1. שאלה 1 - Buffer overflow א'
   1. קבלת קלט מהמשתמש וכתיבתו ללוג בצירוף התאריך
   2. מחסנית:

|  |
| --- |
| logMessage 128B |
| B 64B |
| dateLength 4B |
| Date 4B |
| Canary 4B |
| Old EBP 4B |
| Ret address 4B |

* 1. על מנת לדרוס את הreturn address ללא שינוי של ה-canary נרצה שהכתיבה לזכרון בשורה 8 תצביע על ה-return address או במילים אחרות נרצה ש:
  2. ולכן, נרצה שהערך של datelength יהיה 208 אז בקליטה של b נכניס את הערך 0xDEADBEEF נוסיף 60B של זבל על מנת להגיע ל- datelength ונכניס את הערך 208. וכך בכתיבה של b לזיכרון בשורה 8   
     יעבור הערך 0xDEADBEEF ל- return address.

1. שאלה 2 - Buffer Overflow ב'
   1. מימוש נאיבי - מעבר על כל האפשרויות לcanary בגודל 4B- אפשרויות, שניה אחת עבור כל אפשרות ולכן זמן ממוצע של שניות.
   2. במקום לנסות ולנחש את כל ה-canary ביחד, ננסה לנחש כל byte בנפרד ונשתמש באתחול המערכת כאינדיקטור. כלומר, נבצע 4 סדרות ניחושים עבור כל byte- אפשרויות כך שבמקרה הגרוע ייקח לנו שניות או 17 דקות לנחש הכל ובמקרה הממוצע יקח לנו 512 שניות.
   3. נתחיל במציאת הבייט הראשון על ידי הזנת איזה שהוא ערך שוב ושוב עד שהבייט הראשון של הcanary יהיה שווה לבייט שהזנו (לכל היותר איטרציות) משם נמשיך כמו בסעיף הקודם רק שבכל איטרציה נגדיל את הבייט שמצאנו ב-1. זמן הריצה לא השתנה.
2. שאלה 3 – Firewalls
   1. נכון – בכל ממצב עוברים דרך PRX
   2. לא נכון – למשל גישה ל- FS לא עוברת בכלל דרך חומת אש וגישה ל-MS עוברת רק דרך חומה אחת.
   3. לא נכון – כל בקשה לשרת HS עוברת דרך שרת הפרוקסי שמהווה תווך ולא חיבור ישיר
   4. לא נכון – אין חומת אש בין C ל-FS, לכן אפשר להגיד שיש ביניהם חיבור ישיר

2. ניתן לזהות את העובד הזדוני ע"י כתובת source ip שלו בPRX כאשר יש בקשה זדונית.

3.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Rule ID | In/Out | Src.IP | Dst.IP | Protocol | Src.port | Dst.port | Ack | Action |
| 1 | In | External IP | 10.0.0.5 | TCP | \* | 25 | \* | Allow |
| 2 | In | 212.\*.\*.\* | 10.0.0.9 | TCP | \* | \* | \* | Deny |
| 3 | In | External IP | 10.0.0.9 | TCP | \* | 80 | \* | Allow |
| 4 | Out | Internal IP | \* | \* | \* | \* | \* | Allow |
| 5 | \* | \* | \* | \* | \* | \* | \* | Deny |

4.1 Proxy firewall – אנו מעוניינים לסנן תוכן ברמת האפליקציה לכן נשתמש בחומת אש אשר יכולה הלתמודד עם סינון מידע ברמת האפליקציה לכן נבחר חומת אש זאת.

4.2 Packet filter firewall – בשביל סינון שכזה מספיק לנו חומת אש כזאת כדי לסנן הודעות לפי IP ו- PORT מתאימים לקריטריונים.

4.3 Next generation firewall – חומת אש מסוג זה מספקת את כל השירותים הנדרשים.

4.4 Stateful inspection firewall – בשביל המקרה הנ"ל אנו צריכים לנתר סשן בין לקוח לשרת לכן חומת אש מסוג זה יודעת להתמודד עם המצב הנתון וכך תוכל לנתר סשנים, וברגע שתהיה התקפה מסוג DDOS חומת האש תוכל להתמודד עם זה.

