שיעור 10 - פרוייקט גליטר

**דרישות הפרוייקט לקראת ההגנה**

במסמך זה נפרט את כלל הדרישות לפרוייקט "גליטר", ויותר בפירוט את הדרישות מכלי האולר אותו תציגו במהלך ההגנה.

**חלק א' – דרישות הפרוייקט**

במעמד ההגנה נבדוק את שלושת חלקי הפרוייקט:

* מספר החולשות שנמצאו והושמשו (חולשות של שלב א' + אתגרים של שלב ב')
* מסמך מחקר האבטחה המלא
* כלי האולר

**מהן החולשות שצריך למצוא?**

* **בשלב א'** למדנו למצוא חולשות אשר אנחנו חוקרים לפי הבקשה. עלינו לבדוק כמה שיותר בקשות באתר או באפליקציה, ולנסות להמציא תרחישים הקשורים לפרמטר כלשהו, או לעצם הבקשה. יש לעבוד יחד עם מסמך האבטחה ולתעד כל סייקל.

כל החולשות של שלב א' בנויות על מחקר ספציפי של בקשה ואינן דורשות שילוב של מספר תהליכים או מחקר ארוך מאוד.

* **בשלב ב'** היכרנו את 5 האתגרים אשר החברה פתחה לציבור הרחב במסגרת Bug Bounty:
  + **אתגר הלוגאין –** חולשות הקשורות למנגנון הלוגאין של האפליקציה.
  + **אתגר הססמא –** חולשות המאפשרות השגת ססמא של משתמשים (שימו לב, זה אתגר שאינו קשור לאתגר הלוגאין).
  + **אתגר העוגיה –** חולשות הקשורות להשגת עוגייה של משתמש אחר.
  + **אתגר הפרטיות –** חולשות הקשורות להשגת מידע על פעולות פרטיות שביצעו משתמשים אחרים (כאלה שלא אמורות להיות חשופות).
  + **אתגר XSRF –** חולשות הקשורות להשמשת XSRF (בעזרת XSS, בדומה למה עשינו בפעילות פסח).

אתגרים אלו מורכבים יותר ודורשים מאיתנו מחקר של תהליכים באתר (או במילים אחרות, לא נמצא אותן על ידי שינוי של פרמטר וזהו).

**כמה חולשות צריך למצוא?**

* **בשלב א'** יש חולשות רבות. ככל שתמצאו יותר – הרי זה משובח. שימו לב שעליכם למצוא חולשות מכמה שיותר נק' כניסה שונות.
* **בשלב ב'** ישנם 5 אתגרים מורכבים יותר. מומלץ לפתור לפחות 3 מהם, וכמה שיותר – הרי זה משובח.

**חלק ב' – הוראות לכלי האולר** 

מטרתו של כלי האולר היא להדגים את היכולות שנרכשו במהלך מחקר האבטחה של גליטר, כלומר על האולר לרכז את כל קודי ה-POC (הוכחת מימוש) של החולשות שגיליתם בגליטר. כעת נפרט על הדרישות הכלליות מכלי האולר.

**חווית משתמש**

האולר צריך להיות נוח לשימוש לאדם שאינו מתמצא בקוד, כך שיוכל להוות הדגמה חיה לעובדי חברת גליטר. בעת הכניסה לתוכנה, יתבקש המשתמש לבצע לוגאין עם שם וססמא (יתקבלו בקלט ולא בקבועים).

לאחר מכן יוצג למשתמש תפריט של מגוון החולשות האפשריות.  
במידת הצורך, בכל חולשה יתבצע קלט נוסף, אך נסו שהקלט יהיה כמה שיותר ידידותי ולא ידרוש ידע נוסף. למשל, אם החולשה שלכם קשורה לפוסט מסוים, הדרך הפשוטה תהיה לבקש את ה-ID שלו מהמשתמש. אבל מכיוון שמשתמש ממוצע לא מכיר ID, ניתן להציג לו את רשימת הגליטים הקיימים עם המזהים שלהם, מה שיקל עליו את הבחירה. או, כדוגמא נוספת, אם החולשה קשורה למשתמש אחר, נסו לבקש את השם המוצג שלו ולא את ה-ID שלו.

ככל שהאולר יהיה יותר ידידותי למשתמש תרוויחו נק' נוספות.

**מבנה כללי**

על הפרוייקט לכלול קובץ main בשם swissknife.py וספרייה (מודול) בשם glitter. הספריה תרכז את כל הפונקציונליות שקשורה ל-glitter (מעין API, ממשק גישה לפרוטוקול של גליטר), ולא תתעסק בתקשורת עם המשתמש או בתצוגה למסך. הקובץ הראשי שלכם יייבא את הספריה, וישתמש בפונקציות שלה כדי לבצע את כל הפעולות הנדרשות.

שימו לב שרצוי לשמור על חיבור קבוע עם השרת (במידת האפשר) ולא לבצע התחברות בכל פעם מחדש. כדי לשתף משתנים או סוקטים בין פונקציות בספריה, תוכלו להשתמש במשתנים גלובלים (global).

**דגשים**

* ההפרדה בין הקובץ swissknife לספרייה glitter היא חשובה. אם לא ברור לכם/ן לגמרי מה צריך להיות בכל קובץ, התייעצו איתנו!
* נסו כמה שיותר להשתמש בפונקציות עזר, שישמשו אתכם בכמה מקומות בקוד.
* הימנעו **~~ככל האפשר~~** **לחלוטין** משכפול קוד!
* זכרו לתעד כפי שתיעדנו לאורך כל הקורס – גם על זה תקבלו ניקוד.
* בדקו שהאולר עובד אל מול גליטר – כל מימוש חולשה צריך להופיע באפליקציה או באתר.
* עליכם לממש את כל החולשות שמצאתם, אך אם לא הספקתם אנא הסבירו למדריך בהגנה אילו חולשות שמצאתן לא קיימות באולר.

**דוגמא לפלט אולר (החולשות לא בהכרח אמיתיות.. ☺) :**

(הפלט לא מחייב מילה במילה אבל שמרו על המבנה הזה בגדול)

