REVISIÓN DEL ESTADO DEL ARTE Y DE LAS TÉCNICAS Y HERRAMIENTAS ORIENTADOS AL DESARROLLO DE UN SISTEMA INTEGRADO DE SOPORTE PARA ANÁLISIS DE VULNERABILIDADES EN SISTEMAS WEB

Cuevas, Juan Carlos, Muñoz Roberto Miguel, Gibellini Fabian Alejandro, Parisi German, Zea Cardenas Milagros, Barrionuevo Diego

Laboratorio de Sistemas
Departamento de Sistemas
Facultad Regional Córdoba
Universidad Tecnológica Nacional

Abstract

"En el contexto de las aplicaciones web se presenta en forma creciente y cotidiana el interés y la necesidad de abordar los aspectos referidos a su seguridad. A tal fin los análisis de vulnerabilidades constituyen la principal estrategia para evaluar su seguridad y las pruebas de penetración su tecnología por excelencia.

El objetivo del presente trabajo es realizar un estudio del arte en lo que a seguridad de la información se refiere en general y al análisis de vulnerabilidades en aplicaciones web y las pruebas de penetración asociadas a éste análisis en particular, todo ello en el contexto del proyecto de I+D denominado Sistema Integrado de Soporte para Análisis de Vulnerabilidades en Sistemas Web"

Palabras Clave

Aplicaciones web. Seguridad de la información. Vulnerabilidades. Pruebas de penetración.

Introducción

En la actualidad, el desarrollo de aplicaciones web posee un crecimiento sostenido en cuanto a cantidad, complejidad y su penetración en casi todos los aspectos de la vida de las personas, no sucediendo lo mismo con la calidad de estos desarrollos fundamentalmente en sus aspectos de seguridad. El rápido desarrollo en las tecnologías de redes y computación ha incrementado mucho la popularidad, y por ende su uso, lo que incrementa la circulación de información en la Web [1].

Se ha acentuado la cantidad de ataques a estas aplicaciones web debido fundamentalmente a las vulnerabilidades que poseen. Habida cuenta de una creciente toma de conciencia (especialmente a raíz del aumento de la cantidad de ataques) sobre estas vulnerabilidades y sus riesgos asociados, se ha producido un incremento por parte de las comunidades científica y empresarial en acciones orientadas al desarrollo de metodologías, técnicas y herramientas destinadas al abordaje de la problemática relacionada a dichas vulnerabilidades, esencialmente a través de una gestión de riesgos y las pruebas de penetración [2].

Los ataques cibernéticos se han tornado en una de las más grandes amenazas a la privacidad y confidencialidad de las personas como así también al mundo de la economía de negocios. El monto de los daños ha estado creciendo día a día y más compañías e instituciones se han tornado víctimas de violaciones de datos ejecutados por personas malintencionadas (black hat - cracker). Por lo tanto, las compañías e instituciones están a la búsqueda de mejores formas para proteger sus sistemas de información esencial. La forma más popular es probar sus sistemas vía pruebas de penetración por equipos éticos y calificados los cuales contribuyen en forma reactiva

y proactiva, a mejorar sus mecanismos de defensa para sus sistemas de información.
Estas acciones están enfocadas en incrementar la seguridad de aplicaciones web mediante evaluaciones de la seguridad de la información, proceso éste destinado a determinar el nivel de efectividad para alcanzar los objetivos de seguridad específicos de la entidad que está siendo accedida (por ejemplo, servidores, sistemas, red, procedimientos y personas -conocido como el objeto de la evaluación). [3]
En este contexto, desde el Laboratorio de Sistemas (LabSis) de la carrera Ingeniería de Sistemas de Información de la Universidad Tecnológica Nacional - Facultad Regional Córdoba (UTN - FRC), se ha decidido desarrollar un programa de I+D sobre esta temática de seguridad en sistemas de información. Uno de los proyectos que integran dicho programa es el denominado Sistema Integrado de Soporte para Análisis de Vulnerabilidades en Sistemas Web, el cual ha sido homologado por la Secretaría de Ciencia y Tecnología de la UTN en el corriente año.
El mencionado proyecto tiene por objetivo desarrollar un sistema integrado y evolutivo que permita gestionar la ejecución de múltiples pruebas de penetración en el contexto de la seguridad de la información de sistemas web en producción, basado en metodologías abiertas, para identificar y analizar sus vulnerabilidades. Con este sistema proyectado, y análogamente a lo que un IDE representa en el desarrollo de software, se pretende brindar un sistema con funcionalidades integradas y evolutivas que permitan incorporar nuevas técnicas y herramientas, estableciendo sus relaciones en el contexto de las metodologías que guían el proceso del pentesting, como así también, llevar a cabo un análisis de seguridad de software: lógica de programación, acceso al sistema, autenticación, base de datos, entre otros objetos de pruebas de penetración. A partir de los resultados de los análisis de seguridad realizados se procederá a generar informes técnicos que faciliten la tarea de resumir los resultados, registrando también los procesos aplicados para arribar a ellos, y las correspondientes recomendaciones para sus correcciones. Si bien existe una marcada evolución en cuanto a técnicas y herramientas, el sistema propuesto propende a incorporar esta evolución a los fines de brindar un soporte actualizado a los niveles académicos y de la industria del software en general, y a los profesionales de las pruebas de penetración (pentesters) en particular.
En resumen, el proyecto a desarrollar postula dos diferencias importantes en cuanto al estado del arte actual: el carácter evolutivo en cuanto a la incorporación continua de nuevas técnicas y herramientas y el establecimiento de sus relaciones, como así también, gestionar los resultados de las pruebas de penetración realizadas registrándolos.

La importancia de las pruebas de penetración en el contexto de la gestión de vulnerabilidades.

Actualmente todas las personas mantienen una estrecha conexión, sistemas de información mediante, con los servicios de Internet dando lugar a una relación intrínseca, ya que para realizar tareas cotidianas como llevar a cabo una compra, una reserva de un viaje, trámites con instituciones públicas y/o privadas, transacciones bancarias y muchas otras operaciones que son parte de la vida diaria, pueden ser realizadas a través de los servicios que se brinda en la web. En cualquiera de las operaciones antes mencionadas se intercambian datos y muchas veces los datos deben ser intercambiados a través de los canales de conexión que nos brindan los proveedores de Internet, canales que son sensibles y en algunos casos hasta pueden ser interferidos por lo que la información que se transfiere puede ser interceptada, la que de caer en la manos equivocadas genera riesgos vinculados a fraudes, estafas o engaños [4].

Existen disponibles normas, modelos y estándares para asistir a las organizaciones en la implementación de programas y controles apropiados para identificar y

mitigar estos riesgos, como por ejemplo, la norma ISO 27001 [5] y los modelos ITIL [6] y COBIT [7] y estándares del NIST [8].

En este contexto de los sistemas de información aparece como uno de sus ejes fundamentales su seguridad respecto a la información que estos almacenan. La norma ISO 27001 [5] define a la seguridad de la información como la preservación de la confidencialidad, integridad y disponibilidad de la información.

Al abordar la temática de seguridad en los sistemas de información resulta de carácter indivisible a ésta la cuestión relativa a los riesgos. Concepto este que, la antes mencionada norma, define como ”posibilidad de que una amenaza concreta pueda explotar una vulnerabilidad para causar una pérdida o daño en un activo de información. Suele considerarse como una combinación de un evento y sus consecuencias”. En esta definición aparece el concepto de vulnerabilidad, el cual siguiendo con la misma norma es definida como “debilidad de un activo o control que puede ser explotada por una o más amenazas”. Este mismo concepto de vulnerabilidad, el modelo OWASP lo define como “una falla o debilidad en el diseño, implementación, operación o administración de un sistema que puede ser explotada para violar la política de seguridad del sistema”. Una amenaza es un ataque potencial que, mediante la explotación de una vulnerabilidad puede dañar el propio activo (recurso de valor, tal como el dato en una base de datos o en el sistema de archivos) [9]. Así mismo OWASP define a una prueba como una acción que tiende a mostrar una vulnerabilidad en un sistema. Por último para terminar este breve recorrido hacía el tratamiento de las vulnerabilidades debemos resaltar el concepto de las pruebas de penetración, y en nuestro caso fundamentalmente orientadas a las aplicaciones web.

Sobre este tema OWASP define a las pruebas de penetración de aplicaciones web como un método de evaluación de la seguridad de una red o sistema computarizado mediante la simulación de un ataque haciendo referencia solo a la evaluación de la seguridad de una aplicación web. Este proceso involucra un análisis activo de la aplicación para cualquier debilidad, falla técnica o vulnerabilidad. Cualquier cuestión de seguridad que es encontrada será presentada al propietario del sistema junto con una evaluación de su impacto y aún con una propuesta de mitigación o de solución técnica [9].

No existe el sistema 100% seguro. En este contexto aparecen las figuras del hacker y cracker, siendo ambos personas altamente capacitadas y motivadas que buscan vulnerar el sistema, pero, los aspectos éticos y morales de qué acciones realizar con las vulnerabilidades identificadas es lo que los diferencia. El primero identifica vulnerabilidades para mejorar la seguridad del sistema y en el caso del segundo sus acciones son malintencionadas.

Por otro lado, existen metodologías como OWASP [9], OSSTMM [10], ISAAF [11], técnicas como inyección de código [12] [13] [14], Cross Site Scripting [15] y herramientas tales como SQLMap [16] [17], entre otras, que combinadas con las normas, modelos y estándares antes mencionadas conforman un conjunto de buenas prácticas con técnicas y herramientas orientadas a abordar la complejidad implícita de la temática de la seguridad de la información que a la fecha han logrado escasos resultados proactivos y si, mayoritariamente, reactivos.

Las pruebas de penetración son diferentes de las pruebas funcionales de seguridad, estas últimas procuran demostrar el comportamiento correcto de los controles de seguridad del sistema mientras que las pruebas de penetración busca determinar la dificultad que implica para alguien el intentar penetrar los controles de seguridad de la organización y lograr accesos no autorizados a su información y sistemas de información. Estas pruebas son llevadas a cabo mediante la simulación de un usuario no autorizado atacando el sistema mediante el uso de herramientas automatizadas, métodos manuales o una combinación de

ambos. Estas pruebas de penetración proporcionan beneficios tanto desde las perspectivas de negocio como desde la manera operacional [18].

Metodología

El desarrollo de este proyecto tiene previsto llevarse a cabo a través del método científico, en cuanto a la forma investigativa y el tratamiento de los datos, pero aplicando administración de proyecto, con el método espiralado de seguimiento para la planificación del desarrollo, que estará basado en cuatro etapas, que se repiten en forma cíclica hasta la culminación del proyecto: planeamiento, adquisición del conocimiento, codificación y evaluación. Si bien la investigación se enfoca en el área de la seguridad informática, y tomando en cuenta que hay un importante conocimiento disponible en este campo pero que carece de la organización e integración del mismo, no es intención en este proyecto realizar una investigación progresiva en el paso del tiempo, por ello se encuadra en el tipo de investigación transversal. Esta investigación está enfocada en los conocimientos contemporáneos sobre la temática. Este proyecto está enmarcado en los recientes conceptos y métodos de Investigación, Desarrollo e Innovación (I+D+i), que están fuertemente ligados a la mirada de desarrollo tecnológico y el ingreso del producto al mercado y su uso, basado en la secuencia: síntesis y teoría; explorar, hipotetizar y clarificar; diseño, desarrollo y prueba; implementación, estudio y mejora de la eficacia.

Las acciones llevadas a cabo hasta el momento en el contexto del proyecto de investigación antes mencionado se resumen en:

- Investigación exploratoria a los fines de profundizar y completar el estado del arte.
- Identificación de las metodologías, técnicas y herramientas disponibles en la actualidad.

Resultados obtenidos

La seguridad de la información describe actividades relativas a la protección de la información y los activos de la infraestructura de la información contra riesgos de pérdida, uso inadecuado, revelación o daño.

Los riesgos de estos activos pueden ser calculados mediante el análisis de las siguientes cuestiones:

Amenazas a sus activos: Eventos no deseados que pueden causar pérdida, daño o uso inadecuado de los activos en forma deliberada o accidental.

Vulnerabilidades: Se refiere a cuán susceptibles son sus activos a ataques.

Impacto: La magnitud de la pérdida potencial o la seriedad del evento.

