מבוא לקריפטוגרפיה - תרגיל בית 3

נא להגיש עד ה 12.3.2017 דרך המודל. ההגשה היא ביחידים. התייעצות מותרת, אך חובה לכתוב את הפתרונות לבד.

נושאים: קודים לאותנטיקציה של הודעות (MAC)

1. בהרצאות האחרונות הגדרנו קודים לאותנטיקציה של מידע. (Message Authentication Codes). בניגוד למערכות הצפנה, מטרתם של קודים כאלה לוודא את נכונות המידע העובר בערוץ, במידה והתוקף מנסה לשנות אותו. נתנו הגדרה לקוד MAC עם בטיחות חישובית (זו תזכורת, ולא הגדרה פורמלית לחלוטין. ההגדרה בכיתה נוסחה באמצעות תוקף ו chellanger):

- Correctness: For all n and all $k \in K_n, m \in M_n$, $Pr[Verify(m, MAC(k,m)) = 1] = 1$.
- Existential unforgeability (short reminder): For any PPT adversary $A(1^n)$ obtaining an oracle access to $MAC_k()$, where k is picked at random from $Gen(1^n)$, manages to generate a pair m, t such that $Verify(k, m, t) = 1$ and m was not sent to the oracle as a query is negligible in n. The probability is taken over the random choice of k, and the randomness of the MAC_k oracle.

א. נתבונן במשפחה של קודי אותניקציה $MAC = (Gen_n, MAC_n, Verify_n)$ המוגדרת מעל הקבוצות $M_n = K_n = \{0,1\}^n, T_n = \{0,1\}^{10}$ המקיימת נכונות.

הוכיחו שמשפחה זו בהכרח אינה מקיימת existential unforgeability.

ב. בהרצאה בנינו MAC להודעות חסומות $M_n = K_n = T_n = \{0,1\}^n$. התחלנו ממשפחה של פונקציות חד כיווניות $f_n(k, m) : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^n$. כזכור, $Gen(1^n)$ מייצר ייצוג k עבור פונציה אקראית במשפחה (במאצעות קביעת האינדקס k) $f_{n,k}(m)$, ו $MAC_k(m) = f_{n,k}(m)$. הוכיחו שהבניה אכן מקיימת נכונות ו existential unforgability.

2. בשאלה זו נרחיב את הבניה שראינו בכיתה להודעות באורך משתנה. כלומר $M_n = T_n = \{0,1\}^*, K_n = \{0,1\}^n$. הגישה הכללית היא לקחת בניה ל MAC עבור אורך קבוע (עם פרמטרים $M_n = K_n = T_n = \{0,1\}^n$ לדוגמה, כמו שעשינו בהרצאה), ולבסס עליה MAC' עם אורכים משתנים. הגישה הכללית היא לחלק הודעה m לבלוקים $m_1, m_2, \ldots, m_l$ באורך n ולהפעיל את הבניה המקורית באופן כלשהו על הבלוקים. בסעיפים 1-3 נבחן שלוש גישות לבניה כזו. בכל סעיף הראו מדוע הבניה אינה עובדת ע"י בניית תוקף מתאים עבור משחק ה existential unforgeability.

א. נציע בניה שבה $MAC_k(m_1, \ldots, m_l) = m_1 \oplus m_2 \ldots, \oplus m_l$.

תנו דוגמה לתוקף יעיל המנצח במשחק ה existential unforgeability.

ב. נציע בניה שבה $MAC_k(m_1, \ldots, m_l) = f_k(m_1), f_k(m_2), \ldots, f_k(m_l)$.

כאשר f_k היא פונקציה פסוודו אקראית, כמו בבניה שראינו בהרצאה. הסבירו בקצרה מדוע ההתקפה שהראיתם בסעיף הקודם אינה רלוונטית כאן. הראט התקפה אחרת שכן תעבוד.

ג. נציע תיקון למערכת מהסעיף הקודם המוסיף את אורך ההודעה בבלוקים. $MAC_k(m_1, \ldots, m_l) = f_k(1, l, m_1), f_k(2, l, m_2), \ldots, f_k(l, l, m_l)$. כדי להשאיר מקום גם לאינדקסים, אורך ה m_i ים כאן יהיה $n/3$ ולא n. בפועל, בניה זו מגבילה את אורך ההודעות ב M_n ל $2^{n/3}$ בלוקים, אולם זה לא בעייתי, כי זוהי פונקציה סופר-פולינומית, וגם כך בטיחות מתקיימת רק להודעות באורך פולינומי ב n. תנו דוגמה לשתי התקפות שונות על הבניה בסעיף הקודם שלא יעבדו כאן. תנו דוגמה להתקפה שעדיין אפשרית כאן.

ד. להלן פיתרון שעובד $MAC(m_1, \ldots, m_l) = f_k(r, 1, l, m_1), f_k(r, 2, l, m_2), \ldots, f_k(r, l, l, m_l)$. כאן כל אחד מחלקי הקלט ל MAC הוא באורך $n/4$ (כולל ה m_i ים). r כאן היא מחרוזת אקראית באורך $n/4$. השלימו את אלגוריתם $Verify$, והסבירו מדוע ההתקפה מהסעיף הקודם אינה עובדת. עדיף לתת הוכחה מדוייקת (בונוס 10 נקודות), עם חישובי הסתברויות וכו המראה שכל תוקף PPT אינו מצליח לזייף. אבל, אפשר גם להראות רק מדוע ההתקפה מסעיף 3 אינה עובדת.