuarios	**5.3**	Medio
4.6.3.26	Gestión de archivos	**5.3**	Medio
4.6.3.33	Archivo DS_Store en el servidor web	**5.3**	Medio
4.6.3.40	No se detecta Web Application Firewall (WAF)	**5.3**	Medio
4.6.3.58	Archivos predeterminados configurados en el sistema	-	Informativo

Resumen de las vulnerabilidades encontradas

Refiérase a la sección 4.6.3.15 para mayor información del uso de HTTP para autenticación.

- http://www.mep.go.cr/user

Figura 87. Uso de HTTP para acceso al MEP



Refiérase a la sección 4.6.3.24 para mayores detalles del uso de certificados digitales para sitios web.

Vulnerabilidad descrita en la sección 4.6.3.25. La posibilidad de listar usuarios al intentar ingresar, para verificar su validez, es un tipo de ataque en que el script de validación de la página web muestra al atacante si el nombre de usuario proporcionado es correcto o no.

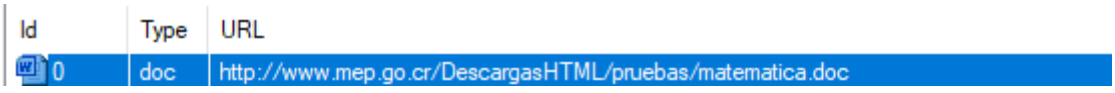
Figura 88. Puede validar usuarios disponibles en MEP



Refiérase a la sección 4.6.3.26 para mayor información relacionada con la gestión de archivos en sitios web.

- http://www.mep.go.cr/DescargasHTML/pruebas/estudios_sociales_solucionario.doc
- http://www.mep.go.cr/DescargasHTML/pruebas/civica_solucionario.docx

Figura 89. Archivos expuestos



- http://www.mep.go.cr/DescargasHTML/pruebas/matematica_solucionario.docx
- http://www.mep.go.cr/DescargasHTML/pruebas/frances_solucionario.docx

Vulnerabilidad descrita en la sección 4.6.3.33. Estos archivos incluyen en algunas ocasiones información sobre la configuración del sistema. Al ser un servidor web, los archivos podrán ser descargados por los usuarios de la página.

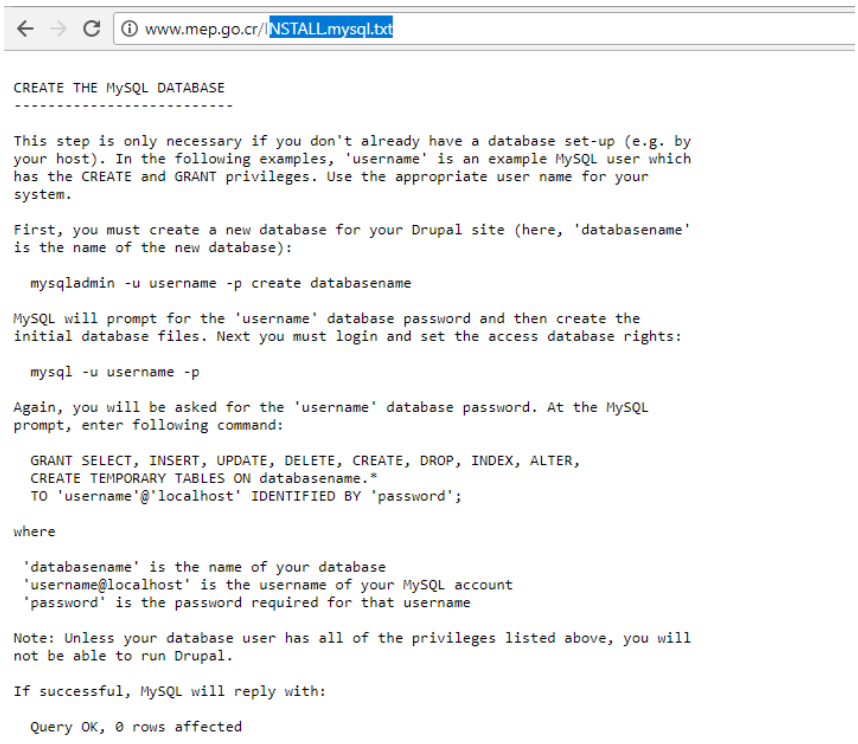
- http://www.mep.go.cr/.DS_Store
- http://www.mep.go.cr/sites/.DS_Store
- http://www.mep.go.cr/sites/default/files/recursos/recursos-interactivos/cyberlab_kids/.DS_Store

Refiérase a la sección 4.6.3.40 para obtener información acerca de lo que es un WAF y la importancia de implementarlo.

Vulnerabilidad descrita en la sección 4.6.3.58. La página web posee los archivos de configuración de Drupal por defecto visibles, archivos como install.php, CHANGELOG.txt, entre otros.

