ממ"ן 13

שאלה 1

1. על מנת למצוא האם הבנייה המוצעת היא פונקציית תמצות קריפטוגרפית חסינת התנגשויות, עלינו בעצם להראות האם ניתן למצוא x ו- x' שונים באופן ישים כך שמתקיים h(x)=h(x') . נראה כי פונקציה זו היא אכן חסינת התנגשויות, נוכיח בשלילה שהיא אינה ונגיע לסתירה. נניח כי קיימים x' ו- y' כך ש x'||y' != x||y ומתקיים ש h(x||y)=h(x'||y') כלומר מתקיים ש AESx(y)= AESx'(y') . הפונקציה AES היא פונקציית הצפנה בטוחה כך שאם המפתח או הקלט שונה נקבל טקסט מוצפן שונה ורק עבור טקסט גלוי זהה ומפתח זהה נקבל טקסט מוצפן דומה, מכאן נסיק שהשוויון AESx(y)= AESx'(y') נכון אם ורק אם x=x' וy=y' כלומר חייב להתקיים ש x'||y' = x||y ולכן התשובה היא שהבנייה הנתונה היא פונקציית תמצות קריפטוגרפית חסינת התנגשויות.
2. על מנת למצוא האם הבנייה המוצעת היא פונקציית תמצות קריפטוגרפית חסינת כנגד preimage, עלינו בעצם להראות האם לא אפשרי למצוא בתקופת זמן שלנו מקור לתמונה. נניח כי יש לנו את y שזה ערך התמונה של קלט מסויים, מהבנייה של h נוכל להסיק שאם נשרשר לy את הספרה 1 (בלי הגדרת הכלליות) נקבל את הפלט של הצופן AES . מכיוון שההצפנה מתבצעת בקופסא שחורה עם מפתח k שאינו ידוע , נצטרך לעשות חיפוש על 2128 מפתחות, ודבר זה בלתי ישים בתקופתנו. (יכולנו למצוא מקור כלשהו בכך שנבחר מפתח באקראי ונמצא מקור כלשהו אך אותו מקור לא ייתן לנו את y עם הקופסא השחורה הנתונה) מכאן שפונקציית התמצות הקריפטוגרפית היא חסינה כנגד preimage.

שאלה 2

1. הפונקציה h אינה חסינת התנגשויות, נראה דוגמא. נתון כי h1 חסינת התנגשויות לכן ניקח הקלט שלה יהיה x . לעומת זאת h2 אינה חסינת התנגשויות ולכן נניח שעבור הקלטים y ו z שונים מתקיים ש h2(y)=h2(z). מכאן שקיבלנו שהקלטים של h יהיו שונים (קרי, x||y!=x||z) אך התמונה תהיה זהה כלומר h1(x)||h2(y)== h1(x)||h2(z) . ולכן הפונקציה h אינה חסינת התנגשויות
2. הפונקציה h אינה חסינת התנגשויות, נראה הדוגמא לסעיף קודם היא גם רלוונטית לסעיף. נקבל ש h1(x)⊕ h2(y)== h1(x) ⊕h2(z) ולכן הפונקציה h אינה חסינת התנגשויות
3. הפונקציה אינה חסינת התנגשות, נראה דוגמא. יהיו x ו-y שונים כך ש h2(y)=h2(x) ,מכאן ש

h(x)=h1(h2(x))= h1(h2(y))=h(y)

מכאן שקיבלנו קלטים שונים עם אותו תמונה, ולכן הפונקציה h אינה חסינת התנגשויות

1. הפונקציה אינה חסינת התנגשות, נראה דוגמא. יהיו x ו-y שונים, מכיוון ש h1 חסינת התנגשויות נקבל ש h1(y)!=h1(x) ,נסמן אותם ב z וב- w . כלומר,z=h1(y)!=h1(x)=w . פונקציה h2 אינה חסינת התנגשויות ולכן ניתן להסיק כי h2(w)=h2(z) מכאן שנקבל

h(x)=h2(h1(x))=h2(w)=h2(z)=h2(h1(y))=h(y)

ולכן הפונקציה h אינה חסינת התנגשויות.

1. אני מניח כי x'=x⊕{1}|x| אם כן אז פונקציית זו חסינת התנגשויות, נוכיח זאת. h1 היא חסינת התנגשויות ולכן כל מקור x שניתן לה יפיק ערך שונה בתמונה. כמו כן, x' נותן את ההופכי של x ולכן h1(x)!=h1(x') .מכאן שעבור כל קלט x הפלט של h(x) יהיה שונה גם ברישא וגם בסיפא כי השרשור בניהם יהיה שונה תמיד. מכאן שפונקציית זו חסינת התנגשויות.

שאלה 3

טענה זו אינה נכונה. ביצוע פעולת xor על 2 פונקציות תמצות שונות לא חסינות התנגשות, אינה תביא פונקציית תמצות חסינת התנגשות. מכיוון שכל אחת מהן אינה פונקציית תמצות חסינת התנגשות ייתכן שעבור 2 קלטים שונים x ו-y הפלט שלהן יהיה זהה, כתוצאה מכך אם נפעיל את פעולת xor בין הפלטים נקבל שהפונקציית התמצות הכולל תיתן פלט אפסים עבור 2 קלטים שונים. לכן נסיק שמסקנה זו אינה נכונה.

שאלה 4

1. בניית פונקציית האותנטיקציה אינה בטוחה, נדגים התקפה בה היריב יכול לזייף MAC . מכיוון שהודעת m והפונקציה עצמה ידועות ליריב, נוכל להפוך את הסדר של mi בהודעה m ולקבל m'. מכיוון שהXOR היא פעולה קומוטטיבית אז הAES ייתן אותה תוצאה לMAC עבור הודעות שונות ולכן נוכל לזייף את הMAC.
2. בניית פונקציית האותנטיקציה אינה בטוחה, נדגים התקפה בה היריב יכול לזייף MAC. מכיוון שהודעת m ידועה ליריב והפונקציה עצמה ידועה נוכל על פלט כלשהו,y, שמתקבל מהפונקציה לעשות רברסינג. כלומר לחשב את AES-1(y) ואז נקבל את הערך המקור k שהוא המפתח הלא ידוע. כאשר נמצא את k נוכל לבחור איזה הודעה m' שנרצה (לפי הדרישה של m) וניצור בשבילו MAC חדש, הפונקציה אינה בטוחה כלל מכיוון שברגע שנגלה את המפתח k נוכל לעשות כרצוננו.
3. בניית פונקציית האותנטיקציה אינה בטוחה, נדגים התקפה בה היריב יכול לזייף MAC. מכיוון שהודעת m ידועה ליריב והפונקציה עצמה ידועה נוכל על פלט כלשהו,y, שמתקבל מהפונקציה ושינוי סדר חלקי ההודעה mi ליצור הודעה m' חדשה, ליצור MAC חדש בכך שנשנה את הסדר של חלקי ההודעה y ,yi , כך שיהיה תואם לסדר של ההודעה החדשה. כלומר, AESk(mi)=yi ונשנה את הסדר של yi בהתאם לסדר החדש של mi המופיע בm'.

לדוגמא, נניח שנשנה את הסדר בין ההודעות m1 לבין m2 וזו תהיה m' אז

MAC(m',k)=AESk(m2)|| AESk(m1)||…|| AESk(mn)

ולכן יהיה MAC חדש, עבור m' החדש.