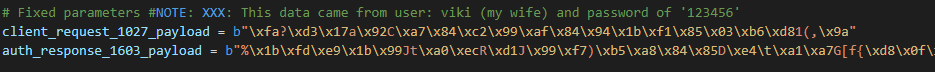
מבוא לאבטחת המרחב המקוון (20940)

**שגיב אבו – 206122459**

**שאלה 2 – הסבר התקפת מילון לא-מקוונת נגד הפרוטוקול שבניתי**

1. התבססתי על כך ש"שלפתי" מן התקשורת 2 הודעות:
   1. הודעה 1027 – בקשה למפתח סימטרי מהקליינט אל שרת ההרשאות.
   2. הודעה 1603 – תשובה משרת ההרשאות אל הקליינט המכיל תוכן מוצפן.



1. 2 הודעות אלו מכילות את ה ‘nonce’, פעם אחת כאשר הוא לא מוצפן בהודעה 1027, ופעם אחת כאשר הוא מוצפן יחד עם המפתח הסימטרי עצמו! (1603)
2. המטרה היא לנסות לבנות מפתח משלי ע"י ניחוש הסיסמא, ביצוע hash ובניית המפתח (כי הIV עובר בצורה גלויה), עם המפתח הזה לנסות לפענח את ה nonce המוצפן, להשוות ביניהם – ואם הם זהים -> הפענוח עבד -> מצאתי את הסיסמא של המשתמש.
3. בתחילת הקוד אני מחלץ את הnonce הלא מוצפן – זה יחסית קל כי אני יודע למוזר ש ID של server הוא 16 בתים ולכן די קל לדעת מניין ה nonce מתחיל.
4. כנ"ל לגבי התשובה – הגורם הראשון הוא client\_id שזה 16 בתים, ולכן ה Encrypted\_key המכיל בתוכו את המפתח הסימטרי והnonce מתחיל מ16.  
   (ניתן בקלות לחשב כמה בתים סה"כ, גם אחרי Padding כי padding מעגל לקפיצות של 16 לפי הגרסה שלנו).
5. בשלב הבא אני בונה את המילון, בניתי אותו יחסית מצומצם רק לטובת הדוגמא.
6. רץ על כל סיסמא במילון, מבצע לה HASH.
7. **הערה:** בקוד אני אכן משתמש בהגנה נוספת של SALT, שמקשה על ניחוש הסיסמאות. פה יצאתי בהנחה כאילו לא השתמשתי בsalt, או שאני יודע אותו, כדי להראות כמה זה מקל על תהליך התקיפה.
8. מייצר מפתח הצפנה מה hash של הסיסמא.
9. מנסה לפענח את ההצפנה.
10. אם נפלתי על incorrect unpadding – המפתח אינו תקין.
    1. עובר לסיסמא הבאה.
11. לוקח את הnonce שפענחתי, משווה אותו לnonce המקורי ובכך יודע אם המפתח תקין ובהתאם האם הסיסמא נכונה.

**פתרון?** – פשוט להשתמש בsalt – זה מקשה פי כמה וכמה על ניחוש הסיסמא עד כדי בלתי אפשרי.

כמובן שניתן גם לבחור סיסמא מסובכת וארוכה, מגוונת.