ציתות למקלדת ע"י חיישנים של טלפונים חכמים

מאת: שלמה יונה

מבוא

דרך קלה לאסוף מידע רגיש ממערכות מחשב היא באמצעות ריגול אחרי אמצעי קלט. למשל, צפייה במקלדת בעת שמקלידים עליה יכולה להניב את רצף התווים שהוקלדו ומכאן לטקסט שהוקלד ואולי אפילו לפעולות שונות שבוצעו. אפשר לצלם מקלדת של אדם שמקליד (למשל סיסמה בכספומט או בעת גישה למשאב מוגן אחר כלשהו במערכת מחשב). אפשר גם להטמין תוכנה שאוספת את הארועים מאמצעי הקלט, כמו עכבר ומקלדת שבעצם מקליטה את התוים שהוקלדו (ואולי גם פעולות עכבר). שלומי נרקולייב הציג בגליון 10 שיטה לאיסוף כזה של התוים שהוקלדו ושל פעולות העכבר באמצעות אתר אינטרנט. בכל אלה לא נעסוק במאמר זה. תחת זאת נבחן גישה אחרת שאותה נתאר באמצעות התרחיש הבא:

דמיינו לעצמכם שבשעה שאתם עובדים על המחשב ניגש אליכם אדם ששואל אתכם שאלה או שנמצא בקרבת שולחן העבודה שלכם. אתם עירניים מאוד ואפילו מספיקים להאפיל את המסך ולנעול אותו או אפילו את החשבון שלכם לפני שהאדם מתקרב. נפלא. הוא נעצר לידכם נשען על שולחן העבודה ומניח את הטלפון הנייד שלו תוך כדי כך. הוא משוחח עמכם ואגב כך מסתובב ומחליף מילה עם אדם שעובד בסמוך אליכם ואז עוזב. אתם כבר איבדתם קשב כי כל מה שעניין אותכם זה שהוא עזב ושאתם יכולים לחזור לעבודתכם. בשלב הבא אתם מקישים את הסיסמה כדי לגשת לחשבון או כדי להסיר את הנעילה מהמסך וממשיכים בשלכם. לא הבחנתם שהנייד של האדם נשאר אי שם על שולחן העבודה שלכם, או שהבחנתם וחשבתם שתשיבו לו אותו אחר כך כשתקומו ממקומכם. בכך הזמן שאתם עובדים על המחשב והטלפון הזה נמצא על שולחן העבודה שלכם המכשיר מקליט את הקשות המקלדת שלכם ואולי אפילו משדר את המידע בזמן אמת דרך אינטרנט אלחוטי או אמצעי דומה.

עכשיו חשבו שמישהו מצליח להכניס קוד כזה לאפליקציות שמורידים לטלפונים החכמים. הרשאות שימוש בחיישנים אינרטיים[2] קל לנו מאוד לתת כי איך כבר יכולים להזיק לנו עם חיישן אינרטי?

הנה תרחיש שבוודאי חוזר על עצמו בכל משרד בכל מקום בעולם כל הזמן: אתה מתיישב לעבודה על המחשב שלך ומניח את הטלפון החכם שלך לצידך על שולחן העבודה ומתחיל בעבודה. ומה אם קוד זדוני שמותקן בטלפון החכם מקליט באופן מתוחכם את הקשות המקלדת?
איך זה עובד? איך עושים את זה? מי עשה את זה כבר? איך להתגונן מפני זה? מה הלקחים? בשאלות אלה נעסוק במאמר זה.

וקטור ההתקפה

משתמש בטלפון חכם מתומרן להוריד אפליקציה שאינה דורשת שימוש בהרשאות מחשידות. משעה שהותקנה ועובדת התוכנה מנסה לזהות שהמכשיר נייח, ואז מתחילה בניסיונות לזיהוי הקשות על מקלדת ובאיסוף ו/או בשידור המידע שזוהה.

מי כבר עשה את זה? מה החידוש?