Entre los principales resultados obtenidos de la realización de las dos etapas primigenias mencionadas ut supra, a la fecha de elaboración de este documento, corresponde destacar los siguientes resultados:

En lo que respecta al Estado del arte diferentes autores abordan en sus bibliografías las temáticas de las vulnerabilidades y de las pruebas de penetración en el contexto de la Seguridad en los sistemas de información, como por ejemplo: Von Solms [24] identifica diez aspectos esenciales (pecados mortales de la seguridad de la información), los cuales si no son tenidos en cuenta en un plan de gobierno de la seguridad de la información, seguramente causarán que el plan falle o de última seguramente causarán fallas o defectos en el plan. El NIST, además de definir una evaluación de la seguridad de la información, plantea tres métodos de evaluación que pueden ser usados para alcanzarla: las pruebas, el examen y las entrevistas. Entre otras pruebas, este estándar, hace referencia a las pruebas de penetración [8], por su parte, OWASP, organización enfocada en la mejora de la seguridad del software, propone un conjunto de guías, a saber: Referencia de escritorio en seguridad de aplicaciones, Guía de desarrollo, Guía de pruebas y Guía de

Revisión de código, [28], Song et al postulan que la seguridad en los sistemas de información está conformada por la seguridad física, de red, de host, de aplicación y de datos, destacando que la seguridad del sistema operativo constituye la base sobre la que se fundamentan la seguridad de las aplicaciones y de los datos [26], Grossman postula que las aplicaciones web de diferentes dominios de uso más frecuente (por ejemplo: banca, salud, TI, educación, redes sociales) son más propensas a ser vulneradas [33], Shahriar propone. en el contexto de la seguridad de la información, una tipología, que no es extensiva, de vulnerabilidades en aplicaciones web, y que vinculadas a ellas existen dos enfoque de mitigación posibles: el testing y el de monitoreo [27], Pauli sostiene que existen aplicaciones web descaradamente vulnerables, principalmente porque los programadores están más concentrados en la funcionalidad que en la seguridad [34], Steve Durbin [25], director administrativo del Information Security Forum (ISF), afirma que a pesar de que el año calendario 2014 se ha cerrado, se espera que el tamaño, la severidad y complejidad de las amenazas a la web continuarán incrementándose y postula cinco tendencias que dominarán la seguridad de la información en el 2016, Para una organización es muy importante adoptar una cultura de seguridad de información, para lo cual Alnatheer [23] identifica la existencia de factores críticos para el éxito en esta adopción.

A su vez, algunos autores hacen foco en las vulnerabilidades, como por ejemplo: Bates et al postulan que en los últimos años, los fabricantes de navegadores e investigadores han tratado de desarrollar filtros del lado del cliente para mitigar estos ataques. Pero algunos de estos filtros podrían introducir vulnerabilidades en sitios que antes estaban libres de errores [31], Xie et al postulan que muchas de las vulnerabilidades de seguridad en las aplicaciones actuales son introducidas por los desarrolladores de software al escribir código inseguro los cuales, continúan los autores, se pueden deber a la falta de comprensión de programación segura y/o lapsus de atención de los desarrolladores sobre el tema de seguridad. Esta propuesta está orientada fundamentalmente al tema de vulnerabilidades de seguridad durante el ciclo de vida del desarrollo de la aplicación [30], Tripp et al postulan que los autores de publicaciones recientes sugieren que las aplicaciones web son altamente vulnerables a ataques de seguridad. En referencia a lo cual citan un reporte reciente de la WASC que proveen estadísticas de seguridad sobre 12186 sitios web en producción, listando un total de 97554 vulnerabilidades detectadas en estos sitios web. Más severamente aún es que cerca del 49% de los sitios analizados se encontró que contenían vulnerabilidades de alto riesgo [32], OWASP propone los top ten de vulnerabilidades y de controles proactivos en las aplicaciones web [29].

En lo que se refiere a pruebas de penetración algunas organizaciones y autores se focalizan en ellas, como por ejemplo: ITIL [6] en su apartado de Gestión de la Seguridad de la Información hace referencia a ejecutar, revisiones, auditorías y pruebas de penetración, Una prueba de seguridad puede ser definida como un intento legal y autorizado para ubicar y explotar sistemas de información con el propósito de hacer un sistema más seguro. El proceso incluye pruebas de vulnerabilidad que deben terminar con recomendaciones específicas para arreglar los problemas descubiertos durante las mismas. Estas pruebas de seguridad son conocidas como: pentesting, PT, hacking, ethical hacking, white hat hacking [35], Bavici, por su parte, postula que las pruebas de penetración ayudan a determinar cuáles vulnerabilidades son explotables y el grado de exposición de la información o control de la red que la organización podría esperar de un atacante alcanzará después de explotar exitosamente una vulnerabilidad [21], Una prueba de penetración es un intento autorizado y proactivo para probar y revisar la seguridad de una infraestructura de TI mediante

intentos seguros para explotar las vulnerabilidades del sistema incluyendo sistema operativo, fallas de la aplicación o del servicio, configuraciones impropias y a un comportamiento del usuario riesgoso o peligroso. Las pruebas son ejecutadas usando tecnologías automatizadas y manuales para comprometer servidores, sistemas, aplicaciones web, redes inalámbricas, dispositivos de redes, móviles y otros puntos de exposición. [37]. Jacobs hace referencia que las pruebas de penetración involucran un análisis activo del sistema objetivo para vulnerabilidades potenciales que pueden resultar de una configuración de sistema impropia o pobre, fallas de software o hardware conocidas o desconocidas, o debilidades operacionales en procesos [22].

