Øvelse 17 - Opsamlingsøvelser med Juice Shop

Objective

Efter denne opsamlingsøvelse vil du kunne: - Genanvende dine færdigheder i API-sikkerhed og sårbarhedsanalyse - Dokumentere og reproducere sårbarheder i form af test cases - Arbejde mere selvstændigt med opgaveformuleringer og værktøjer som Burp Suite og Postman - Gennemføre og reflektere over øvelser fra PortSwigger Academy Learning Path

---

Information

Formålet med øvelserne er at repetere de færdigheder, der er blevet tillært fra tidligere undervisningsgange.

Beskrivelserne i øvelserne er mere kortfattede end tidligere og giver kun en beskrivelse af målet for gennemførelsen af øvelsen, men ikke de nødvendige skridt.

Øvelse 1–5 er repetitionsøvelser der ligner tidligere crAPI-øvelser og udføres i Juice Shop-applikationen.
Øvelse 6 (samt valgfri øvelse 7) er øvelser, der udarbejdes på webportalen *PortSwigger Academy*.

PortSwigger Academy-øvelserne er udarbejdet af teamet bag Burp Suite.
Du kan tilmelde dig PortSwigger Academy her.

Disse øvelser er mindre guidede end tidligere, men fremgangsmåden er den samme som de tidligere øvelser med crAPI.

Når du arbejder med opgaverne på PortSwigger Academy, kan det være en fordel at proxy din browser gennem Burp Suite (f.eks. Chrome eller Firefox).
Dette kan gøres på flere måder. For eksempel er her en guide til anvendelse af proxy med *Firefox*.
Alternativt kan du bruge browserudvidelsen *FoxyProxy* til nemt at skifte mellem forskellige proxy-konfigurationer. En guide findes her.
Du kan naturligvis også bruge Burp Suites indbyggede browser eller OWASP ZAP.

I øvelse 5, skal du skrive en *test case* for hver af de tidligere øvelser samt repetitionsøvelserne.

Test cases har to formål: - Du kan senere vende tilbage og se, hvordan du gennemførte øvelsen - Du kan kommunikere og dokumentere sårbarheder overfor andre (f.eks. udviklere)

Dette er en meget kort introduktion til test cases, og vi vender tilbage til dem i næste uge

Eksempel på en test case (BOLA)

Trin	Handling
1	Opret bruger **A** via registreringsformularen i crAPI.
2	Autentificer som bruger **A** og tilføj et køretøj.
3	Brug Burp Suite til at opsnappe en **GET**-forespørgsel til `/api/vehicles/<id>`.
4	Notér køretøjets **id** fra URL'en.
5	Opret bruger **B** via samme metode.
6	Autentificer som bruger **B**.
7	Åbn Burp **Repeater** og genskab forespørgslen med bruger A's `<id>`.
8	Send forespørgslen og observer svaret.

Forventet resultat:
Forespørgslen bør returnere `403 Forbidden` eller `404 Not Found`. Hvis svaret er `200 OK` med bruger A's data, er der tale om en **Broken Object Level Authorization (BOLA)**-sårbarhed.

Du kan med fordel dokumentere dine test cases i din GitLab Pages, men du er velkommen til at bruge andre metoder.

Instruktioner

Repetition

1. Gennemfør alle tidligere øvelser, som du endnu ikke har færdiggjort.
2. Skriv en test case for hver tidligere øvelse. (Skridt-for-skridt hvordan du tester sårbarheden)

Juice Shop

1. Gennemfør Juice Shop BOLA og EDE-øvelserne
   Link til øvelserne
2. Gennemfør Juice Shop Active Reconnaissance
   Link til øvelserne
3. Skriv en test case for hver Juice Shop-øvelse.
4. Gennemfør de ekstra Juiceshop Bola øvelser
   Link til øvelserne

PortSwigger Academy

1. Gennemfør Labs 1, 2 og 3 i "API Testing" Learning Path:
   PortSwigger Learning Paths
2. *(Valgfrit)* Gennemfør hele API Testing Learning Path'en.

Reflektions spørgsmål

1. Hvilke typer sårbarheder var lettest og sværest at identificere? Hvorfor?
2. Hvilken rolle spiller værktøjer som Burp Suite i dit arbejde med sårbarhedsanalyse?
3. Hvad lærte du af at skrive dine egne test cases – både teknisk og kommunikativt?
4. **Etisk refleksion:** Hvad er dit ansvar, hvis du opdager en sårbarhed i en offentlig eller professionel API?

Last update: 2026-09-07 05:23:23