מתברר שבאמצעות חיישנים אינרטיים: מד-תאוצה (אקסלרומטר) ומד-מהירות-זויתית (ג'יירוסקופ) שקיימים בטלפונים חכמים רבים (באייפון כבר מגרסה 4 יש בנוסף למד התאוצה גם מד מהירות זויתית, באנדרואיד יש כאלה למשל בסמסונג IIs). ניתן להסיק מתוך התנודות, שיוצרות האצבעות שלנו שמקלידות על המקלדת ועוברות גם לשולחן העבודה ונקלטות בחיישני המכשיר, מה הקלדנו. כל מקש שעליו אנחנו לוחצים. מתברר שאפשר להסיק בדיוק של כ-80% (זאת אומרת בממוצע להסיק נכון 8 מתוך כל 10) תוים. בעבר כבר ביצעו ציתות להקשות במקלדת באמצעות מיקרופון. השימוש בחיישנים אינרטיים לצורך הציתות מנצל את אותם העקרונות. המיקרופון רגיש בהרבה מהחיישנים האינרטיים ובנוסף גם תדירות הדגימה גבוהה באופן ניכר: 44.1 קילוהרץ למיקרופון בטלפון חכם (במקרים מסויימים אפילו בקצב של 2.5 ג'יגה-הרץ) לעומת כ-100 דגימות בשנייה בחיישנים האינרטיים של המכשיר. הרשאות על שימוש במיקרופון שמתבקשות בעת התקנת האפליקציה כנראה ירימו דגל אדום אצל משתמשים זהירים וחשדניים, שהרי התקפות באמצעות הקלטת שמע ידועות ומפורסמות כבר ממזמן - אבל מי מפחד מהתקפה באמצעות מד-תאוצה וג'יירוסקופ? מסתבר שאפילו ברוב המקרים אין מבקשים אישור המשתמש כדי להרשות לאפליקצייה לגשת למידע מהחיישנים הללו.

יש לא מעט דוגמאות קודמות מהעבר שעשו דברים דומים באמצעים אחרים, אולי אחד המגניבים ביותר שעשו בעבר זה לצותת להקשות מקלדת באמצעות לייזרים וולטמטרים [4].

איך עושים את זה?

משתמשים בתנודות שנקלטות בחיישן התאוצה, האקסלרומטר. מתברר שיש קושי רב בזיהוי מדוייק. כשמצרפים גם את הפלט ממד המהירות הזוויתית, הג'ירוסקופ, ניתן להעלות את רמת הוודאות עוד יותר ולהגיע לדיוק סביר. הצד השלילי עם חיישנים אלה הוא שיש להם בעיות קשות עם דיוק ועם רעש. הצד החיובי הוא שהבעיות שלהן בתחומים שונים ולכן ניתן לפצות על חולשות של חיישן מסוג אחד עם החוזקות של החיישן מהסוג השני ולהיפך. למתעניינים ביישומים מגוונים של שימוש בשילוש של אקסלרומטר (3 צירים), ג'יירוסקופ (3 צירים) ומצפן (3 צירים) מומלץ לצפות בשיחה טכנית בגוגל של דיויד זאקס מאינוונסנס[3].

התרגום מפלט החיישנים בתגובה לתנודות לתווים וברמה גבוהה יותר למילים ולטקסט נעשה בשיטות הסתברותיות באמצעות מודל סטטיסטי של זוגות של הקשות. בעוד שזיהוי ברמת וודאות סבירה של מקש בודד הוא בהסתברות נמוכה, ההסתברות לזיהוי של צמדי הקשות מקשים מתוך המידע הוא גבוה באופן משמעותי ומביא לזיהוי ברמת ודאות סבירה. עבור כל צמד הקשות המודל מנבא האם מדובר ב-צד ימין של המקלדת לעומת צד שמאל של המקלדת, והאם המקשים קרובים זה לזה או רחוקים זה מזה. בניבוי הזה המערכת משתמשת לצורך חיפוש במילון שבו כל מילה מיוצגת על ידי רצף דומה של מאפיינים, האם האותיות הן "ימין" או "שמאל" והאם הן "קרובות" או "רחוקות" על מקלדת תקנית בפריסת QWERTY. הניבוי אמין עבור מילים שאורכן לפחות 3 תווים ושקיימות במילון. עבור מילון שבו מיוצגות חמישים ושמונה אלף מילים ההסתברות למציאת מילה במילון כך שהמילה גם נכונה היא 80%.

כדי לממש זאת, השתמשו בדגימת הנתונים מהחיישנים (אפשר בקלות לחפש בגוגל קוד ולמצוא), מהאותות הגולמיים שנשמרים מחשבים כמה ערכים מספריים חדשים כדי לשקף תכונות שונות של המידע ושל צירופים של המידע מהחיישנים. למעשה, עבור כל הקשת מקלדת (משך לחיצה בממוצע, אגב הוא כעשירית השנייה) הוא אוסף סדור של הערכים הבאים שמוסקים מתוך ערכי האקסלרומטר בשלושה צירים: mean, kurtosis, variance, min, max, energy, rms, mfccs, ffts. לפי הסדר מדובר ב-ממוצע (תוחלת), גבנוניות, שונות, ערך מינימום, ערך מקסימום, אנרגיה, שורש ממוצע הריבועים, ספסטרום תדירות מל, התמרת פורייה מהירה. בנתונים הללו משתמשים באלגוריתמי למידה כדי לתייג את הקידוד (ימין / שמאל / קרוב / רחוק). ולבסוף מאמנים מודל לחיזוי זוגות של אותיות בהנתן האותות המתוייגים. למתעניינים בלמידת מכונה אני ממליץ על קורס המבוא המקוון של אוניברסיטת סטנפורד בנושא[5].

