Universitatea Tehnică a Moldovei

MODEL DE SECURIZARE A APLICAȚIILOR WEB

WEB APPLICATION SECURITY MODEL

Masterand:

Napadaică Mihail

Conducător:

lector universitar: Bulai Rodica

Chişinău 2020

Ministerul Educației Culturii și Cercetării al Republicii Moldova

Universitatea Tehnică a Moldovei

Facultatea Calculatoare, Informatică și Microelectronică

Departamentul Ingineria Software și Automatică

Admis la susținere

Șef departament: conf. univ., dr.

19 decembrie 2019

Model de securizare a aplicațiilor web

Teza de master în

Securitate Informațională

Masterand: Napadaica Mihail

Conducător: Bulai Rodica

Chișinău – 2020

Rezumat

Acest proiect vine ca o analiză a nivelului de securitate pentru aplicațiile web la momentul actual. Cu acest scop au fost analizat raportul oferit de OWASP referitor la topul vulnerabilităților în domeniul tehnologiei web și altă documentație oferită de OWASP și specialiștii din domeniu.

A fost elaborată și o aplicație care analizează și cu ajutorul cărui putem detecta vulnerabilitățile dintr-o aplicație web.

Primul capitol din această lucrare reprezintă o analiză a domeniului de studiu. Totul începe cu evoluția și dezvoltarea cu pași mari a tehnologiei web la momentul actual. În acest capitol este descris arhitectura aplicațiilor web și securitatea acestor aplicații.

În capitolul doi este analizat proiectul OWASP și riscurile de securitate. Au fost analizate mai amănunțit cele zece vulnerabilități raportate de OWASP începând de la modul în care acestea apar și sunt exploatate și terminând cu metodele de protecție.

Capitolul trei este în general despre aplicația elaborată. În acest capitol este descris limbajul de programare ales, tehnologiile utilizate la interfața acestuia dar și la realizare.

Și la sfirșit capitolul patru redă un mic ghid de utilizare a sistemului elaborat, diagrame pentru descrierea comportamentală a sistemului folosind limbajul de modelare UML, dar și explicația funcționalității aplicației prin secvențe de cod.

Abstract

This project comes as an analysis of the security level for web applications at the moment. To this scopes was analyzed the report provided by OWASP regarding the top vulnerabilities in the area of web technology and other documentation provided by OWASP an other specialists in the area.

Also was developed an application with which we can analyzes and detect the vulnerabilities of a web application.

The first chapter of this work is an analysis of the area of study. It start whit the evolution and big step development of the web technology at the moment. This chapter describes the architecture of the web application and the security of these application.

In chapter two was analyzed the OWASP project and security risk. The ten vulnerabilities reported by OWASP were analyzed in more detail starting from the way they appear and are exploited and ending with the protection methods.

Chapter three is generally about the developed application. This chapter describes the chosen programming language, the technologies used at its interface but also in the realization.

And finally, the fourth chapter gives a small guide for the use of the elaborated system, diagrams for the behavioral description of the system using the UML modeling language, but also the explanation of the functionality of the application through code sequences.

Cuprins:

INTRODUCERE

Astăzi Internetul este cel mai puternic instrument din lume. Internetul este o colecție de diverse servicii și resurse, de care este nevoie atât în viața profesională și cele de studii, cât și zi de zi. Cu toate acestea, ca orice inovație în domeniul științei și tehnologiei, internetul are propriile avantaje și dezavantaje. Printre primele avantaje cele mai importante ale internetului este comunicarea. Datorită Internetului oamenii pot comunica într-o fracțiune de secundă cu o altă persoană care este la mii de km distanță. Un alt mare plus oferit de către internet este nivelul de informații oferit. Orice tip de informații despre orice subiect sunt disponibile pe internet. Există o cantitate uriașă de informații, de la legislație și servicii guvernamentale, la informații despre piață, și idei noi. Este foarte util pentru oamenii care studiaza la o instituție sau pur și simplu doresc să învețe lucruri noi, din alte domenii.

Divertismentul este un alt motiv popular care explică popularitatea internetului. Descărcarea jocurilor și muzicii, vizitarea camerilor de chat sau doar navigarea pe web sunt câteva din plăcerile pe care oamenii le-au descoperit. Oamenii navigând pe internet găsesc numeroase lucruri ca muzică, hobby-uri noi, știri și multe altele.