Por último se hace referencia a autores que abordan las relaciones entre vulnerabilidades y Pruebas de Penetración, como por ejemplo: Withman postula que mientras en la mayoría de las pruebas de seguridad, tales como la evaluación de vulnerabilidad, se toma mucho cuidado en no interrumpir las operaciones normales del negocio, en las pruebas de penetración el analista trata de llegar tan lejos como sea posible, simulando las acciones de un atacante [20], Vacca, entre otros interesantes aspectos, aborda en el contexto de la seguridad de la información las diferencias entre las pruebas de penetración y la evaluación de vulnerabilidad [19] y las nociones definidas por las palabras penetración y vulnerabilidad son complementarias a cada una de la otra. Si hay vulnerabilidad, como la destructividad que corresponde a la funcionalidad del sistema web puede penetrarla como si fuera un agujero. Lo contrario es verdadero, si la penetración fue detectada ella sucedió debido a la vulnerabilidad (agujero) [36].

En lo referente a la Identificación de las metodologías, técnicas y herramientas disponibles se identificaron, entre otros, los siguientes aportes: Sadeghian et al sostienen que la mayoría de las aplicaciones web usan motores de bases de datos relacionales, esto hace que sean el centro de atención de los atacantes, encontrar vulnerabilidad en el SQL de la aplicación web suele ser catastrófico porque fácilmente uno puede cambiar datos, por supuesto, teniendo los privilegios apropiados [38]. Ke Wei et al. proponen una forma de prevenir SQL Injection usando procedimientos almacenados en la base de datos [39]. Takahashi et al describen las vulnerabilidades XSS que permiten ejecutar código de un atacante para robar las credenciales de autenticación de usuarios víctimas, además proponen un método de “desafío-respuesta” para validar si un usuario es el propietario legítimo de sus credenciales [40]. Stasinopoulos et al abordan los ataques de tipo XSS que permiten ejecutar código de un atacante para robar las credenciales de autenticación de usuarios víctimas, además proponen un método de “desafío-respuesta” para validar si un usuario es el propietario legítimos de sus credenciales [40]. Stasinopoulos et al abordan los ataques de tipo XSS que los navegadores web intentan filtrar mediante procesos previos a la renderización de un sitio web, ellos, sin embargo, proponen una forma de burlar este sanitizado que es efectiva bajo ciertas condiciones [41].

Estos autores postulan modelos teóricos focalizados en técnicas: como ser SQL Injection, XSS, etc., no haciendo referencia a herramienta alguna en sus publicaciones. En lo que se refiere al estado del arte surgen algunas reflexiones primigenias: la cantidad de publicaciones relativas a pruebas de penetración son menores a las referidas al tema de vulnerabilidades, las cuales a su vez también son menores referidas al tema de seguridad de la información; de la documentación analizada se identificó que sólo una de ellas integra los temas de seguridad de la información, vulnerabilidad y pruebas de penetración, a saber el estándar del NIST; la mayoría de las publicaciones analizadas abordan la temática para aplicaciones web en producción con algunas excepciones orientadas al ciclo de vida de desarrollo de la aplicación; y por último

podemos reflexionar que algunas publicaciones hacen referencia a que las vulnerabilidades son generadas por los programadores o desarrolladores.

En cuanto a la identificación de las metodologías, técnicas y herramientas contemporáneas disponibles, se está realizando actualmente un análisis exhaustivo que ha precipitado resultados parciales de esta etapa, los cuales son representados en un mapa de tipos de ataques que permiten apreciar el vínculo implícito entre los identificados, sus técnicas y algunas herramientas probadas que permiten descubrir ciertas vulnerabilidades.

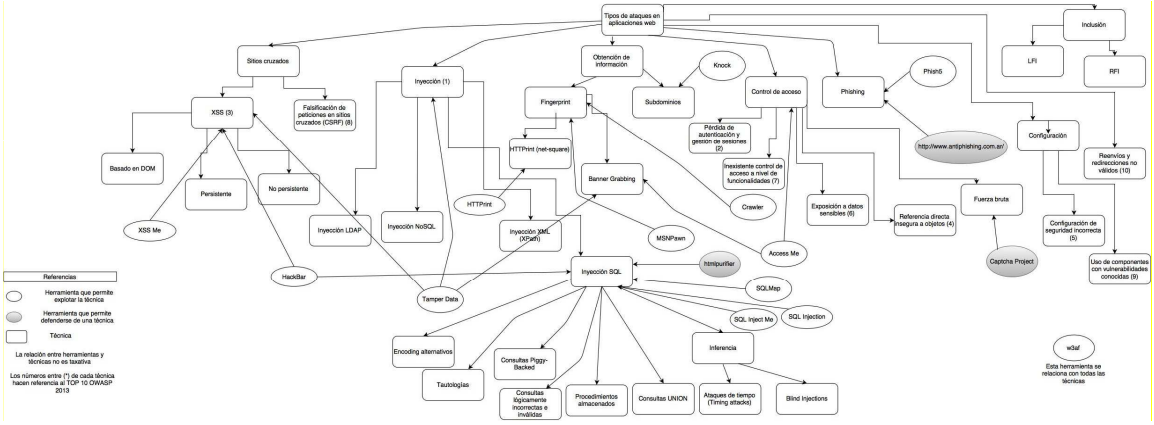

Figura 1. Relaciones entre tipos de ataque, técnicas y herramientas. (Ver Anexo A)

En la Figura 1 se pueden visualizar técnicas de ataque hacia un sistema web. Las técnicas de ataque están representadas mediante rectángulos. Las asociaciones determinan la dependencia de las técnicas (1) y pueden agruparse bajo una misma técnica de ataque (padre).

Por otro lado, están las herramientas representados por óvalos. Se pueden visualizar dos tipos de herramientas:

- Las que permiten ejecutar una técnica de ataque (color de fondo blanco).
- Las que permiten defenderse de una técnica de ataque (color de fondo gris).

Las relaciones representadas por flechas direccionales que inician en las herramientas y se dirigen a las técnicas representan que una herramienta puede ejecutar (o defender) la técnica apuntada.

Conclusión

En muchas organizaciones las pruebas de seguridad que se llevan a cabo son realizadas fuera de las pruebas que se realizan durante el desarrollo, siguiendo un enfoque de "búsqueda de vulnerabilidades". El equipo de seguridad a través de herramientas de escáner o través de técnicas conocidas o personalizadas realiza dichas pruebas de seguridad y sus resultados son presentados al equipo de desarrollo en una lista de vulnerabilidades a ser "arregladas".

En lo referido a sistemas web, de no tener presente durante su desarrollo la seguridad del sistema, sus datos pueden quedar expuestos o vulnerables y pueden ser el punto de entrada para un ataque malintencionado. Además, el software es un producto cuya evolución es continua, traduciéndose esta evolución en nuevas funcionalidades y componentes que deben ser analizados en busca de vulnerabilidades. Estos factores incorporados a la aplicación web pueden afectar al sistema, siendo también de particular atención el mantenimiento del sistema, donde cambios y/o actualizaciones que se lleven a cabo con el tiempo pueden generar vulnerabilidades que anteriormente no existían.

Por lo expuesto, al momento de culminar su etapa de implementación y pasar a producción es indispensable que el control y análisis de los riesgos y vulnerabilidades se lleve a cabo periódicamente, realizando distintas pruebas para verificar la consistencia y permanente convergencia del sistema a la integridad y confidencialidad de los datos que este mismo resguarda.

Con respecto al estudio de técnicas y tipos de ataques podemos resumir que su importancia ha crecido mucho debido al amplio desarrollo de sistemas web, de la tecnología y de los accesos a las mismas. La exposición de las aplicaciones web permiten desarrollar nuevas técnicas y tipos de ataques que son más efectivos y conllevan esfuerzos de muchos recursos para mitigarlos. Para el personal de seguridad el objetivo es verificar cuál es o será el comportamiento de los mecanismos de defensa y detectar vulnerabilidades que puedan ser comprometidas en el sistema bajo análisis.