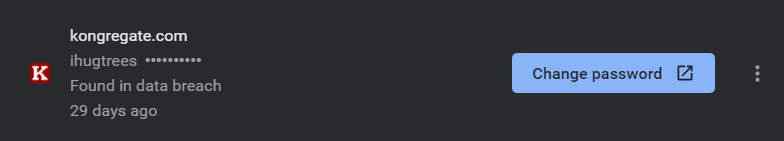
1. שאלה 4
2. Google FLoC - מטרתו להוריד את השימוש ב-cookies אשר לטענת גוגל פוגעת בפרטיות המשתמשים, ובמקום להשתמש במכונה לומדת בכל דפדפן אשר ילמד את המשתמש והפעולות שלו ויכתלג אותו בתוך קבוצה של משתמשים, ובכך תנגיש פרסומות ותוכן שיווקי על בסיס קבוצות ולא המשתמש הפרטי. כך גוגל טוענים שישפר את פרטיות המשתמש האישי.

יתרון – גוגל מסבירה שיש יתרון לפרטיות המתשמש האישי מכיוון שלמפרסמים תהיה גישה רק לקבוצות שגוגל יוצרת ולא למשתמש הפרטי.

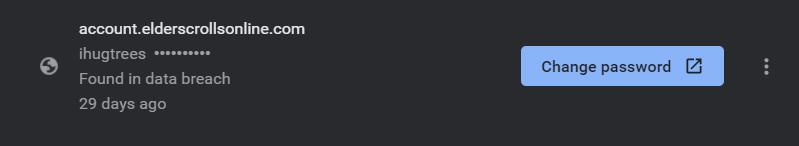
חסרון – עדיין תהיה למידת מכונה רצינית על כל משתמש בנפרד, ולגוגל תהיה עדיין גישה לכל ההיסטוריה של המשתמשים ולמידת המכונה עליהם. בעצם יורידו בעיות פרטיות שיש לקוקיז וייצרו בעיות פרטיות חדשות שאולי גרועות יותר.

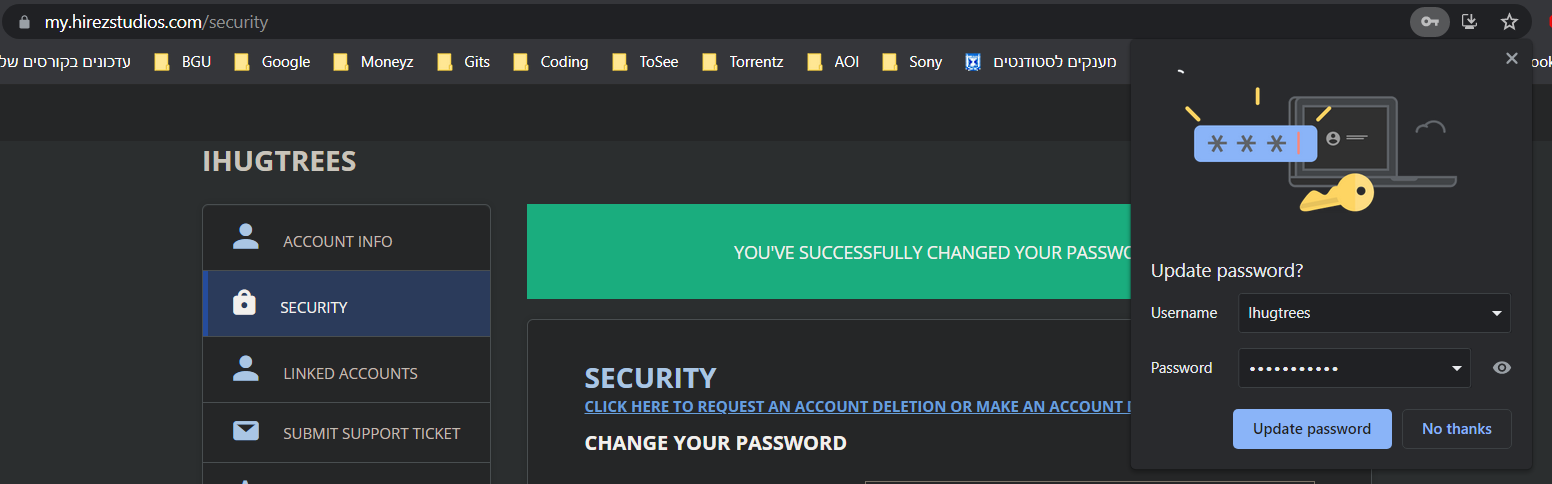
1. ההתקפה:
   1. האקרים פרצו למערכות SolarWind והתקינו שם Backdoors אשר ישתמשו בהם בהמשך, החברה מספקת שירותים להרב חברות וכאשר היה עדכון תוכנה כל החברות שמשתמשות בשירותי SolarWind ירשו את הקוד הזדוני שהיה בה. בעזרת הקוד הזדוני ההאקרים יכלו להכנס למערכות אלה ולגנוב מידע.
   2. מדוברת במתקפה מסוג supply chain attack. במתקפה זו ההאקרים תוקפים ארגון צד שלישי שמספק שירותים לחברות שאותם ההאקרים מעוניינים לתקוף באמת.
   3. ה-attack vector שהיה בסיפור הזה הוא backdoor.
   4. ההתקפה גרמה לנזק עצום בקנה מידה בכך ש-18,000 משתמשים הורידו את הקוד הזדוני וביניהם חברות ממשלתיות וביטחונית של ארה"ב, וגם חברות פרטיות גדולות כמו מיקרוסופת ניזוקו בהתקפה זו בכך שנגשנבו פרטים בעיקר.
   5. ניתן להתגונן מפני התקפות מסוג זה ע"י כמה דרכים, הבולטות מביניהם: honeytokens, secure privileged access management, zero trust architecture...