Odată cu creșterea popularității internetului a crescut și numărul aplicațiilor web. O aplicație web este un program care rulează într-o arhitectură client-server folosind tehnologiile deschise World Wide Web. Spre deosebire de aplicațiile desktop tradiționale, care sunt lansate de sistemul clientului de operare, aplicațiile web trebuie accesate printr-un browser web.

Aplicațiile web au mai multe avantaje față de aplicațiile desktop. Deoarece utilizează browserele web, dezvoltatorii nu trebuie să dezvolte aplicații web pentru mai multe aplicații, nu trebuie să distribuie actualizări de software utilizatorilor atunci când aplicația web este actualizată. Prin actualizarea aplicației pe server, toți utilizatorii vor avea versiunea actualizată.

Din punct de vedere a utilizatorului, o aplicație web poate oferi o interfață utilizator mai consistentă pe mai multe platforme, deoarece aspectul depinde de browser, mai degrabă decât de sistemul de operare. În plus, datele introduse într-o aplicație web sunt procesate și salvate de la distanță. Acest lucru permite accesarea aceleași date de pe mai multe dispozitive, mai degrabă decât transferul fișierelor între sisteme de calculator.

Astfel, fie că discutăm de baze de date, de unelte și medii le folosim software pe care, de algoritmi, funcții și proceduri pe care le implementăm, sau de standardele care trebuie să fie respectate pentru dezvoltarea aplcațiilor web - toate acestea țin de tehnologiile web.

Conform statisticii cele mai multe atacuri vizează pe aplicațiile web. Adesea aceste atacuri nu fac nimic decât exploatarea vulnerabilităților atenuate în codul aplicațiilor. Cele mai frecvente probleme

găsite în codul aplicațiilor web sunt: forging și cross-site scripting. Vulnerabilitatea aplicațiilor se datorează mai multor factori:

- Securizarea scăzută a arhitecturii web, care permite instalarea și executarea în interiorul browser-ului a programului malware.
- Creșterea complexității, care ascunde bug-urile, deoarece codul devine mai complex și conține o sintaxă mai obscură, ceea ce face mai dificil detectarea vulnerabilităților existente în aplicația web elaborată.
- Accesibilitatea publică a internetului, aceasta fiind și cel mai mare avantaj a aplicațiilor web dar și în același timp și permite atacatorilor externi să creeze o cale către datele confidențiale și mediile IT dintr-o companie.

Fiecare piață de tehnologii are nevoie de o sursă imparțială de informații privind cele mai bune practici, precum și de o comunitate activă care pledează pentru standarde deschise. În spațiul de securitate web, unul dintre aceste grupuri este Proiectul Open Web Application Security Project (OWASP).

OWASP este o comunitate deschisă dedicată pentru a permite organizațiilor să conceapă, să dezvolte, să achiziționeze, să opereze și să mențină aplicații care pot fi de încredere.

Fiecare 4 ani Open Web Application Security Project elaborează un raport de securitate, privind cele mai raspandite atacturi asupra aplicațiilor web.

Raportul se bazeaza pe informație din mai mult de 40 de înregistrări de date primite de la companii specializate în securitatea aplicațiilor, cu datele care acoperă vulnerabilitățile colectate de la sute de organizații și peste 100.000 de aplicații web și API implementate.

OWASP raportează fiecare risc în funcție de exploatabilitatea acestuia, de prevalența slăbiciunilor, de detectabilitatea slăbiciunilor și de impactul tehnic. Aceste riscuri se schimbă mereu.

CONCLUZIE

În urma elaborării acestei lucrări a fost realizată o analiză a domeniului de studiu pentru ca mai apoi pe baza acestei analize să fie elaborată teza de masterat. În special a fost pus accentul pe aplicațiile web studiind arhitectura și realizarea acestora. Aplicațiile web au un șir de puncte forte prin care se deosebesc și sunt mai presus ca celelalte tipuri de aplicații. Ca de exemplu accesibilitatea, disponibilitatea. Dar în același timp datorită modului de funcționare a acestora apar un șir de vulnerabilități care sunt descoperite zi de zi, și care pot aduce daune proprietarilor acestor aplicații și nu numai dar în unele cazuri pot avea de suferit și utilizatorii obișnuiți.

O dată cu dezvoltarea tehnologiilor noi pentru crearea și securizarea aplicațiilor web se detectează și noi vulnerabilități ce pot afecta sistemele.

Ca scop al acestei lucrări și pe viitor a tezei de masterat este pus studierea celor mai critice vulnerabilități asupra aplicațiilor web. Implementarea cu ajutorul rapoartelor primite și analizate de la Open Web Application Security Project atacuri asupra securității aplicațiilor.

