



Bezbednost sistema

Bezbednost u sistemima elektronskog plaćanja: KT2

Kontekst

Razvoj bezbednog softvera podrazumeva ugrađivanje bezbednosnih kontrola u softver tokom njegovog razvoja, prateći koncept *built-in security*. Trošak ugrađivanja bezbednosti na ovaj način je najmanji i, po pravilu, bezbednost je implementirana najkvalitetnije, jer se koncipira bezbedan dizajn koji će kod poštovati, umesto da se bezbednost “budži” i prilagođava već napisanom kodu.

Cilj zadatka

Kao glavni rezultat ovog zadatka, svaki student treba da se upozna sa najčešćim mehanizmima za zaštitu veb-baziranog softvera, što uključuje:

- Načine integracije datih kontrola u dizajn softvera – gde će se kontrola iskoristiti;
- Implementacije datih kontrola za odabrani skup jezika i tehnologija u kojim će se konačan softver realizovati – kako kontrola izgleda.

Specifikacija

Kroz ovaj zadatak, studenti treba da razviju niz *proof-of-concept* rešenja za bezbednosne kontrole, koje će potom integrisati u ostatak projekta kako ga budu razvijali za konačnu odbranu.

Konkretno, neophodno je implementirati sledeće bezbednosne mehanizme:

- **HTTPS komunikacija, gde treba demonstrirati:**
 - **Bezbednu komunikaciju između browser-a i servera;**
- **Validacija podataka koji stižu na aplikacije, gde je neophodno:**
 - **Sprečiti relevantne Injection napade;**
 - **Sprečiti XSS napade;**
 - **Izvršiti validaciju podataka za proizvoljne podatke, koristeći kriterijume validacije definisane po najboljim praksama za pisanje bezbednog koda;**
- **Autentifikacije i kontrola pristupa, gde je neophodno:**
 - **Omogućiti registraciju i prijavu korisnika na sistem (kao i mehanizme za potvrdu naloga i oporavak lozinke).**
 - **Kontrolisanje pristupa *endpoint*-ima po RBAC modelu;**
 - **Kontrolisanje pristupa datotekama i direktorijuma od strane aplikacije (ACL);**
 - **Testirati i demonstrirati da sve kontrole pristupa rade (pozitivan i negativan ishod).**

Napomene

- 1) **Potrebno je implementirati funkcionalnosti tek toliko da se podrži smislena demonstracija bezbednosnih kontrola.**
- 2) **Dozvoljeno je da se jedan član tim posveti jednoj tački, no neophodno je da svaki član tima bude svestan kako svaki segment (pa i onaj koji nije implementirao) radi i da razume celu priču na srednjem nivou detalja (ne nužno na liniji koda ili konfiguracije, ali ne samo ni na visokom nivou koncepta).**
- 3) **Pojedine tačke je moguće rešiti uz pomoć tehnologije za implementaciju softvera (jezika, radnog okvira) ili alata – ovo je dozvoljeno, no neophodno je razumeti kako tehnologija rešava problem i o čemu treba voditi računa da se pružena bezbednosna kontrola „ne pokvari“.**
- 4) **Za potrebe kontrolne tačke stvari poput konfiguracije ACL-a i testiranja mogu da se rade manuelno.**
- 5) **Prilikom istraživanja i implementacije kontrola, neophodno je voditi računa o bezbednoj konfiguraciji kontrole – skup parametara koje kontrola ima i njihova *best practice* vrednost.**