מה שנותר זה להסיק את המילים בהינתן רצפים של זוגות אותיות משוערות. את זה עושים באמצעות שימוש במודל המילים המשוער (תוצר של עוד למידת מכונה) באורך n-1 כאשר לכל מילה במילון שאורכה n. אלגוריתם התאמת המילים מרצפי זוגות ניבויי האותיות מוצג בתרשים.

המילים במילון שקיבלו עתה ציון לפי סבירות שמישותן על פי קלט רצפי זוגות האותיות ממויינות לפי הציון מהגבוה לנמוך.

Algorithm 1 Word matcher scoring

```
1: curScore = 0
2: for all words in dic of len(n) do
3:     for i = 0 to 2 do
4:         if dic.word.profile[i] = prediction.profile[i]
           then
5:             if dic.word.profile[i] = L or
               dic.word.profile[i] = R then
6:                 curScore++
7:             else if dic.word.profile[i] = N or
               dic.word.profile[i] = F then
8:                 curScore++
9:             end if
10:        end if
11:    end for
12: end for
13: return curScore
```

[אלגוריתם התאמת המילים בו נעשה השימוש]

חשוב לציין כי בתצורת המימוש הנוכחית, יש מספר אילוצים שחייבים לעבוד בהם, ואלו הם:

- ניתן לקלוט את הקלדות המקלדת אך ורק ממרחק של לא יותר מ-8 ס"מ.
- ניתן לבצע זאת רק על שולחן מעץ.
- ניתן לבצע זאת רק בסביבה מבוקרת (לא מדברים תו"כ, לא מתעסקים עם השולחן, אין מוסיקה בחדר וכו').
- על הטלפון החכם להיות רק בצידה השמאלי של המקלדת.

חשוב להבין כי מגבלות כאלה אינן יהיו קיימות לאורך זמן וברור כי ניתן התגבר עליהן, מגבלות אלו קיימות אך ורק מפני שמדובר בהוכחת יכולת, אופן המימוש וקריאת המידע מהחיישנים הוא הגורם לכך, ומי שיהיה מעוניין לבצע זאת בכדי להשיג מטרות זדוניות- יוכל לממש את המודל באופן מושלם יותר שיסבול מאילוצים הרבה פחות קשיחים.

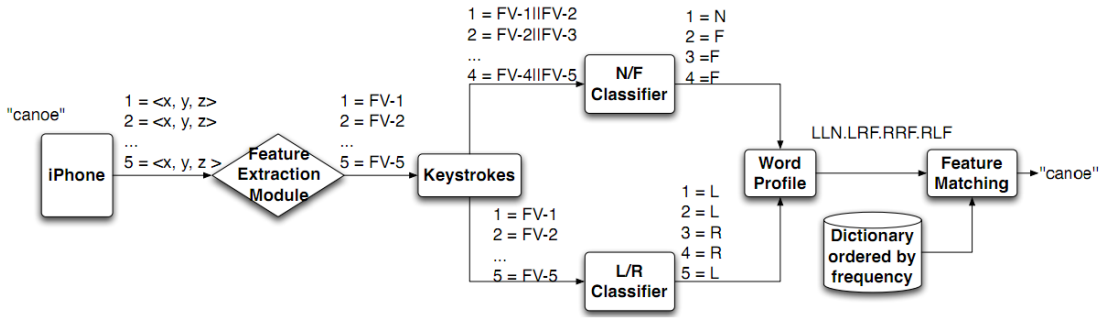
לדוגמה:

אם אקליד את המילה password האפליקציה תסווג את הקשות המקלדת באופן הבא: צמד התוים pa יזוהה כ-ימין-שמאל-רחוק, צמד התוים as כ-שמאל-שמאל-קרוב, צמד התוים ss כ-שמאל-שמאל-קרוב, הצמד sw כ-שמאל-שמאל-קרוב, ה-wo כ-שמאל-ימין-רחוק, ה-or כ-ימין-שמאל-רחוק ואת צמד התוים rd כ-שמאל-שמאל-קרוב. אם נקודד בקצרה נקבל (ימין-R, שמאל-L, קרוב-N, רחוק-F):

```
RLF-LLN-LLN-LLN-LRF-RLF-LLN
```

