Paradigmas de Programação Causadores de Falhas de Segurança em Aplicações Web

Willian C. De S. De Oliveira[1], André R. Zavan[1]
Curso Técnico em Informátca Integrado ao Ensino Médio
[1]Instituto Federal do Paraná (IFPR) – Campus Paranavaí
CEP: 87.706–304 – Paranavaí – PR – Brasil

williancligor@gmail.com[1],andre.zavan@ifpr.edu.br[1]

Atualmente as redes de computadores não se limitam ao uso apenas dentro de uma organização para facilitar a troca de informações e arquivos entre seus funcionários. A implementação da rede mundial de computadores, conhecida como *Internet*, deu o poder aos usuários domésticos de acessar e compartilhar informações, comunicar-se com outros indivíduos e até mesmo fazer negociações, tudo isso em escala planetária [Tanenbaum 2013]. Por esses motivos, a *Internet* nos dias de hoje é um grande nicho de mercado. Nela pode-se comprar e encomendar produtos, fazer transações bancárias, quitar dívidas, e gerar valor monetário de muitas outras maneiras utilizando-se dos chamados sistemas *web*. Esses sistemas estão em constante risco por se encontrarem conectados com o mundo todo e por isso merecem uma atenção especial de seus analistas e desenvolvedores.

Ataques contra aplicações disponíveis na *web* representam a maior parte de incidentes de segurança e a preocupação é tanta que organizações foram criadas para discutir métodos de prevenção. No ano de 2017 foi constatado que as dez maiores falhas de encontradas nessas aplicações são: *injection*, *broken authentication*, *sensitive data exposure*, *xml external entities*, *broken access control*, *security misconfiguration*, *cross-site scripting*, *insecure deserialization*, *using components with known vulnerabilities* e *Insufficient Logging&Monitoring* [OWASP 2017]. Algumas vulnerabilidades citadas acontecem por equívocos na configuração de servidores web e servidores de banco de dados, como a vulnerabilidade *Insufficient Logging&Monitoring*. Porém, a grande maioria das vulnerabilidades possuem como seus causadores os paradigmas de programação utilizados, como nas vulnerabilidades *injection*, *data exposure* e *cross-site scripting*.

Este artigo pretende apontar as práticas de desenvolvimento *web* comuns que geram falhas de segurança, apresentando qual foi a vulnerabilidade gerada e o motivo do ocorrido. Para sustentar a afirmação, serão apresentadas algumas formas de explorar tal vulnerabilidade. Logo em seguida alguns exemplos de implementar a mesma funcionalidade de maneira segura serão mostrados e será refeita a tentativa de exploração da vulnerabilidade para verificar se a falha foi extinta da aplicação.

Para o desenvolvimento deste trabalho foi realizado um levamento bibliográfico sobre os principais paradigmas que causam falhas de segurança, bem como, estudo das ferramentas para implementação dos exemplos sendo elas, o Php e Mysql para o desenvolvimento e *Nginx* como servidor *web*. Por fim, para consolidação dos testes de segurança serão usadas as ferramentas Burp Suite, SQLMap, BeEF e w3af.

Referências

OWASP (2017). OWASP Top 10 Application Security Risks – OWASP. https://www.owasp.org/index.php/Top_102017_Top_10.

TANENBAUM, Andrew S.; WETHERALL, D. J. Redes de Computadores. 5 ed. São Paulo: Pearson, 2011.