Analiza securității pentru aplicație

1. Prezentare Generală și Principalele Riscuri de Securitate

Acest document descrie principalele riscuri de securitate asociate cu o aplicație web **.NET MVC** și oferă tactici pentru a le aborda eficient.

Cele mai frecvente atacuri asupra aplicațiilor web sunt cele din **OWASP Top 10**, care includ:

- **SQL Injection**: Un atac în care un atacator injectează comenzi SQL malițioase într-o interogare pentru a accesa sau modifica baza de date.
    - **Prevenire**: Utilizați **Entity Framework** sau **Dapper** cu interogări parametrizate pentru a preveni inserarea de cod SQL malițios.

- **Cross-Site Scripting (XSS)**: Permite atacatorilor să introducă scripturi malițioase care sunt executate în browserul utilizatorului.
    - **Prevenire**: Folosiți **Razor Encoding (@Html.Encode()), Content Security Policy (CSP)** pentru a preveni executarea scripturilor nesigure.

- **Cross-Site Request Forgery (CSRF)**: Un atac care forțează un utilizator autentificat să execute acțiuni nedorite.
    - **Prevenire**: Implementați **@Html.AntiForgeryToken()** și verificați token-urile în backend folosind **ASP.NET Core CSRF Middleware**.

- **Probleme de autentificare și autorizare**: Atacatorii pot exploata conturi slabe sau configurări greșite pentru a obține acces neautorizat.
    - **Prevenire**: Utilizați **ASP.NET Identity** pentru autentificare și **OAuth 2.0** pentru acces securizat.

- **Expunerea nesecurizată a API-urilor**: API-urile nesecurizate pot oferi acces la date sensibile.
    - **Prevenire**: Protejați API-urile folosind **JWT (JSON Web Token)** și implementați **rate limiting** în **ASP.NET Core**.
- **Expunerea datelor și probleme de criptare**: Datele sensibile pot fi expuse dacă nu sunt criptate corespunzător.
    - **Prevenire**: Folosiți **ASP.NET Data Protection API** și asigurați-vă că parolele sunt hasherate folosind **bcrypt** sau **PBKDF2**.

Aceste riscuri trebuie abordate prin implementarea de practici de codare sigură și configurare adecvată a aplicației.

2. Tactici pentru Abordarea Riscurilor de Securitate

a. Practici de Codare Sigură

- Urmați recomandările **OWASP Top 10**.
- Realizați **revizuiri regulate ale codului** și analize statice.
- Evitați utilizarea **credențialelor hardcodate** și folosiți **Azure Key Vault** sau **AWS Secrets Manager**.

b. Configurare Securizată

- Utilizați **HTTPS (SSL/TLS)** pentru tot traficul web.
- Dezactivați **mesajele de eroare verbose** în producție.
- Restricționați **politicile CORS** la origini de încredere utilizând **ASP.NET CORS Middleware**.

c. Testare și Monitorizare a Securității

- Implementați **scanări automate de securitate** cu **SonarQube, Snyk și OWASP ZAP**.
- Efectuați **testare de penetrare** periodic.
- Configurați **logare și monitorizare** cu **Serilog, ELK Stack, și Application Insights**.

d. Gestionarea Dependențelor

- Actualizați regulat **pachetele NuGet**.
- Monitorizați vulnerabilitățile în bibliotecile terțe utilizând **OWASP Dependency Check** și **NuGet Audit**.

e. Cele Mai Bune Practici pentru Controlul Accesului

- Aplicați principiul **celui mai mic privilegiu**.
- Securizați **zonele administrative** cu **autentificare suplimentară MFA (Multi-Factor Authentication)**.
- Logați și monitorizați **încercările eșuate de autentificare** folosind **Azure Monitor și Splunk**.

3. Concluzie

Implementarea acestor tactici de securitate va contribui la reducerea riscurilor și la creșterea rezilienței aplicației **.NET MVC** împotriva atacurilor. Auditările regulate de securitate, respectarea principiilor de codare sigură și adoptarea celor mai bune practici din industrie sunt esențiale pentru menținerea unui sistem sigur.

Referințe

- OWASP Top 10 Riscuri de Securitate
- Microsoft - Cele Mai Bune Practici de Securitate