Top 10 Web Application Security Risks

OWASP

- ## Injection

  Izvršena je validacija podataka, kako na frontend-u, tako i na backend-u. Za validaciju podataka na frontend-u korišćeni su Angular Validatori, dok je na backend-u validacija implementirana upotrebom javax anotacija u okviru DTO objekata koji se preuzimaju u kontrolerima kao RequestBody.
  Što se tiče SQL Injection-a, Hibernate interno koristi prepared statements. Prepared statements obezbeđuju to da napadač ne može da promeni namenu upita,čak i ako pokušava u upit da ugradi maliciozni SQL.
  Što se NoSQL Injection-a tiče, korišćena je MongoDB baza podataka, te je ugrađivanje malicioznog koda moguće ukoliko postoji upotreba group i $where operatora, kao i MapReduce operacija, te ove operatore i operacije nismo koristili.

- ## Broken Authentication

  Napomena: Za rukovanje JWT tokenima, korisnicima i sesijama zadužen je Auth0.
  Broken Authentication podrazumeva ranjivosti kao što su: dozvola brute force napada, dozvola korišćenja slabih lozinki, čuvanje lozinki u plain formatu, prikazivanje session id-a,...
  Neki od napada koji su mogući su:
  - Session Hijacking
  - Session ID Url Rewriting – session ID je prikazan u okviru URL-a.
  - Session Fixation – Ne postoji rotacija session ID-a nakon što se korisnik uloguje.
  - Credential Stuffing
  - Password Spraying – Korišćenje slabih najčešćih lozinki kako bi se napadač ulogovao na korisnički nalog.

  Auth0 ne čuva lozinke u plain formatu, lozinke su heširane, za heširanje korišćen je Bcrypt. Podešena su različita ograničenja za lozinke – lozinka ne može da bude jedna od lozinka sa spiska 10000 najčešćih, mora da bude minimum 8 karakera dužine, treba da zadovoljava barem 3 od sledeća 4 kriterijuma – sadrži veliko slovo, sadrži malo slovo, sadrži specijalne karaktere, sadrži brojeve. Takođe, prilikom promene lozinke nije moguće postaviti onu koja je korišćena u prethodna tri puta
  Uključena je i zaštita protiv brute force napada (10 uzastopnih neuspešnih logovanja od strane istog korisnika i sa iste IP adrese, 100 neuspešnih pokušaja logovanja sa iste IP adrese u poslednjih 24 sata).
  Što se JWT tokena tiče, JWT token je potpisan RSAwithSHA256 algoritmom, te se čuva u okviru behaviour subject-a. Dužina trajanja tokena je podešena na 10 sati. U payload-u je podešen audience, pa se tako na backend-u proverava da li JWT token ima odgovarajući audience, u slučaju da nema, svaki zahtev će biti odbijen.
  Ukoliko korisnik nije aktivan 24 sata, sesija mu se invalidira.Uz to, bez obzira na aktivnost, ponovno logovanje se traži nakon 7 dana.

- ## Sensitive data exposure

  Osetljivi podaci koji su u stanju mirovanja su enkriptovani (za kriptovanje korišćen je simetrični algoritam, AES256) i mogu ih pregledati samo korisnici koji imaju odgovarajuća prava pristupa. Podaci koji se prenose preko mreže su samo oni koji su neophodni i zaštićeni su HTTPS-om.

- ## XML External Entities

  U projektu se ne radi sa podacima u XML formatu.

- ## Broken Access Control

  Definisane su tri role – admin, superadmin i doktor. Svakoj od ovih rola dodeljene su adekvatne permisije. Svaki endpoint u projektu je zaštićen, odnosno, podešene su potrebne permisije koje korisnik mora da ima da bi pristupio endpoint-u. Angular rute su isto tako zaštićene odgovarajućim permisijama.

- ## Security Misconfiguration

  Ne koriste se nalozi koji imaju default-no korisničko ime i lozinku. Ne postoje nepotrebni feature-i i web stranice. Poruke o greškama ne otkrivaju senzitivne podatke, niti daju previše informacija.

- ## Cross Site Scripting (XSS)

  Angular je korišćen za razvoj klijentskog dela aplikacija. Angular poseduje ugrađenu zaštitu od XSS napada. Na koji način? Da bi sprečio XSS napade, Angular sve vrednosti tretira kao untrusted. Vrši se inspekcija untrusted vrednosti, koja će vrednost pretvoriti u sigurnu i ubaciti je u DOM stablo.
  Za serverski deo aplikacije korišćen je Spring Boot, koji, takođe, štiti od XSS napada. X-XSS Protection header signalizira browseru da blokira sve što liči na XSS. Spring Security automatski dodaje pomenuti header u response. Ovo smo podesili u okviru Spring Security konfiguracione klase. Na ovaj način browser ne renderuje ukoliko detektuje XSS. Nažalost, neki browseri ne koriste ovaj header, te se koriste Content-Security-Policy (CSP) filteri.

- ## Insecure Deserialization

  Na backend-u sav saobraćaj koji bi trebao da stigne je validiran (u vidu anotacija za atribute DTO-a ili u okviru metode), na frontend-u validirane su sve forme. Svaki potpisan saobraćaj podrazumeva validiranje samog potpisa.

- ## Using Components With Known Vulnerabilities

  Sve komponente, poput biblioteka i framework-a, koje su korišćene u projektu su stabilne i najnovije verzije.

- ## Insufficient Logging & Monitoring

  Admin sistema je obavešten o događajima u sistemu. Prikazani su mu logovi kako iz same aplikacije, tako i logovi za autentifikaciju sa Auth0-a.