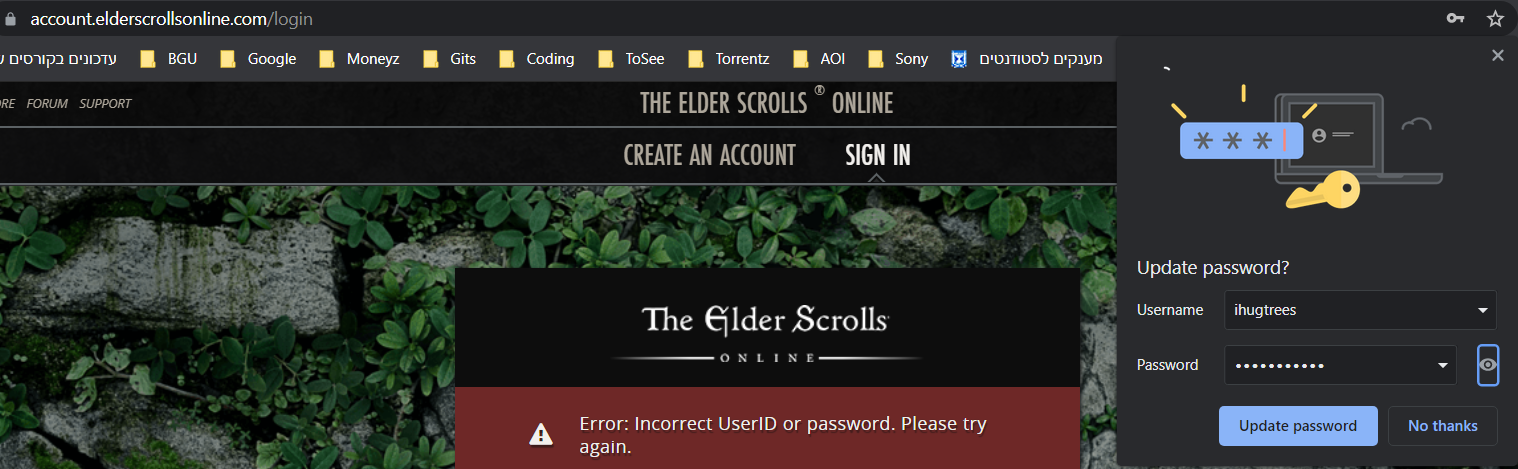
בונוס:

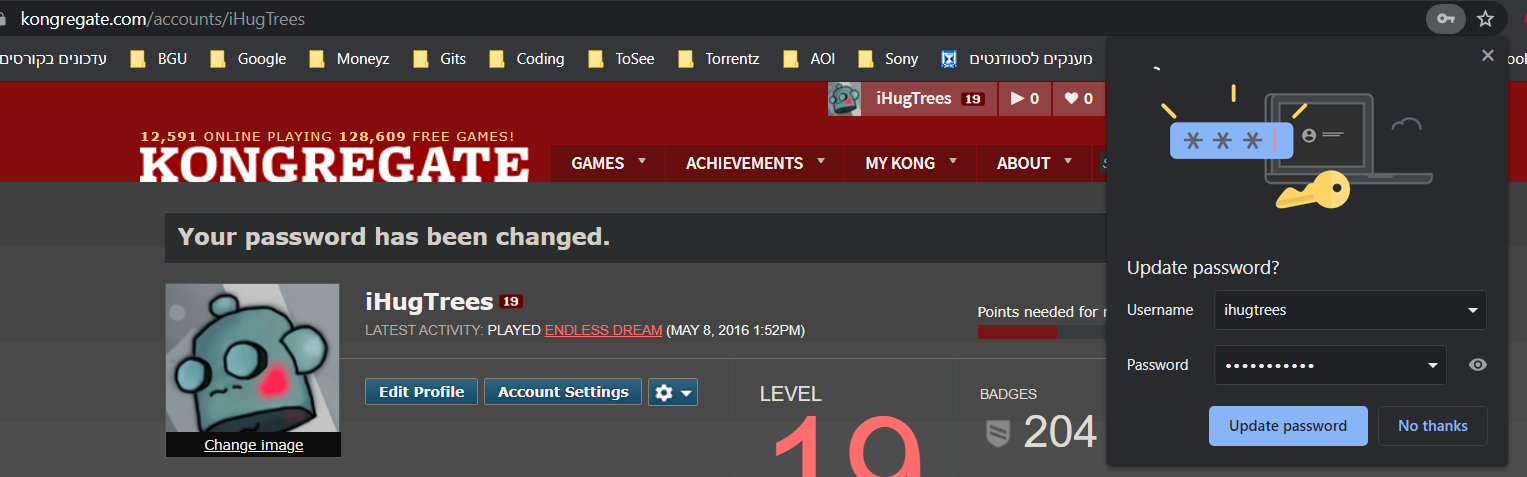












1. שאלה 5
   1. מעל המחסנית נמצא מרחב ה-kernel space, תפקידו הוא ניהול הזיכרון והתהליכים של מערכת ההפעלה, ניתן לעבור בין המרחבים ע"י system calls.
   2. ניתן לבצע התקפות buffer overflow, ניקח לדוגמא את הקוד משאלה 1:

char\* date = getDateString();

int dateLength = strlen(date);

char b[64];

ESP

|  |
| --- |
|  |
| logMessage 128B |
| B 64B |
| dateLength 4B |
| Date 4B |
| Canary 4B |
| Old EBP 4B |
| Ret address 4B |

gets(b);

char logMessage[128];

strcpy(logMessage, date);

strcpy(logMessage + dateLength, b);

writeLog(logMessage);

כפי שראינו בשאלה 1 ניתן לדרוס את הreturn address ללא פגיעה בcanary ולכן, ניתן להכניס לכתובת החזרה כתובת גדולה שתגיע לתוך ה-kernel space.

1. שאלה 6
   1. ניתן לראות את ההבדלים העיקריים בטבלה הבאה:

|  |  |
| --- | --- |
| TCP | UDP |
| דורש חיבור רציף בין שני הצדדים | אין צורך בחיבור רציף, שולח מידע לכל מי שמאזין. |
| מבטיח את הגעת המידע ליעד | אינו מבטיח. |
| שולח מחדש חבילות שאבדו | אינו שולח מחדש |
| בודק את תקינות המידע במגוון שיטות, מחכה לאישור שהמידע הגיע | בודק שגיאות רק באמצעות checksum |