את הרצף הזה אפשר לנסות ולאתר במילון שמיוצג לפי אותה שיטת קידוד. ננסה דוגמה נוספת, המילה canoe. את המילה נפרק לצמדים: C-A, A-N, N-O and O-E. את הצמדים הללו נקודד ונקבל:

```
LLN-LRF-RRF-RLF
```



[תרשים זרימה של אלגוריתם התאמת המילים למילת הקלט "canoe"]

מתרשים הזרימה ניתן להתרשם מזרימת המידע ומהייצוגים השונים עד אשר משודרת המילה canoe מהתוכנה שבטלפון החכם.

כמה חמור האיום? ואיך מתגוננים מפניו?

לעת עתה השיטה יעילה כל עוד המכשיר קרוב למקלדת כ-8 ס"מ, כך שווידוא שאין טלפון חכם במרחק שכזה מהמקלדת כנראה יספיק, קל וחומר אם הטלפון נשאר בתיק, במגירה או בכיס. עדכון מדיניות האבטחה בעת התקנה של אפליקציות כך שתידרש התרת הרשאה גם לשימוש בחיישנים אינרטיים, לכל הפחות לטובת שימוש בתדירות דגימה מגובה מסויים.

באייפון, כזה שלא "שופר" באופן שמבטל את האחריות (והמבין יבין...), קשה מאוד עד לא ניתן לאפשר דגימה ושימוש בחיישנים עבור אפליקציה שרצה ברקע. באנדרואיד, זה לא המקרה שם- זה אפשרי ודיי בקלות.

המילון שנבנה הוא לאנגלית - טרם בנו אחד לעברית או לשפות "נידחות" אחרות. אמנם אין מפריע לבנות מילון שכזה לעברית, אולם יידרשו יותר משאבים לשם כך, בגלל טבעה של המורפולוגיה העברית שמאפשרת הרבה הרבה יותר נטיות וגזירות של מילים מכל שורש ותבנית (או ערך מילוני) מאשר באנגלית[6]. כמובן, שזאת אינה נחמה למי שממילא רובו ככולו של המידע הרגיש שלו הוא באנגלית.

מוסר השכל

אפשר לנסות ולהתמודד עם הגנות קריפטוגרפיות מסובכות ועם מערכות הגנה מורכבות כדי לפרוץ סיסמאות וכדי לגנוב מידע רגיש, אבל קל הרבה יותר ופשוט לעקוף את הסיבוך לגישות אחרות נטולות סיבוך מתמטי שמשתמשות בהתנהגות אנושית צפויה.

על המחבר

שלמה יונה מפתח אלגוריתמים ובשעות הפנאי עוסק בחינוך מתמטי. בנוסף, שלמה מריץ את הבלוג:
http://shlomoyona.blogspot.com

קריאה נוספת

[1] המאמר המלא:

Philip Marquardt, Arunabh Verma, Henry Carter, and Patrick Traynor. 2011. (sp)iPhone: decoding vibrations from nearby keyboards using mobile phone accelerometers. In Proceedings of the 18th ACM conference on Computer and communications security (CCS '11). ACM, New York, NY, USA, 551-562. DOI=10.1145/2046707.2046771 http://doi.acm.org/10.1145/2046707.2046771

[2] על מערכות ניווט אינרציאליות (מתוך ויקיפדיה בעברית)

[3] שיחה טכנית בגוגל של דיויד זאקס מאינוונסנס על SensorFusion

[4] ציתות למקלדת באמצעות לייזרים וולטמטרים:

A. Barisani and D. Bianco. Sniffing Keystrokes With Lasers and Voltmeters. In Proceedings of Black Hat USA, 2009.

והנה סרטון מההצגה בכנס: http://www.youtube.com/watch?v=lCKCCyLtRvQ (משעשע במיוחד!)

[5] קורס מבוא מקוון בלמידת מכונה של אוניברסיטת סטנפורד:

http://www.ml-class.org

[6] מנתח מורפולוגי לעברית מבוסס מכונות מצבים סופיות

A finite-state morphological grammar of Hebrew. Shlomo Yona and Shuly Wintner. Natural Language Engineering, Volume 14, Issue 2, April 2008, pp 173-190. (copyright Cambridge University Press)