Analizând punctele slabe ale aplicațiilor web ne vom putea face o părere ce trebuie pe viitor de îmbunătățit în această tehnologie sau cel puțin la elaborarea unor astfel de aplicații evitarea acestor vulnerabilități și crearea unei aplicații securizate.

La etapa actuală a fost identificată lista cu top 10 vulnerabilităților oferite de OWASP și analizate fiecare în parte. Pentru toate vulnerabilitățile identificate au fost analizate esența problemei, cauza din care apar vulnerabilitățile, cât și soluțiile posibile pentru evitare și prevenire a acestora.

A fost elaborată o aplicație folosind libajul Java pentru detectarea vulnerabilităților într-o aplicație web. Această aplicație are o interfață user friendly care este intuitivă pentru utilizator.

BIBLIOGRAFIE

1. Aplicații web [Resursă electronică] Regim de acces: https://www.webdesign-galati.ro/blog/ce-este-aplicatia-web
2. Internet și arhitectura aplicațiilor web [Resursă electronică]Regim de acces: http://www.creeaza.com/referate/informatica/internet/INTERNET-si-arhitectura-de-baz875.php
3. Web Architecture [Resursă electronică] Regim de acces: https://en.ryte.com/wiki/Web_Architecture
4. ON Security of Web Application [Resursă electronică] Regim de acces: http://revistaie.ase.ro/content/34/Buraga.pdf
5. Web App Security [Resursă electronică] Regim de acces: https://www.acunetix.com/websitesecurity/web-app-security/
6. The Open Web Application Security Project [Resursă electronică] Regim de acces: https://www.owasp.org/index.php/Main_Page
7. SQL Injection [Resursă electronică] Regim de acces: https://www.php.net/manual/ro/security.database.sql-injection.php
8. Web Application Firewall (WAF) [Resursă electronică] Regim de acces: https://www.cloudflare.com/learning/ddos/glossary/web-application-firewall-waf/
9. SQL Injection [Resursă electronică] Regim de acces: https://portswigger.net/web-security/sql-injection
10. Broken Authentication [Resursă electronică] Regim de acces: https://resources.infosecinstitute.com/2017-owasp-a2-update-broken-authentication/#gref
11. The OWASP TOP 10: Sensitive Data Exposure [Resursă electronică] Regim de acces: https://www.sitelock.com/blog/owasp-top-10-sensitive-data-exposure/
12. XML External Entity(XXE) Processing [Resursă electronică] Regim de acces: https://www.owasp.org/index.php/XML_External_Entity_(XXE)_Processing
13. Broken Acces Control [Resursă electronică] Regim de acces: https://www.owasp.org/index.php/Broken_Access_Control
14. Security misconfiguration [Resursă electronică] Regim de acces: https://subscription.packtpub.com/book/networking_and_servers/9781788627252/7/ch07lvl1sec56/security-misconfiguration
15. Cross-site Scripting (XSS) [Resursă electronică] Regim de acces: https://www.owasp.org/index.php/Cross-site_Scripting_(XSS)

16. Insecure deserialization [Resursă electronică] Regim de acces: https://www.acunetix.com/blog/articles/what-is-insecure-deserialization/
17. Using Components with Known Vulnerabilities [Resursă electronică] Regim de acces: https://blog.secureideas.com/2019/07/using-components-with-known-vulnerabilities.html
18. How to prevent Insufficient Logging and Monitoring [Resursă electronică] Regim de acces: https://medium.com/@javan.rasokat/owasp-appsensor-logging-and-monitoring-2518712ee0fe
19. Introducere în limbajul de programare Java [Resursă electronică], Regim de acces: http://www.cs.ubbcluj.ro/~vcioban/Geografie/MasterGeoPOO/Curs/CursJava.docx
20. Curs Practic de Java [Resursă electronică], Regim de acces: http://web.info.uvt.ro/~iordan/P_III/Cristian_Frasinaru-Curs_practic_de_Java.pdf
21. JavaFX Overview [Resursă electronică], Regim de acces; https://docs.oracle.com/javafx/2/overview/jfxpub-overview.html
22. JavaFX Scene Builder [Resursă electronică], Regim de acces: http://www.oracle.com/technetwork/java/javase/downloads/javafxscenebuilder-info2157684.html
23. Zap Documentation [Resursă electronică], Regim de acces: https://www.zaproxy.org/docs/api/?java#introduction