תרגיל שלישי

חולשות Web

מועד הגשה –22.05.22

תכנות בטוח – 89509

הנחיות כלליות

- עבודה ביחידים בלבד.
- ניתן להיעזר במקורות אינטרנטיים, אך יש לציין באילו מקורות השתמשתם. בכל מקרה, אל הפתרון הסופי יש להגיע לבד.
- כל סקריפט אותו כתבתם לצורך פתרון התרגיל חייב להיות מצורף. בראש הסקריפט יש לכתוב בהערה מדוע כתבתם את הסקריפט ואיך הוא עזר לכם לפתרון התרגיל. הסקריפט צריך להיות מותאים לפייתון גרסה 3.8 ללא תלויות שאינן טריוויאליות.
  - שימו לב – בתרגיל זה, התלויות הטריוויאליות כוללות את כל הספריות שנמצאות ב-requirements.txt, לדוגמא flask. אך בלי לשנות גרסאות.
- כל האמור בתרגיל מתייחס לשימוש בדפדפני msedge ו-chrome, גרסאות 90 ומעלה.
- הפעילו שיקול דעת.

סעיפי התרגיל

1. נתון אתר בשם XSSApp-1.0.tar.gz. האתר הוא חבילת פייתון (3.8) שניתן להתקין, ולאחר מכן להריץ את האתר לוקאלית על ידי הפקודה python -m XSSApp, או בעזרת start_xss_app.exe במערכת ההפעלה Windows. במערכות הפעלה אחרות, הסיומת תהיה שונה.
   וודאו שהאתר עובד וקראו את הקוד שמריץ את הלוגיקה של האתר.
   נסו להשתמש בתכולות של האתר (אין הרבה).
   הסיסמא לחשבון admin היא c90fcd9b2c5b3000299db8c12c3d2157. אין להשתמש בסיסמא לצורך פתרון סעיפי התרגיל, אך ניתן להשתמש בה על מנת לדבג.
   א. הוסיפו certificate לאתר על מנת שהוא יוכל לעבוד מעל https. שימו לב – אין חובה ש-chrome יחשוב שהסרטיפיקט בטוח, אבל במידה וchrome חושב שהcertificate אינו בטוח, יש לציין למה וכיצד הייתם פותרים את זה.
   ב. מצאו חולשת XSS באתר. השמישו אותה על מנת לפרסם הודעה בתור משתמש חזק, ועל מנת למחוק את כל ההודעות באתר.
   ג. הסבירו **באופן מפורט** כיצד אתם מוודאים שההודעה החזקה שפרסמתם בסעיף קודם, תתפרסם פעם אחת בדיוק. שימו לב – לא ניתן להניח שום הנחות לא טריוויאליות, אבל מספיק להסביר את האופן בו הייתם מממשים את הסעיף. לא צריך להשמיש אותו.
   ד. נסו להריץ קוד על ידי שימוש בתגית <object> תקנית של HTML. הפעולה צפויה להיכשל, הסבירו מדוע.
   במידה והפעולה מצליחה לכם, צרפו או הפיתרון ונסו להבין מדוע ציפינו שהפיתרון ייכשל.
   ה. ספקו ארבע דרכים לשפר את הבטיחות של האתר. תנו הסבר קצר של שורה עבור כל אחת מהדרכים.
   אין לכלול את התיקון של החולשה, את העובדה שהאתר לא משתמש ב-HTTPS, או הנחה שגורם זדוני נגיש ל-stdout. (אם כי, בשרת אמיתי, לא סביר שנדפיס ערכים סודיים למסך או ללוג).
   ו. הסבירו כיצד הייתם מבצעים את התקיפה של סעיף ב' אם ה-session key היה מוגדר כ-HTTPOnly.
   ז. (רשות) הסבירו מה זה RFI/LFI.
   ח. (רשות) הריצו קוד על השרת.

הערה חשובה – באתר יש לוגיקה של העלאת קבצים. הלוגיקה רלוונטית אך ורק לסעיפי הרשות בתרגיל. אין צורך לקרוא אותה או להבין איך היא עובדת, ואין להשתמש בה לצורך פתרון סעיפים א-ו.

רמזים לתרגיל

1. את סעיף 1 ניתן לפתור באמצעות PowerShell על ידי המודול PKI, או על ידי OpenSSL.
2. BeautifulSoup לא טוב במיוחד בפרסור דפי HTML לא תקניים, ובפרט לא טוב בפרסור תגיות שבורות (שאינן סגורות כמו שצריך). לעומתם, דפדפני chromium טובים מאוד בפרסור דפים כאלה. אחת מהחולשות XSS באתר משתמשת בסעיף זה.
3. ניתן לכלול javascript על ידי <object type="text/x-scriptlet>
4. סעיף ד' הוא מעט אמורפי – מאחר שתמיד ניתן לשפר את הבטיחות של כל אתר בדרכים רבות. אין הגבלה על סוג ההגנות שהייתם רוצים להוסיף לאתר, אך השתדלו לא לבחור התקפות תיאורטיות, ולא לחזור על עצמכם. לסעיף זה יש פתרונות רבים מאוד.
5. ניתן להניח שמדי פעם משתמשים חזקים מתחברים לאתר. כלומר, אם הצלחתם להריץ javascript על כל מי שגולש לאתר, אז הצלחתם גם להריץ javascript על administratorים.

הנחיות להגשת התרגיל

את התרגיל יש להגיש בסבמיט לקבוצה 01. עיקר ההגשה הוא קובץ PDF המסביר עבור כל אחד מסעיפי התרגיל כיצד פתרתם אותו באופן מפורט – דהיינו, עם צירופי סקריפטים ואמצעי עזר נוספים בהם השתמשתם.

כל שמות הקבצים שאתם מגישים צריכים להיות הגיוניים, ועם תווים אנגליים וסימנים בקידוד ASCII בלבד (בלי אותיות בעברית או בשפות שהן לא אנגלית) – על מנת לא לבלבל את ה-submit.

לכל תרגיל צריך להיות מצורף קובץ בשם details.txt, המכיל את תעודת הזהות של המגיש, ושם המשתמש שלו בסבמיט.

שימו לב שהקובץ הוא קובץ טקסט פשוט מקודד ב-ASCII על מנת שיוכלו לקרוא אותו בקלות.
פורמט מבוקש –

name
id

במידה ויש שאלות בנוגע לתוכן התרגיל, נא להפנות לסגל הקורס בהקדם האפשרי.

הערות

טעות לעולם חוזר.

בהצלחה!
