**10. Unvalidated redirects and forwards** (Dejan)

**RESENO**

Klasican phishing napad, ne moze se odbraniti ako je sam korisnik budala. Ali za redirektovanje treba validirati inpute, ali mi ne radimo redirekciju na serveru (imamo samo return podatke). Whitelist da se proveri url, radjeno nesto za XSS napad, proveriti tamo mozda SPRING ima neku zastitu.

Da li je state.go dovoljna zastita za ovaj problem posto moze da se promeni putanja.

**9. Using components with known vulnerabilities** (Dejan)

Napisati koje komponente koristimo i proveriti da li su bezbedne.

**8. Cross-site request forgery** (Dejan)

Da li JWT i HTTPS resavaju ovaj problem.

**7. Missing function level access control** (Marko)

JWT ima rolu po kojoj dozvoljavamo korisniku odredjene radnje, proveravamo i u springu i u angularu koja je rola.

**6. Sensitive data exposure** (Aleksandar)

TLS(HTTPS), digitalno potpisicanje, enkripcija podataka uradjeno.

**5. Security misconfiguration**

Da li ovo treba da se resava?

Ne treba da se radi!!!

**4. Insecure direct object references** (Aleksandar)

**RESENO**

Reference korisnika su sakrivene u JWT, ali za amandmane i propise imamo zato sto nam traze iz XML-a.

**3. Cross-site scripting** (Marko)

Odraditi whitelist za sve inpute da nam se ne bi desio neki <script> tag. Jsoup ne dozvoljava <script> i href, da li je to dovoljno za XSS napad? (UserControler, login)

**2. Broken authentication and session management** (Drazen)

Koristimo TLS i JWT, a ne koristimo sesiju tako da ne moze da se ukrade.

Napraviti jacinu password-a.

**1. Injection** (Drazen)

Proveravati sve inpute i paziti kako se rukuje sa bazom. Pitati da li se resava sa Jsoup kao i XSS i HTTPS kao i validacijom inputa?