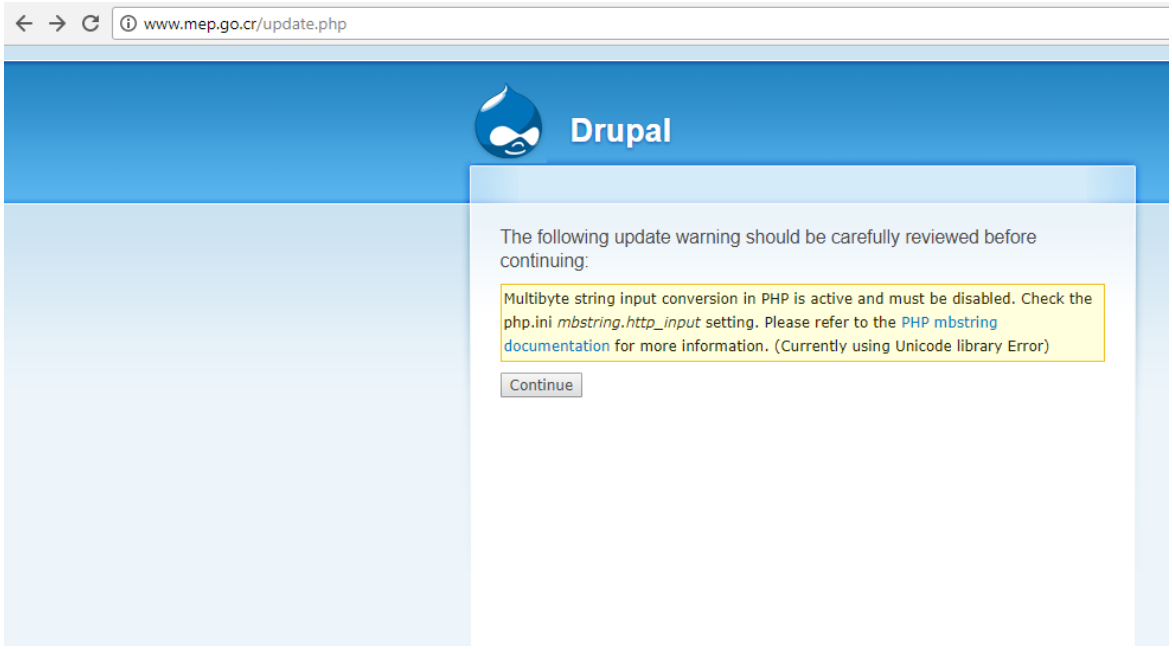
- http://www.mep.go.cr/INSTALL.mysql.txt

Figura 90. Archivos de configuración visibles en MEP



- http://www.mep.go.cr/update.php

Figura 91. Mensaje de actualización de Drupal



- http://www.mep.go.cr/CHANGELOG.txt

Figura 92. Changelog



4.8 Informe completo de las vulnerabilidades encontradas

En la siguiente tabla se muestra un resumen de todas las vulnerabilidades encontradas en cada uno de los ministerios, la cual facilita la visión general del estado actual a la fecha en la que se realizaron las pruebas.

ID	VULNERABILIDAD	CVSSv3	Cantidad
4.6.3.1	Inyección de código SQL	**10.0**	**1**
4.6.3.2	ISC BIND < 4.9.8 / 8.2.3 Multiple Remote Overflows	**10.0**	**1**
4.6.3.3	ISC BIND < 4.9.5 DNS Resolver Functions Remote Overflow	**10.0**	**1**
4.6.3.4	Múltiples vulnerabilidades en el servidor de Oracle GlassFish	**10.0**	**1**
4.6.3.5	ISC BIND < 4.9.7-REL / 8.2.2-P5 Multiple Remote Vulnerabilities	**10.0**	**1**
4.6.3.6	ISC BIND < 4.9.7 / 8.1.2 Inverse-Query Remote Overflow	**10.0**	**1**
4.6.3.7	OpenSSL 'ChangeCipherSpec' MiTM Vulnerability	**9.8**	**1**
4.6.3.8	Aplicación Web desactualizada	**9.8**	**5**

4.6.3.9	Sitio Web hackeado	**9.8**	**1**
4.6.3.10	Divulgación de información en OpenSSL Heartbeat	**9.4**	**1**
4.6.3.11	Listado de directorios (Directory Listing)	**7.5**	**4**
4.6.3.12	Métodos inseguros habilitados	**7.5**	**9**
4.6.3.13	Ataque de denegación de servicio XML cuadrático	**7.5**	**1**
4.6.3.14	SSL Protocolo versión 2 y 3 detectado	**7.5**	**4**
4.6.3.15	Uso de HTTP para autenticación	**6.5**	**12**
4.6.3.16	Puerto RDP 3389 abierto al público	**6.5**	**2**
4.6.3.17	Archivo inseguro crossdomain.xml	**6.5**	**1**
4.6.3.18	Posible exposición de información a través de WSDL	**6.5**	**1**
4.6.3.19	Biblioteca de JavaScript vulnerable	**6.5**	**1**
4.6.3.20	SSL Anonymous Cipher Suites Supported	**5.9**	**1**
4.6.3.21	DNS Server Spoofed Request Amplification DDoS	**5.8**	**3**
4.6.3.22	DNS Server Recursive Query Cache Poisoning Weakness	**5.8**	**1**
4.6.3.23	Microsoft Windows Remote Desktop Protocol	**5.6**	**1**

	Server Man-in-the-Middle Weakness		
4.6.3.24	HTTPS no configurado	**5.3**	**15**
4.6.3.25	Posibilidad de listar usuarios	**5.3**	**2**
4.6.3.26	Gestión de archivos	**5.3**	**14**
4.6.3.27	Mensaje de error en aplicación	**5.3**	**4**
4.6.3.28	Páginas de pruebas encontradas en sitios de producción	**5.3**	**4**
4.6.3.29	Divulgación de la versión ASP.NET	**5.3**	**1**
4.6.3.30	Exposición de XAMPP	**5.3**	**1**
4.6.3.31	Ataque remoto de DDoS por la versión del sistema operativo	**5.3**	**2**
4.6.3.32	El análisis en algunos hilos particulares de PHP permite DDoS	**5.3**	**1**
4.6.3.33	Archivo DS_Store en el servidor web	**5.3**	**1**
4.6.3.34	Posible exposición de directorios sensibles	**5.3**	**1**
4.6.3.35	Exposición de información sensible	**5.3**	**5**
4.6.3.36	SSL Medium Strength Cipher Suites Supported	**5.3**	**4**

