Laborator 10

▼ **Exercitiul 1**

1. Ca să analizați/testați securitatea aplicației, ajută să gândiți ca un atacator. **(A)**
2. Pentru că sunt foarte multe, din punct de vedere al logicii/design-ului aplicației, nu încercați să acoperiți toate cazurile posibile pentru a preveni un comportament neașteptat. **(A)**
3. Întotdeauna validați câmpurile de input, atât ca format (tip de date, protejare împotriva SQL injection, etc.) dar și ca valori (dimensiuni, valori minime/maxime, verificări între diferite câmpuri de input; ex. data de început a unei activități anterioară datei de final, prețurile să aibă valori pozitive, etc.) **(A)**
4. Aveți în vedere vulnerabilități de tip buffer overflow. **(A)**
5. În general nu e o practică bună să stocați log-uri, pentru că ocupă spațiu și cresc timpul de așteptare al utilizatorului. **(F)**
6. Oferiți cât mai multe detalii posibile utilizatorilor când eșuează autentificarea prin username și parolă sau când implementați mecanisme de recuperare a parolei, pentru a facilita accesul acestora (spre exemplu menționați „Adresa de e-mail nu corespunde unui cont activ” la încercarea de a recupera parola prin e-mail). **(F)**
7. Nu rețineți parole în clar. **(A)**
8. Hardcodați parole în cod **(F)**

▼ **Exercitiul 2**

- Validare pentru upper si lowercase
- Validare pentru lungimea parolei
- Salvarea hashului si saltului
- Validare prin confirmarea parolei ?

▼ **Exercitiul 3**

- Mesaj generic de eroare la login
- Parola are field de tip password

- Paginile nu pot fi accesate decat dupa login