Alumno: Silva Pablo

Estado del arte

Muchas veces se confía que las aplicaciones web donde se deposita nuestra información es confiable y que los datos están seguros, pero no siempre es así, según una encuesta de Oracle [1] se puede observar que la mayoría de las bases de datos de las organizaciones carecen de seguridad suficiente, y por el número de ataques a sitios web dejando expuesta sensible a manos de los atacantes, se está aumentando la presión por mejorar la seguridad de las bases de datos [2].

Actualmente hay un proyecto llamado OWASP[3] que realizo un TOP 10 de los principales riegos de las aplicaciones web, entre los que se encuentran las inyecciones y las secuencias de comandos de sitios cruzados como los más explotados.

En este proyecto se van a tratar técnicas para mitigar las inyecciones SQL[4] y XSS[5] [6] [7][8][9] para lo que se hará será controlar las entradas a las funciones de la aplicación para prevenir que estas causen un comportamiento indebido en la misma [10] [7][6], siendo esta una de las vulnerabilidades más comunes en los aplicativos web.

Objetivo general:

Estudiar las técnicas de Seguridad Web orientadas a la prevención de inyecciones de código malicioso y aplicarlas en el desarrollo de un sistema Web para el INICNE.

Objetivos específicos:

-Aplicar medidas de protección contra inyecciones y cross site scripting

-profundizar el estudio de conceptos de seguridad web

-aplicar configuraciones del servidor web para evitar posibles brechas de seguridad

-Implementar un framework para aumentar la seguridad

- Diseñar y desarrollar una base de datos

Fundamentación

La idea central es estudiar las distintas formas de protegerse contra las secuencias de comandos de sitios cruzados (XSS) e inyecciones SQL (SQLI) las cuales a pesar de que son riesgos desde ya hace muchos años, se puede ver en las encuestas de WhiteHat [11] que siguen estando activas, explotando vulnerabilidades en sistemas heredados, o aplicaciones modernas con fallas en seguridad [12].

Para llevar a la práctica estos conocimientos, se desarrollara una aplicación multiusuario con orientación web para el INICNE (Instituto de Ictiología del Nordeste), que actualmente carece de un sistema, sino que utiliza planillas Excel individuales como medio de resguardo de la información tanto para su archivado como para su análisis.

Con una aplicación web multiusuario con una base de datos centralizada se crearían nuevas posibilidades como compartir información entre los distintos investigadores del instituto, poder diferenciar datos de distintas campañas, realizar búsquedas más específicas, obtener datos de forma ordenada y la capacidad de generar informes.

Metodología

CICLO DE VIDA DEL PROYECTO

Se optara por una metodología incremental[13]. En la etapa de análisis antes de la primera iteración, se hará un análisis de riesgos de seguridad del proyecto, y se definirán políticas de seguridad para mitigar los mismos, lo que permitirá que en cada incremento, se pueda hacer un nuevo análisis de riesgos de seguridad del incremento y aplicar las políticas definidas.

En cuanto a la seguridad se seguirá una metodología basada en [14], consistiendo en pasar las entradas por 3 filtros, el primero para escapar caracteres que puedan ser peligrosos, el segundo consiste en usar librerías para limpiar el código ingresado y por último se usaran técnicas de codificación para prevenir ataques en entornos HTML.

Ámbito de trabajo:

El trabajo se desarrollara principalmente de forma independiente, consultando a los futuros usuarios de la aplicación para obtener información e ir comprobando las versiones de la aplicación, y acudiendo a la FaCENA para consultas al profesor orientador

Recursos:

Como recursos informáticos hará uso de una computadora en la cual se programara con HTML, PHP y JavaScript con el uso un framework de PHP, ejecutados sobre un servidor Apache y abasteciéndose de datos de Mysql. Para poder realizar esto se hará uso de las documentaciones de los lenguajes frameworks, como también uso de bibliografía adicional para profundizar más sobre la temática de la seguridad [7], [6] y [8].

Desarrollos propuestos:

En este trabajo se propone investigar y aplicar medidas defensivas y preventivas para proteger los datos y la aplicación, empezando por la protección contra accesos indebidos, siguiendo con la administración de la base de datos para anular posibles intentos de operaciones indebidas y finalmente, realizando acciones para mitigar intentos de inyección maliciosa de código, tanto inyecciones SQL como cross site scripting.

La aplicación que se propone diseñar y desarrollar será web, capaz de almacenar y explotar grandes volúmenes de datos muestrales, para lo cual se configurara un servidor Apache, no solo para que pueda hospedar la aplicación sino para que pueda actuar como una medida más de seguridad, evitando accesos indebidos y registrando distintos tipos de sucesos.

Dentro de la a aplicación se usaran distintos perfiles de usuarios, por motivos de que los investigadores trabajan en distintos proyectos, todos los usuarios podrán agregar y consultar datos y generar reportes, y el administrador tendrá la capacidad de gestionar las cuentas asi como asignar investigadores a proyectos.

Por otra parte se deberá diseñar e implementar una base de datos. La misma se deberá crear en base a datos tabulados en planillas Excel que contienen actualmente los datos.

Resultados esperados:

Se espera poder aprender sobre los distintos riesgos de seguridad de aplicaciones web, especialmente sobre las inyecciones SQL y cross site scripting, ya que para poder defenderse se debe conocer cuáles son los posibles riesgos.

También espera diseñar una aplicación que pueda servir no solo para satisfacer las necesidades del INICNE, sino también como una guía para futuros alumnos que quieran profundizar más en la programación web, ya sea con el uso de Frameworks para php, como también en realizar aplicaciones más seguras.

Referencias:

[12] L. Salgado, M. Ron, F. Solis “Análisis de Riesgos de las Aplicaciones Web de la Superintendencia de Bancos y Seguros, Utilizando las Recomendaciones TOP TEN de OWASP.” Universidad de las Fuerzas Armadas ESPE. Marzo 2014.

[3]OWASP (12, junio, 2013) OWASP Top Ten Project [online].Disponible: https://www.owasp.org/images/5/5f/OWASP\_Top\_10\_-\_2013\_Final\_-\_Espa%C3%B1ol.pdf

[1] J. McKendrick” Dbs Security Superhero” 2014 IOUG ENTERPRISE DATA SECURITY SURVEY. Oracle. Octubre 2014

[d] R. Hansen “XSS cheat sheet for filter evasion.” OWASP. [online]disponible: https://www.owasp.org/index.php/XSS\_Filter\_Evasion\_ Cheat\_Sheet

[5] B. B. Guptaa, S. Guptaa, S. Gangwara, M. Kumara & P. K. Meenaa” Cross-Site Scripting (XSS) Abuse and Defense: Exploitation on Several Testing Bed Environments and Its Defense” Journal of Information Privacy and Security, Vol 11, No 2, Special Issue: Secured Communication in Wireless and Wired Networks. pp 118-136, 2015

[6]P. Brady (2015, octubre, 26)”Survive the Deep End: PHP Security” [online]. Disponible: http://phpsecurity.readthedocs.org/en/latest/

[g]S. Amirmohammad, M. Zamani, and A. A. Manaf. “SQL Injection is Still Alive: A Study on SQL Injection Signature Evasion Techniques," ICICM. Informatics and Creative Multimedia, 2013 Conferencia internacional, pp. 265-268.

[7] D. Stuttard, M. Pinto “The Web Application Hacker's Handbook: Finding and Exploiting Security Flaws” 2da Edicion, Indianapolis, Indiana: John Wiley & Sons, Inc. 2011.

[4] E. Couture “Web Application Injection Vulnerabilities a Web App’s Security Nemesis?” GIAC (GWAPT) Gold Certification .SANS Institute. Mayo 2013

[9] S. Fogie, J. Grossman, R. Hansen, A. Rager, P. D. Petkov “XSS Attacks: Cross Site Scripting Exploits and Defense.” Syngress Publishing, Inc. Burlington, MA. 2007

[10]B. Gupta “Web Application Security – What You Need to Know” THIRTY-THIRD ANNUAL PACIFIC NORTHWEST SOFTWARE QUALITY CONFERENCE. World Trade Center Portland Portland, Oregon. pp 271-280 October 2015

[l]Open Web Application Security Project (OWASP). OWASP Web Goat project. Bel Air, MD: OWASP. Retrieved June 2, 2015 from https://www.owasp.org/index.php/Category:OWASP\_WebGoat\_Project

[m]L. Conklin, G. Robinson “code review guide” OWASP. [Online]. Disponible <https://www.owasp.org/images/7/78/OWASP_AlphaRelease_CodeReviewGuide2.0.pdf>

[8] S. Fogie, J. Grossman, R. Hansen, A. Rager, P. D. Petkov “XSS Attacks: Cross Site Scripting Exploits and Defense” Syngress, 1ra edicion. Mayo 2007

[2] A. Bamrara” Evaluating Database Security and Cyber Attacks: A Relational Approach” Journal of Internet Banking and Commerce, Agosto 2015, vol. 20, no. 2

[11] WhiteHat Security (2015)” WhiteHat Security Status Report 2015” [online]. Disponible: <https://info.whitehatsec.com/rs/whitehatsecurity/images/2015-Stats-Report.pdf>

[14] OWASP(2016) “XSS (Cross Site Scripting) Prevention Cheat Sheet”[online]. Disponible: <https://www.owasp.org/index.php/XSS_(Cross_Site_Scripting)_Prevention_Cheat_Sheet>

[13] R. S. Pressman”Ingenieria del Software:un enfoque practico”