4.6.3.37	SSL 64-bit Block Size Cipher Suites Supported (SWEET32)	**5.3**	**2**
4.6.3.38	Divulgación de información de transferencia de zona del servidor DNS (AXFR)	**5.3**	**1**
4.6.3.39	Internet Key Exchange (IKE) Aggressive Mode with Pre-Shared Key	**5.3**	**1**
4.6.3.40	No se detecta Web Application Firewall (WAF)	**5.3**	**11**
4.6.3.41	Caché del servidor DNS divulga información de forma remota	**5.2**	**2**
4.6.3.42	Nivel de cifrado de Terminal Services medio o bajo	**3.7**	**1**
4.6.3.43	Same site scripting	**3.7**	**1**
4.6.3.44	Soporta SSL RC4 Cipher Suites	**3.7**	**1**
4.6.3.45	Inicios de sesión de POP3 sin cifrar permitidos	**3.7**	**1**
4.6.3.46	Carga de archivos sin validación	**-**	**3**
4.6.3.47	Referencias a un sitio web en lista negra	**-**	**1**
4.6.3.48	Versión de Microsoft IIS expuesta	**-**	**3**
4.6.3.49	Envío de contraseñas en texto plano al correo electrónico	**-**	**1**

4.6.3.50	Servidor FTP Detectado	-	**3**
4.6.3.51	VERITAS NetBackup Agent Detection	-	**1**
4.6.3.52	Servidor SMTP Detectado	-	**2**
4.6.3.53	Microsoft SQL Server TCP/IP Listener Detection	-	**2**
4.6.3.54	Microsoft SQL Server UDP Query Remote Version Disclosure	-	**2**
4.6.3.55	Detección de servidor PostgreSQL	-	**2**
4.6.3.56	Detección del servidor POP	-	**1**
4.6.3.57	MySQL Server Detection	-	**1**
4.6.3.58	Archivos predeterminados configurados en el sistema	-	**1**

CAPÍTULO V

CONCLUSIONES

La evaluación de vulnerabilidades es un proceso necesario en toda empresa independientemente del área en la que se desenvuelva. El Gobierno Central de Costa Rica no es la excepción, por lo tanto, los ministerios deben enfocar esfuerzos en poder garantizar a los usuarios de los sitios web la seguridad de sus datos, mediante este tipo de prácticas con herramientas especializadas y expertos en la evaluación de vulnerabilidades.

Tras el análisis de la estructura de los sitios web se determina que la mayoría de las páginas de los ministerios poseen una estructura que puede considerarse como compleja, debido a la cantidad de distintas rutas o subpáginas que tienen de contenido, así como la cantidad de archivos de diferentes fechas que se encuentran disponibles para ser descargados por los usuarios. Mediante el uso de herramientas tecnológicas se logra identificar la posible ubicación física de los servidores en donde se encuentran alojadas las páginas web analizadas, así como la tecnología utilizada en el desarrollo de estas. Se determina que solo una de las páginas analizadas se encuentra alojada en Estados Unidos.

Con la colaboración del MICITT se determina la estructura, composición y personal a cargo que brinda el soporte a cada sitio web de los ministerios del gobierno central, y se concluye que cada ministerio posee un equipo encargado, los cuales son el punto de contacto para cualquier tema relacionado con la seguridad de la página web analizada.

Mediante una evaluación y asesoría por parte del Líder del Capítulo de OWASP para Costa Rica, se determina que la metodología que mejor se adapta a las necesidades de esta evaluación es el OWASP Testing Guide V4.

Las herramientas utilizadas para realizar la evaluación fueron seleccionadas de acuerdo con la experiencia y manejo de los expertos, tanto por el uso y aprendizaje durante la carrera de la

maestría, como también herramientas que son de uso diario por los expertos. Se utilizaron las que se adaptaban de mejor forma a esta la evaluación de vulnerabilidades.

Se realizaron las pruebas de penetración y detección de vulnerabilidades en las páginas web, y se detecta una cantidad considerable de vulnerabilidades, las cuales pasan a ser depuradas para remover falsos positivos. Se analizaron las vulnerabilidades detectadas, se depuraron para eliminar los falsos positivos y se confirma la cantidad y severidad de las vulnerabilidades encontradas. Seguidamente, se categoriza cada una de ellas de acuerdo con el CVSS.

Se compararon las vulnerabilidades finales para dar recomendaciones con criterio de experto en la severidad de la vulnerabilidad, así como recomendaciones para que cada una de estas vulnerabilidades pueda ser arreglada por parte de los encargados.

Tras la determinación de las vulnerabilidades existentes, se llevó a cabo un informe dirigido al MICITT para su debida atención y seguimiento con los encargados de las páginas web según cada ministerio.

RECOMENDACIONES

Se recomienda a los ministerios realizar periódicamente un análisis de vulnerabilidades en las páginas web, esto para poder determinar de forma proactiva amenazas que se puedan materializar. De esta forma se puede garantizar el correcto funcionamiento de la página web a los usuarios, así como la seguridad de los datos personales de los usuarios de la página web.

Se recomienda implementar protección a nivel de capa 7, un Web Application Firewall (WAF) para poder detectar escaneos en los sitios web, poder bloquear automáticamente vulnerabilidades como SQL injection, Cross Site Scripting y bloqueos automáticos de técnicas utilizadas por atacantes para vulnerar los sistemas. Con un WAF los encargados de las páginas podrían generar informes de las regiones que más intentos de hackeo registran en las páginas web, así como las direcciones IP más insistentes y las horas o días en las que las aplicaciones sufren más intentos de hackeo por parte de personas maliciosas o robots programados para este tipo de comportamientos.

Se recomienda implementar protección capa 3-4 contra ataques de denegación de servicio (DOS), para poder mitigar los ataques sin afectar el rendimiento de la página y sin afectar a los usuarios finales.

Las páginas web analizadas cuentan con muchos recursos y diferentes subpáginas asociadas, además de esto poseen muchos archivos para ser descargados, algunas de ellas poseen más de 300 archivos para ser descargados por los usuarios. Por lo anterior, se recomienda analizar la información expuesta así como la cantidad de componentes de la página. Este tipo de páginas tan grandes poseen un riesgo, el cual es tener poco control de todo lo que posee la página. Esto a nivel de mantenimiento en el tiempo se vuelve complejo, por lo que puede provocar que exista información sensible de forma pública que no debería ser expuesta.

Para un futuro análisis del sitio y la aplicación web es recomendable utilizar la metodología OWASP específicamente el Testing Guide V4 utilizado para este proyecto, y de ser posible extenderlo a todas las páginas web y servidores relacionados con cada Ministerio.

Se recomienda a los ministerios atender las vulnerabilidades reportadas en cada página web, especialmente dar prioridad a las vulnerabilidades críticas reportadas. Al implementar las soluciones de las vulnerabilidades, especialmente las críticas, se recomienda revisar todos aquellos otros componentes de la página web donde la vulnerabilidad se pueda presentar.

Se recomienda deshabilitar el método de solicitud TRACE, si no son necesarios para el funcionamiento, en servidores web de producción y desarrollo. Los fabricantes de servidores web y navegadores deben informar a sus clientes cómo deshabilitar o no permitir el método TRACE.

BIBLIOGRAFÍA

Acunetix. (2017). *Is your website hackable?* Obtenido de Acunetix: https://www.acunetix.com/resources/acunetix-brochure.pdf

Akamai. (s.f.). *Gestión de vulnerabilidades*. Recuperado el 2017, de Akamai: https://www.akamai.com/es/es/resources/vulnerability-management.jsp

Burns, J., P., R., S., R., Rusenbluth, D., & Surendran, A. (s.f.). *Automatic Management of Network Security Policy.* Boston: Boston University.

Dragon Jar. (1 de 9 de 2017). *Dragon Jar*. Obtenido de Como realizar un pentest: https://www.dragonjar.org/como-realizar-un-pentest.xhtml

Eleven Paths. (2017). *Foca*. Obtenido de https://www.elevenpaths.com/es/labstools/foca-2/index.html

Franco, D. (2013). Herramienta para la detección de vulnerabilidades. *Información Tecnológica*.

Gómez., V. (2000). Auditoría de seguridad informática. (RA-MA, Ed.) Obtenido de Ebrary: http://www.ebrary.com

Google. (2017). *Más información sobre sitemaps*. Obtenido de https://support.google.com/webmasters/answer/156184?hl=es

Grangeia, L. (Febrero de 2004). DNS Cache Snooping. Obtenido de http://cs.unc.edu/~fabian/course_papers/cache_snooping.pdf

Grossman, J. (2003). Cross-Site Tracing. Obtenido de http://www.cgisecurity.com/whitehat-mirror/WH-WhitePaper_XST_ebook.pdf

Hess Araya, C. (Noviembre de 2015). *Normativa de derecho Informático*. Obtenido de Blog: http://hess-cr.blogspot.com/p/normativa-de-derecho-informatico.html

Himanshu, K. (2014). *Learning Nessus for Penetration Testing.* (P. P. Ltd, Ed.) Birmingham, GBR. Obtenido de Ebrary

IETF. (1997). RFC 2109. Obtenido de https://www.ietf.org/rfc/rfc2109.txt

Imperial Violet. (2014). *Early ChangeCipherSpec Attack*. Obtenido de https://www.imperialviolet.org/2014/06/05/earlyccs.html

Imperva. (Marzo de 2015). Attacking SSL when using RC4. *Hacker Intelligence Initiative*. Obtenido de http://www.imperva.com/docs/HII_Attacking_SSL_when_using_RC4.pdf

Matt Tesauro. (5 de 8 de 2014). *Testing Introduction and Objectives*. Obtenido de OWASP: https://www.owasp.org/index.php/Testing:_Introduction_and_objectives

Mendoza, M. (s.f.). *Cómo Utilizar OpenVas para evaluación de vulnerabilidades*. Obtenido de welivesecurity: https://www.welivesecurity.com/la-es/2014/11/18/como-utilizar-openvas-evaluacion-vulnerabilidades/

MICITT. (s.f.). Recuperado en octubre de 2017, de Ley de Promoción del Desarrollo Científico y Tecnológico: https://www.micit.go.cr/index.php?option=com_content&view=article&id=503&Itemid=624

Mitre Corporation. (2017). *CVE Details Joomla*. Obtenido de https://www.cvedetails.com/vulnerability-list/vendor_id-3496/product_id-16499/version_id-145393/Joomla-Joomla--2.5.9.html

NMAP ORG. (2017). *NMAP*. Obtenido de https://nmap.org/

Opens SSL. (Junio de 2014). Open SSL security advisory. *SSL/TLS MITM vulnerability*. Obtenido de https://www.openssl.org/news/secadv/20140605.txt

Pentest Geek. (2017). *What is BurpSuite*. Obtenido de https://www.pentestgeek.com/what-is-burpsuite

Perea, J., Puello, P., & Franco, D. (2012). *Metodología para la detección de vulnerabilidades en redes de datos.*

PortsWigger. (2017). *Web Security*. Obtenido de Ports Wigger: https://portswigger.net/burp

Procuradoría General de la República. (1 de 9 de 2017). *PGR*. Obtenido de Sistema Costarricense de Información Jurídica: http://www.pgrweb.go.cr

Reyes Roig, D. (2010). *Guía de Implementación de la seguridad en redes de Núcleo mpls.* (CUJAE, Ed.) Cuba: Instituto Superior Politécnico José Antonio Echeverría. Obtenido de ProQuest ebrary

Sabino, C. (1992). El Proceso de Investigación. En C. Sabino, *El Proceso de Investigación* (pág. 216). Caracas: Panapo.

Siddharth, S., & Doshi, P. (2006). *Five common web application vulnerabilities*. Obtenido de Symantec: https://www.symantec.com/connect/articles/five-common-web-application-vulnerabilities

SmartBear. (2017). *SOAP UI*. Obtenido de https://www.soapui.org/

Tenable. (2017). Nessus. Obtenido de https://www.tenable.com/products/nessus-vulnerability-scanner/nessus-professional

The Mitre. (2017). *Common Weakness Enumeration*. Obtenido de Mitre: https://cwe.mitre.org/data/definitions/651.html

Thumann, M., & Enno, R. (2017). *PSK Cracking using IKE Aggresive Mode*. Obtenido de ERNW: https://www.ernw.de/download/pskattack.pdf

Weidman, G. (2014). *Penetration Testing A Hands-On Introduction to Hacking.* (P. ebrary, Ed.) CA.

Wesemann, D. (Enero de 2009). DNS queries for. Obtenido de https://isc.sans.edu/diary/DNS+queries+for+/5713

WSDL. (2017). *WSDL Disclosure*. Obtenido de ws-attacks: http://www.ws-attacks.org/WSDL_Disclosure

GLOSARIO

API: Conjunto de funciones y procedimientos que cumplen una o muchas más funciones con el fin de ser utilizadas por otro software. Del inglés Application Programming Interface.

Ciberdelincuente: Se conoce como ciberdelincuentes a las personas que ejecutan actividades delictivas a través de internet, tales como robo de información, acceso a redes privadas, estafas, y todo lo que tiene que ver con los delitos e ilegalidad utilizando la red de internet.

Cookie: Nombre que recibe el almacenamiento temporal de información que usan páginas de Internet donde se puede consultar actividad previa del usuario.

Cross Site Scripting(XSS): Es un ataque de inyección de código malicioso para su ejecución posterior, se puede realizar en sitios web y aplicaciones locales. Es causado por usuarios mal intencionados al enviar código malicioso a la aplicación web, que se coloca en forma de un hipervínculo para que así al cargar la página o el link se podría enviar al usuario a otra página web, o se podría enviar la sesión del usuario a otra página, cargar un sitio de robo de credenciales o ejecutar comandos que pueden dañar el sistema.

Cross Site Tracing XST: Descubierto por el investigador de seguridad Web Jeremiah Grossman en 2003, permite a un atacante conseguir acceso a las cookies e información de credenciales de autenticación de un individuo al explotar controles ActiveX, Flash, Java y otros que permiten la ejecución de una llamada HTTP TRACE.

DNS: (Domain Name System / Sistema de Nombre de Dominio) Se refiere a la asignación de un nombre a una dirección IP, para así simplificar el modo de acceder a un servidor.

El ataque de envenenamiento de caché de servidor DNS se da al vulnerarse la tabla de nombres de dominio de un servidor en internet, por medio de código malicioso. El resultado de esto es que algunas o todas las direcciones de Internet que conforman esta tabla son reemplazadas por

direcciones falsas. Así que, en lugar de dirigir al sitio web legítimo, las solicitudes remiten a los usuarios a páginas falsas.

Envenenamiento: El ataque de envenenamiento de caché de servidor DNS se da al vulnerarse la tabla de nombres de dominio de un servidor en internet, por medio de código malicioso. El resultado de esto es que algunas o todas las direcciones de Internet que conforman esta tabla son reemplazadas por direcciones falsas. Así que, en lugar de dirigir al sitio web legítimo, las solicitudes remiten a los usuarios a páginas falsas.

Escaneo de vulnerabilidades: Análisis, identificación y reporte de vulnerabilidades relacionadas con la seguridad que se tiene en una infraestructura, página o programa de cómputo.

Ethical hacking: Utilizar técnicas de ataque para encontrar fallas de seguridad con el consentimiento de la empresa.

Hacker: Según la RAE puede ser un pirata informático o una persona experta en el manejo de computadoras, que se ocupa de la seguridad de los sistemas y de desarrollar técnicas de mejora.

Hosts: Se utiliza para poder referirse a computadoras u otros dispositivos conectados a una red que proveen y utilizan servicios de ella. En general, son computadores monousuario o multiusuario que ofrecen servicios de transferencia de archivos, conexión remota, servidores de base de datos, servidores web, etc.

Malware: La palabra malware viene del inglés, resulta de la unión de las palabras “Malicious Software” o programa malicioso, que tiene como función dañar un sistema o causar un mal funcionamiento. Por lo tanto, engloba un gran número de amenazas informáticas.

Man in the Middle attack: En español conocido como ataque de hombre en el medio, es un método de ataque en el que se introduce un intermediario (el cibercriminal o una herramienta maliciosa) entre la víctima y la fuente: una página de banca en línea o una cuenta de correo

electrónico. Este método solo necesita que el atacante se sitúe entre las dos partes que intentan comunicarse; interceptando los mensajes enviados y pasando desapercibido.