* 1. מקים השיחה שולח syn לצד השני על מנת לידע אותו שהוא רוצה לפתוח תקשורת,

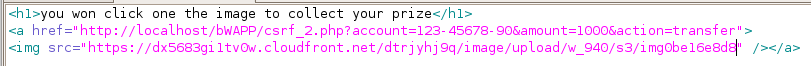
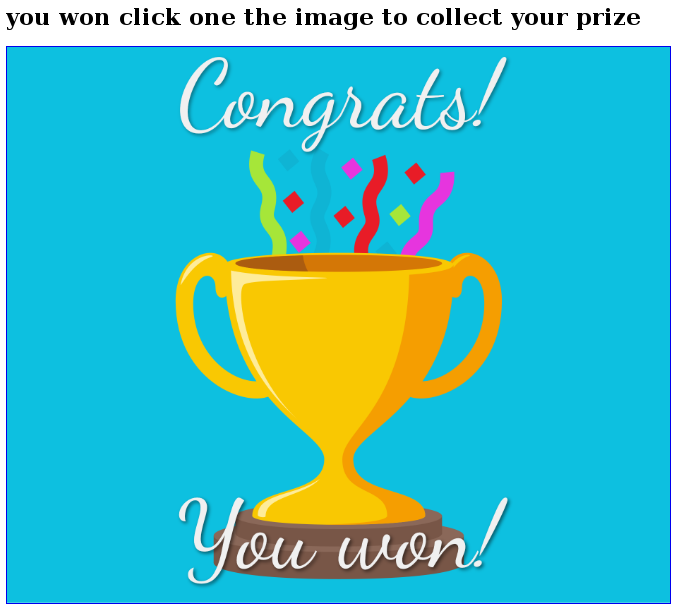
הצד השני מחזיר syn-ack על מנת להודיע כי הוא מסכים לפתיחת התקשורת,

הצד הראשון מחזיר ack וכך יודעים שהתקשורת תקינה לשני הצדדים.

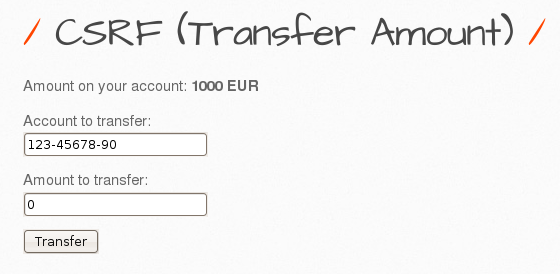
* 1. Syn-flood זוהי התקפת dos שבה התוקף שולח הרבה הודעות syn מכתובות ip מזויפות, השרת צריך לפתוח סוקט עבור כל בקשה כזאת ומחכה ל-ack וכך התוקף מנצל את כל משאבי השרת.
     1. דרך ראשונה: שימוש ב-proxy:  
        ע"י שימוש בפרוקסי בעל קיבולת חיבורים גדולה יותר מהשרת, נעביר לשרת רק את החיבורים אשר אומתו וכך נשאיר את החיבורים החצי פתוחים בפרוקסי ולא ננצל את משאבי השרת.
     2. דרך שניה: הגבלת החיבורים החצי פתוחים

נגדיר איזה שהוא קבוע k שיהווה את מספר החיבורים החצי פתוחים המקסימליים וכך בכל פעם שמספר החיבורים החצי פתוחים יעבור את k נתחיל לסגור חיבורים חצי פתוחים לפי fifo.

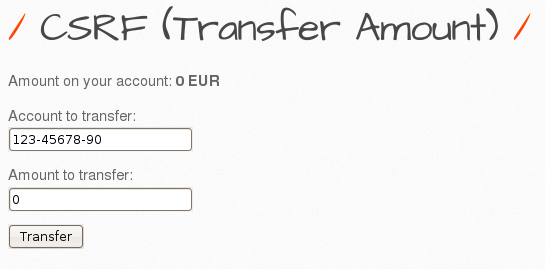
* 1. התקפת syn-Ack זאת התקפה בה התוקף מנסה לתקוף לקוח של השרת. התוקף שולח בקשות syn רבות עם הכתובת ip של הקורבן והשרת שולח לקורבן בקשות syn-ack בהתאם ובעצם מעמיס על המשאבים של הקורבן.