En lo que se refiere al estado del arte surgen algunas reflexiones primigenias: la cantidad de publicaciones relativas a pruebas de penetración son menores a las referidas al tema de vulnerabilidades, las cuales a su vez también son menores referidas al tema de seguridad de la información; de la documentación analizada se identificó que sólo una de ellas integra los temas de seguridad de la información, vulnerabilidad y pruebas de penetración, a saber el estándar del NIST; la mayoría de las publicaciones analizadas abordan la temática para aplicaciones web en producción con algunas excepciones orientadas al ciclo de vida de desarrollo de la aplicación; y por último podemos reflexionar que algunas publicaciones hacen referencia a que las vulnerabilidades son generadas por los programadores o desarrolladores.

Si bien el proyecto que da origen a este documento hace expresa referencia a su aplicación postproducción, esto no implica que este sistema no sea aplicable antes de la puesta de producción del mismo.

Referencias

[1] Sarhan, A., Hamissa, G. M., Elbehiry,. Proposed document frecuency technique for minimized dataset in web crawler. IEEE 2015.

[2] Stenfinko, Y., Piskozub, A., Banakh, R. “Manual automatic penetration testing. Benefics and draw bugs. Modern tendencys”. TCSET ‘2016 LVV - SLAVSKE UCRAINE

[3] Scarfone, K., Souppaya, M., Cody, A. & Orebaugh, A. “Technical Guide to Information Security Testing and Assessment ”, NIST. USA, 2008.

[4] Khan, Rasib. and Ragib Hasan. "The Story of Naive Alice: Behavioral Analysis of Susceptible Internet Users". IEEE 2016

[5] ISO/IEC 27001. “Tecnología de la información”. Técnicas de la seguridad. Sistemas de gestión de la seguridad de la información (SGSI). Requisitos. ISO Ginebra, Suiza 2013.

[6] ITIL. Information Technology Infrastructure Library. Vs. 3 - 2011.

[7] COBIT Control Objectives for Information and Technology. Vs. 5 - 2012.

[8] Technical Guide to Information Security Testing and Assessment. SP 800-115. NIST National Institute of Standards and Technology. 2008.

[9] Guía OWASP (Open Web Application Security Project). Vs. 3. 2008 (vs 1).

[10] “Open Source Security Testing Methodology Manual (OSSTMM)”. Institute for Security and Open Methodologies (ISECOM). Diciembre 2010. Cataluña. España.

[11] “Information System Security Assessment Framework (ISAAF)”. Open Information Systems Security Group. Londres, Inglaterra.

[12] Nagpal, B,. Chauhan, N., Singh, N., Panesar, A.: “Tool Based Implementation of SQL Injection for Penetration Testing”. International Conference on Computing, Communication and Automation (ICCCA2015). 2015

[13] Sadeghian, A., Zamani, M., Abdullah, S. M. “A taxonomy of SQL Injection Attacks”. International Conference on Informatics and Creative Multimedia. 2013.

[14] Qian, L., Zhu, Z., Hu, J., Liu, S. “Research of SQL Injection Attack and Prevention Technology”. International Conference on Estimation, Detection and Information Fusion (ICEDIF). 2015.

[15] Yusof, I., Pathan, A. S. “Preventing Persistent Cross-Site Scripting (XSS) Attack By Applying Pattern Filtering Approach”. IEEE. 2014.

[16] Ciampa, A., Visaggio, C. A., y Di Penta, M. “A heuristic-based approach for detecting SQL-injection vulnerabilities in Web applications”. SESS. Cape Town, South Africa. Mayo 2/2010.

[17] Nagpal, B., Chauhan, N., Singh, N., Panesar, A. “Tool Based Implementation of SQL Injection for Penetration Testing”. International Conference on Computing, Communication and Automation (ICCCA2015). 2015

[18] Bacudio, A., Yuan, X., Bill Chu, B., Jones, M. “AN OVERVIEW OF PENETRATION TESTING”. International Journal of Network Security & Its Applications (IJNSA). 2011

[19] Vacca, J. R. “Computer and Information Security Handbook”. Morgan Kausmann Publishers of Elsevier. USA. 2013.

[20] Withman. M. E., Mattord, H. J. “Principles of Information Security”. Course Technology Cengage Learning. USA. 2012.

[21] Bavici, S. “Managing Information Security”. Syngress of Elsevier. USA. 2014.

[22] Jacobs, S. “Engineering Information Security”. IEEE Press Wiley. USA. 2016.

[23] Alnatheer, M. A. “Information security culture critical success factors”. IEEE. Computer Society. USA. 2015

[24] Von Solms, R. “The ten deadly sins of information security management”. Computers and Security 23,371-376. ELSEVIER ltd delimited. 2004.

[25] Durbin, S., “Information security trends that wil dominate 2015”. Acceso al texto: 27-08-2016. http://www.cio.com/article/2857673/security/5-information-security-trends-that-will-dominate-2015.html

[26] Song, J., Hu G., Xu S. “Operating System Security and Host Vulnerability Evaluation”. School of Information Science and Engineering. Shenyang University of Technology, SUT. Shenyang, China. 2009.

[27] Shahriar, H. “Security Vulnerabilities and Mitigation Techniques of Web Applications”. Department of Computer Science. Kennesaw State University. Kennesaw. USA. 2013.

[28] OWASP. Página oficial. https://www.owasp.org/index.php/Main_Page

[29] “OWASP top 10 2013 Project”. “OWASP Proactive Controls”. Open Web Application Security Project. 2013.

[30] XIE, J., Chu B., Lipfort, H. R., Melton, J. T. “ASIDE: IDE Support for Web Application Security”. ACSAC ’11. Orlando, Florida USA. Dec. 5-9/2011

[31] Bates,D., Barth, A., Jackson, C. “Regular Expressions Considered Harmful in

Client-Side XSS Filters”, Raleigh, NC, USA. April 26-30/2010.

[32] Tripp, O., Weisman, O., Guy, L. “Finding Your Way in the Testing Jungle”. ISSTA ’13. Lugano, Switzerland. July 15-20/2013

[33] Grossman, J. “How does your website security stack up against peers?” White Hat Report. 2012. Acceso al texto: http://es.slideshare.net/jeremiahgrossman/whitehats-12th-website-security-statistics-full-report.

[34] Josh Pauli. "The Basics of Web Hacking: Tools and Techniques to Attack the Web". Ed. ELSEVIER. 25 Wymnan Street, Waltham, MA 02451, USA. 2013.

[35] Engebretson, P. "The Basics of Hacking and Penetration Testing". Elsevier 225 Wyman Street, Waltham, MA 02451, USA. 2011.

[36] Hahanov, V., Hayford, A., Ahmetoglu, A. H., Nunrmirradovich, J. D. Abeid, A. M., Stanley, O. “Pentesting and vulnerability diagnosis”. CADSM. Poyana - Svalyava (Zakrpattia). UKRAINE. 2013.

[37] Nagpal, B., Chauhan, N., Shing, N y Paneser, A. “Tool based implemetation of SQL Injection for penetration testing”. International Conference on Computing, Communication and Automation (ICCCA). IEEE. 2015.

[38] Sadeghian, A., Zamani, M., Abdullah, S. M. “A taxonomy of SQL Injection Attacks”. Advanced Informatics School. Universiti Teknologi Malaysia. Malaysia. 2013.

[39] Ke Wei, M., Muthuprasanna, S. “Preventing SQL Injection Attacks in Stored Procedures”. Dept. of Electrical and Computer Engineering. Iowa State University. 2006.

[40] Takahashi, H., Yasunaga, K., Mambo, M., Kim, K., Youl Youm, H. “Preventing Abuse of Cookies Stolen by XSS”. Eighth Asia Joint Conference on Information Security. 2013.

[41] Stasinopoulos, A., Ntantogian, C., Xenakis, C. “Bypassing XSS Auditor: Taking Advantage of Badly Written PHP Code”. Department of Digital Systems. University of Piraeus. 2014.

Datos de Contacto

Juan Carlos Cuevas.
Dpto de Ing. En Sistemas de Información – UTN FRC
juancarloscue@gmail.com

Roberto Miguel Muñoz
Dpto de Ing. En Sistemas de Información – UTN FRC
robertmunioz@gmail.com

Fabian Alejandro Gibellini
Laboratorio de Sistemas
Dpto de Ing. En Sistemas de Información – UTN FRC
fgibellini@bbs.frc.utn.edu.ar

German Parisi
germannparisi@gmail.com
Laboratorio de Sistemas
Dpto de Ing. En Sistemas de Información – UTN FRC

Milagros Zea Cardenas
Laboratorio de Sistemas
Dpto de Ing. En Sistemas de Información – UTN FRC
milyzc@gmail.com

Diego Barrionuevo
Laboratorio de Sistemas
Dpto de Ing. En Sistemas de Información – UTN FRC
santosdiegob@gmail.com