Open Source: O código abierto, es el software cuyo código fuente y otros derechos normalmente exclusivos para quienes poseen los derechos de autor, se publican bajo licencia de código abierto o son parte del dominio público.

POP3: Es un protocolo utilizado por los clientes de correo electrónico para poder recibir y descargar los mensajes que se están almacenados en un servidor remoto.

Pruebas de penetración: Pruebas para identificar las fallas de seguridad que puede tener un sistema, utiliza las mismas herramientas y procesos que realizaría un atacante para tener acceso a su organización, pero en un entorno totalmente controlado.

Querystring: Es la parte de una URL que contiene los datos que deben pasar a aplicaciones web como los programas.

REST: Deriva de "REpresentational State Transfer", o "transferencia de representación de estado". Un servicio REST no tiene estado, por lo que entre dos llamadas cualesquiera, el servicio pierde todos sus datos. Esto es, que no se puede llamar a un servicio REST pasándole unos datos (ejemplo un usuario y contraseña) esperando que "nos recuerde" en la siguiente petición.

SMTP: Proviene de Simple Mail Transfer Protocol (Protocolo de Transferencia de Correo Simple), es el procedimiento que permite el transporte del email en la Internet.

Sniffing: Técnica que consiste en "escuchar" o capturar toda la información que circula por una red.

SOAP: Protocolo que proporciona un mecanismo simple y ligero de intercambio de información entre dos puntos usando el lenguaje XML.

SSL/TLS: Protocolos criptográficos que proporcionan autenticación y cifrado de la información entre computadoras, servidores y aplicaciones que se encuentran en una red.

Stunnel: Programa Open-source utilizado para crear túneles TLS/SSL, para poder proveer conexiones cifradas seguras para clientes o servidores que no utilizan TLS o SSL de forma nativa.

TICs: Tecnologías de información y comunicación.

WSDL: Web Services Description Language, es un protocolo basado en XML, que describe los accesos al servicio web.

ANEXOS

Anexo 1

¿Nos puede indicar cuál es su rol dentro de OWASP?

Mi rol fundamental es de líder del capítulo local de OWASP Costa Rica, es decir tener una representación local de la fundación OWASP en el país para crear un ecosistema donde se fomenten las mejores prácticas del desarrollo seguro de software.

Como líder del capítulo son varias las actividades que se realizan con el único objetivo de hacer visible la seguridad del software, dentro de estas actividades se enumeran:

Planear y organizar eventos y conferencias abiertas al público en general donde se comparta conocimiento, mejores prácticas, nuevos vectores de ataques, guías y estándares de la industria, herramientas y código fuente para mejorar la seguridad de las aplicaciones.

Fomentar los valores de OWASP en la región para crear una cultura de sensibilización sobre los riesgos que afectan nuestra infraestructura.

Promover en el país y la región Latinoamericana las mejores prácticas del desarrollo seguro para facultar a las empresas y organizaciones a desarrollar aplicaciones que sean resistentes a ataques informáticos actuales.

Crear un ecosistema donde se pueda compartir conocimiento sobre ataques informáticos y sus métodos de prevención y detección.

Organizar talleres prácticos donde se pueda contextualizar, desde un enfoque teórico-práctico, los ataques informáticos más difundidos de forma global y proponer enfoques inteligentes para contrarrestar estas amenazas cibernéticas.

Promover las bondades que ofrece la comunidad de código abierto (del inglés Open Source) de forma que tanto el sector público como privado se puedan beneficiar con un enfoque educativo.

¿Desde hace cuánto apoya en ese rol a OWASP?

Mi travesía inicia en el año 2011 cuando trabajaba como desarrollador de aplicaciones para una empresa internacional desarrollando aplicaciones bancarias para instituciones financieras basadas en Estados Unidos.

El sector de banca y finanzas, debido a la democratización de los ataques informáticos ha sido tradicionalmente blanco de los atacantes por motivaciones puramente financieras por lo cual, al desarrollar aplicaciones públicamente expuestas, es necesario que la barra de la seguridad de Aplicaciones esté lo suficientemente alta de forma que los atacantes no puedan comprometer información sensible o cualquier otro efecto adverso producto de un ataque informático.

Habiendo desarrollado aplicaciones de software durante muchos años, conocer sobre OWASP y sobre el contexto de colaboración que existe detrás me motivó a crear el capítulo en el país y poder generar un poco de conciencia sobre estos temas. Los ataques informáticos ciertamente han evolucionado desde entonces, sin embargo, los esfuerzos por generar conocimiento en el país han ido dando muy buenos resultados.

Precisamente me puse en contacto con la comunidad de OWASP y tuve todo el apoyo necesario para echar andar el capítulo desde sus inicios.

¿Desde hace cuánto tiempo aproximadamente existe el capítulo de OWASP para Costa Rica?

Se podría decir que aproximadamente seis años desde su fundación en el 2011.

¿Cuál cree usted que es la mayor ventaja de utilizar las buenas prácticas de OWASP?

OWASP es una de las organizaciones más importantes y un referente sobre las mejores prácticas del desarrollo seguro de software. La industria informática ciertamente se ha visto

beneficiada con los aportes que la comunidad open source ha puesto a disposición de la industria con el único objetivo de hacer visibles las causas que hacen que el software sea inseguro.

Además, en OWASP participan muchas de las mentes más brillantes –hackers extraordinarios- cuyo aporte en la generación de contenido, guías y herramientas es un elemento fundamental del éxito de la organización.

El contenido de las guías y estándares es avalado por los actores más importantes de la industria, las revisiones periódicas, generación de *releases* y los proyectos son un caso de estudio claro del éxito que se puede alcanzar cuando las personas trabajan en colaboración.

Otro aspecto importante es que el contenido (guías, hojas de trucos, herramientas, código fuente, etc.) ha sido desarrollado por personas con muchos años de experiencia lidiando con problemas de Seguridad y que proporcionan un enfoque holístico para remediar estos riesgos.

¿Cuál cree usted que es la mayor desventaja de utilizar las buenas prácticas de OWASP?

En mi experiencia no he podido encontrar ninguna desventaja de utilizar las mejores prácticas de OWASP, más bien considerando la calidad del contenido es un valor agregado para cualquier empresa poder contar con todo el conocimiento disponible para que sea aprovechado por los desarrolladores.

¿Cuál cree usted es la justificación principal para usar la metodología del Testing Guide OWASP para análisis y detección de vulnerabilidades en páginas WEB, sobre otras técnicas conocidas en la industria?

La guía de pruebas conocida como Testing Guide ha sido el esfuerzo y la combinación durante muchos años por expertos en el campo de la seguridad informática. Es una de las guías más

completas que existe para comprender los riesgos existentes y los diversos ataques informáticos que pude sufrir una aplicación.

Los autores de esta guía son líderes de renombre en la industria los cuales comprenden los retos actuales en el momento de desarrollar aplicaciones de software y al tener un panorama claro sobre superficies de ataque, riesgo, validación de datos de entrada, arquitecturas entre otros, esta guía está diseñada precisamente para realizar pruebas de seguridad de forma tal que se puedan encontrar, de forma temprana en el ciclo de vida del desarrollo de software, vulnerabilidades potenciales.

Considerando entonces las personas detrás de esta guía en conjunto con la premisa de que no es posible tener aplicaciones de software seguras sin que exista un mecanismo para realizar pruebas de seguridad (security testing) y el enfoque que presenta esta guía de OWASP en mi opinión este es un activo y una ventaja competitiva que se puede aprovechar por las organizaciones.

¿Conoce el nivel de impacto que ha generado en las organizaciones que implementan esta metodología?

Desafortunadamente no creo que exista una métrica o estadística públicamente disponible para saber el grado de penetración o los resultados obtenidos por las organizaciones, de hecho, sería sustancialmente bueno que producto de la investigación que ustedes desarrollan se pueda tener un caso de estudio en el país.

Sin embargo, en mi experiencia desarrollando aplicaciones de software he tenido el privilegio de encontrar muchos problemas de seguridad, utilizando esta guía, y desarrollando casos de pruebas basados en esta guía, en una etapa muy temprana donde se han podido resolver estas vulnerabilidades de forma efectiva.

Un aspecto fundamental es que este año 2017, en colaboración con el líder del capítulo de OWASP Perú – John Varga – pudimos desarrollar un taller en el Congreso Mundial de Testing denominado 7th World Congress for Software Quality donde compartimos con muchas personas de distintas partes del mundo sobre la importancia de incluir esta guía durante la etapa de pruebas del software y así incluir la seguridad como un aspecto relevante del proceso de calidad.

¿Cómo cree usted que OWASP contribuye a la comunidad?

El renovado autor estadounidense Clay Shirky, en una conferencia - que en mi criterio todas las personas que desarrollan aplicaciones de software deben escuchar de forma casi obligatoria – y que lleva por título Love, Internet Style 1 pone en perspectiva lo importante que es realizar contribuciones y apoyar las comunidades de código abierto.

Cuando yo visualizo los aportes que OWASP ha brindado a la industria informática per se, se vienen a mi mente los miles de contribuciones de personas, que, de forma desinteresada, se juntan y dedican tiempo y esfuerzo con la única motivación de hacer de Internet un lugar más seguro y por consiguiente que el proceso de desarrollo de aplicaciones – una de las industrias más consolidadas – contemple la seguridad desde cada una de las etapas y de forma continua y repetitiva.

Donaciones de contenido, herramientas, guías, hojas de trucos y código fuente son solo algunos ejemplos de las contribuciones de OWASP para con la comunidad sin dejar de lado los eventos globales enfocados en crear conciencia y compartir conocimiento.

Así mismo, las mejores prácticas de los actores tecnológicos (Google, Microsoft, Apple, Oracle, IBM entre otros) han adoptado las recomendaciones de OWASP y de alguna u otra

[1] https://www.youtube.com/watch?v=Xe1TZaElTAs

manera los productos que consumimos los usuarios finales – en modo de aplicaciones de uso diario – indudablemente han considerado uno que otro aspecto creado por OWASP y sus miembros.

¿Qué herramientas posee OWASP para apoyar una labor de análisis de vulnerabilidades en páginas web?

Talvez la forma apropiada no es visualizarlo como herramientas si no como proyectos los cuales están enfocados en tres vertientes:

Builders que es el caso de apoyar a los desarrolladores e ingenieros de software que crean software y todo el staff involucrado en este proceso.

Breakers que es la comunidad de expertos en seguridad e interesados con el objetivo en común de mejorar el estado de la seguridad de las aplicaciones en el área de pruebas de seguridad

Defenders que es la comunidad de profesionales en el área de seguridad enfocados en mejorar el estado de la seguridad desde una óptica de defensa creando herramientas y técnicas para facilitar la detección y respuesta ante ataques a nivel de capa de aplicación.

Bajo esta perspectiva existen varios proyectos que ayudan significativamente a realizar el análisis de vulnerabilidades en aplicaciones Web, entre las que se incluyen:

OWASP ZAP (Zed Attack Proxy): Es una de las herramientas favoritas utilizada por miles de profesionales a nivel mundial para encontrar, de forma automática, vulnerabilidades en aplicaciones Web durante las etapas de desarrollo y pruebas dentro del ciclo de vida del desarrollo de Software. Es una herramienta muy conveniente tanto para personas experimentadas en el área de Penetration Testing que permite también realizar pruebas manuales. La página principal del proyecto – el cuál es uno de los más activos de OWASP brinda todas las herramientas para comprender y utilizar ZAP.

OWASP Web Testing Environment Project (WTE): Es una colección de herramientas de seguridad de aplicaciones y documentación disponible en múltiples formatos, tales como máquinas virtuales, paquetes de distribución de Linux, instalaciones basadas en la nube e imágenes ISO. La página principal del proyecto se encuentra en este enlace.

OWASP Testing Guide V4

OWASP Code Review Guide: En caso de que se tenga acceso al código fuente de la aplicación esta guía propone mejores prácticas y orientación en cómo y porqué realizar una revisión de código fuente desde el punto de vista de seguridad y que aspectos poner atención especial.

OWASP Dependency Check: Es una utilidad que identifica las dependencias del proyecto y verifica si existen vulnerabilidades conocidas y divulgadas públicamente. Actualmente se admiten Java y .NET. Se ha agregado soporte experimental adicional para Ruby, Node.js, Python y soporte limitado para sistemas de compilación C / C ++ (autoconf y cmake). Esta herramienta es muy útil pues forma parte del proceso de identificar vulnerabilidades en componentes de terceros – el cuál es el riesgo número 9 en el OWASP Top 10 2017 denominado Utilización de componentes con vulnerabilidades conocidas.

OWASP Top 10 : Es una guía agnóstica desde el punto de vista de tecnología y basada en un modelo de riesgos donde se consideran los 10 riesgos fundamentales o vulnerabilidades que afectan las aplicaciones Web. La versión más reciente de este informe la constituye la versión 2017 y se puede descargar y compartir libremente a partir de este enlace.

OWASP Application Security Verification Standard Project (ASVS): Proporciona una base para probar los controles de seguridad técnica de las aplicaciones Web y también proporciona a los desarrolladores una lista de requisitos para un desarrollo seguro.
El estándar proporciona una base para probar los controles técnicos de seguridad de la aplicación, así como cualquier control de seguridad técnica en el entorno, en los que se confía para

protegerse contra vulnerabilidades como Cross-Site Scripting (XSS) e inyección SQL. Este estándar se puede usar para establecer un nivel de confianza en la seguridad de las aplicaciones web. La versión más reciente de este standard es la 3.0.1 y se descarga desde el siguiente enlace

¿Cómo describiría usted el nivel de seguridad en forma general que posee Costa Rica a nivel de sitios web?

Uno de los aspectos fundamentales del tema de Seguridad a nivel global – y del cual Costa Rica es partícipe – es que tradicionalmente la seguridad en el software no se ha visualizado como un requerimiento funcional en el Ciclo de vida del desarrollo del Software. Aunado a ello, no es tan simple justificar el retorno de la inversión en seguridad y que tradicionalmente ha existido una desconexión entre los departamentos de desarrollo y seguridad.

Muchos de los problemas que vemos actualmente- los cuales causan mayor impacto en términos de riesgo- son vulnerabilidades que tiene más de 18 años de estarse explotando, donde existe mucha documentación y donde la cantidad de *exploits* (scripts automatizados para explotar una vulnerabilidad en particular) así como herramientas para detección y explotación de estos riesgos están públicamente disponibles pero que sin embargo aún se siguen presentando en aplicaciones nuevas. Este problema recurrente es un claro ejemplo de que la educación en cuanto a desarrollo de software seguro está todavía carente en la sociedad y lamentablemente en las aplicaciones costarricenses.

Sin embargo, no podemos culpar del todo al desarrollador de aplicaciones el cual tiene una agenda de trabajo bastante ardua donde todo es urgente y donde las dificultades intrínsecas del desarrollo de software le pasan una cara factura cuando se refiere a desarrollo seguro y calidad. Es más bien un esfuerzo en conjunto donde las personas que toman decisiones comprendan lo importante que es destinar esfuerzos para considerar la seguridad como el elemento fundamental

en el diseño de aplicaciones, más si consideramos los efectos tan adversos a los cuales se someten las organizaciones que son víctimas de un ataque informático[2].

A conforme proliferan las amenazas cibernéticas de forma tan acelerada (con vectores de ataque cada vez más novedosos) el país y la academia en combinación con la industria pública y privada deben apoyar iniciativas donde se inculquen principios básicos de desarrollo seguro de software, se incentive más los aportes de organizaciones como OWASP y se inviertan más recursos capacitando al personal en temas de código seguro.

Costa Rica posee una industria muy fuerte en el ámbito del desarrollo de aplicaciones y tenemos todos los recursos para colocarnos en una posición privilegiada y ser líderes en la región por adoptar una poción donde la seguridad en el software sea fundamental.

Debido a la falta de un benchmarking y de la premisa que sostiene que "**lo que no se puede medir no se puede mejorar"** es muy irresponsable dar un estimado del nivel de seguridad (o inseguridad) de las aplicaciones desarrolladas en el país, sin embargo, es desalentador cuando tomamos en perspectiva la frase célebre de Robert S. Muller lll, director del FBI[3] *"...I am convinced that there are only two types of companies: those that have been hacked and those that will be. And even they are converging into one category: companies that have been hacked and will be hacked again."*

[2] http://business.inquirer.net/240604/massive-data-breach-cost-equifax-nearly-90-million

[3] https://archives.fbi.gov/archives/news/speeches/combating-threats-in-the-cyber-world-outsmarting-terrorists-hackers-and-spies