1. שאלה 7
   1. CSRF (Cross Site Request Forgery) היא התקפה נגד גולש אינטרנט המאלצת את דפדפן האינטרנט של הגולש לבצע פעולות לא רצוניות ביישומי אינטרנט בשמו של המשתמש בשירות באמצעות ניצול מגבלות בפרוטוקול HTTP. בהתקפה זו התוקף, על ידי הרצת קוד בצורה סמויה, מבצע פעולות שונות בשם הקורבן, כגון: לגרום למשתמש להעביר כספים לגורם שלישי, לרכוש פריטים שונים, לשנות כתובת דואר אלקטרוני מוגדרת לכתובת אחרת, או להחליף תמונת פרופיל, כל זאת מבלי שהקורבן מודע לביצוע פעולות אלה.
   2. נבנה דף html המכיל תמונה וקישור:  
        
      

מצב החשבון לפני לחיצה על התמונה:



וכעת לאחר לחיצה על התמונה:



* 1. דרכי התגוננות:
     1. אימות המפנה כך שרק שאילתות מהדף המקורי יבוצעו.
     2. אימות המשתמש ע"י שם משתמש וסיסמא בעת ביצוע פעולות.
     3. Cookie timeout – אחרי זמן כלשהו העוגיות יבוטלו.

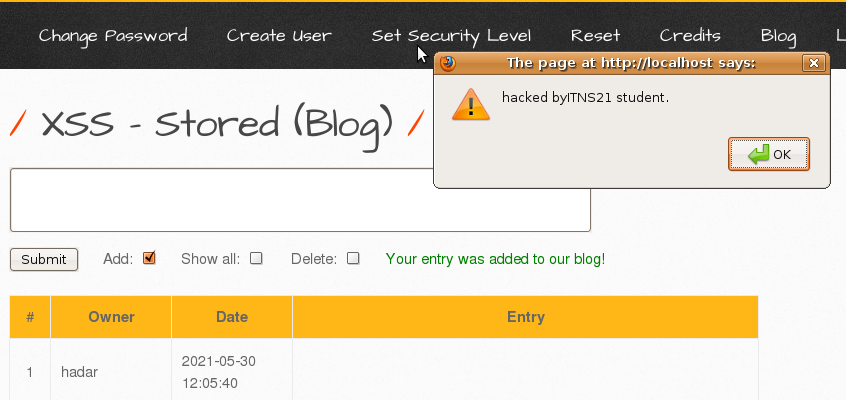
1. שאלה 8
   1. XSS היא התקפה נגד גולש אינטרנט המנוצלת באמצעות פגיעות ביישומי אינטרנט ומאפשרת לתוקף להזריק סקריפט זדוני שמטרתו לרוץ בדפדפנים של משתמשי מערכת אחרים
   2. על התוקף לבצע את הפעולות הבאות:
      1. הזרקת סקריפט שמוציא את ה-ip וה-cookies ושומר את המידע.
      2. הזרקת הסקריפט באמצעות Text input.
      3. שליחת הurl עם הסקריפט לקורבן.

לאחר שהקורבן ילחץ על הקישור לתוקף יהיה גישה למידע הרצוי.

* 1. מספר בדיקות שניתן לעשות:
     1. בדיקה האם הלינק מכיל תגיות <script> ולא לאשר אותן
     2. שימוש ה-WAF.
     3. החלפת תגיות חשודות כגון < בתו lt&.

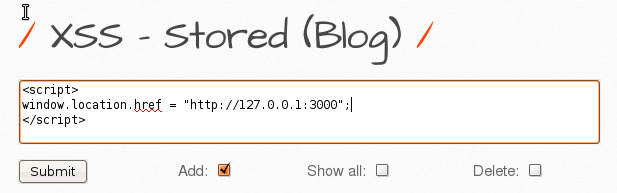
1. שאלה 9
   1. XSS-Stored הינה מתקפה בה התוקף מזריק סקריפט זדוני אל אתר אינטרנט כך שהסקריפט נשאר שם באופן קבוע. בצורה זאת הסקריפט מופעל בכל פעם שגולשי האתר נכנסים אליו.
   2. נכניס את הסקריפט הבא:





ההודעה תחזור גם כשנלחץ OK והדרך היחידה לצאת מהלולאה היא להרוג את התהליך של הדפדפן.

* 1. התקנו bee-box ולכן אנו רוצים להעביר את הנתקף לאתר לבחירתנו ולקבל חיווי על המעבר.  
     לכן על התוקף לבנות שרת שאליו יאזין.

השתמשנו בסקריפט הבא על מנת להעביר את הנתקף לשרת לבחירתינו:

בנוסף הקמנו שרת המדפיס את כתובת האייפי של הנתקף ומעביר אותו לאתר לבחירתינו:



ובכניסה לבלוג התקבל בצד השרת:

