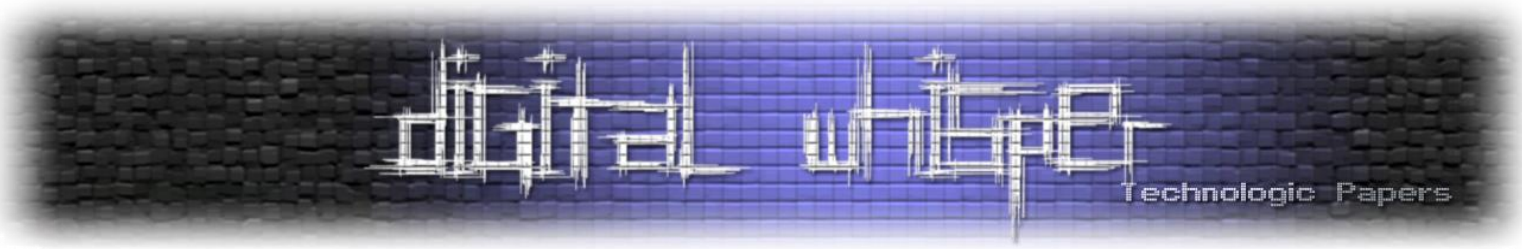


Digital Whisper

גליון 64, ספטמבר 2015

מערכת המגזין:

מייסדים:	אפיק קסטיאל, ניר אדר
מוביל הפרויקט:	אפיק קסטיאל
עורכים:	אפיק קסטיאל
כתבים:	ליאור אופנהיים, יניב בלמס, עו"ד יהונתן קלינגר וישראל (Sro) חורז'בסקי

פניות, תגובות, כתבות וכל הערה אחרת - נא לשלוח אל editor@digitalwhisper.co.il

דבר העורכים

ברוכים הבאים לדברי הפתיחה של הגיליון ה-64 של Digital Whisper!

את הגיליון הנוכחי, גליון מספר 64, הייתי מעוניין להקדיש לאדם יקר. אדם שכמעט כל גליון עוזר ומשקיע מזמנו אך עם זאת לא תמצאו אותו מופיע בדברי הפתיחה או בתודות של כמעט שום גליון, אדם מיוחד שלמרות שהוא נותן לא מעט מעצמו לטובת המגזין - אתם כמעט ולא שומעים עליו, אדם שבלעדיו ובלי תמיכה שלו, באמת אין שום סיכוי שבעולם שהפרוייקט הזה, שקוראים לו "Digital Whisper" היה ממשיך להתקיים.

חלקכם יכולים לנחש מי זאת, אך רובכם בכלל לא מכירים אותה - את הגיליון הזה אני מעוניין להקדיש לאישתי היקרה - ארִיָה.

אני לא אלאה אתכם בכל מה שהפעלת המגזין דורשת, ואני לא אשפוך בפניכם את כמות השעות החודשית הנדרשת על מנת להוציא לאור כל גליון וגליון, אך את כל אותן השעות שאני וניר משקיעים, אנו משקיעים בגלל שמדובר בתחביב שלנו, בתחביב שמהנה אותנו להתעסק בו, כמו שיש אנשים שאוספים רכבות - אנחנו (וכל מי שעוזר לנו וכותב מאמרים כמובן) מוציאים מגזין חודשי.

עם זאת, ארִיָה נאלצת להשקיע מהשעות שלה למרות שלא מדובר בתחביב אישי שלה, היא נאלצת להשקיע בעניין אך ורק בגלל שהיא מעוניינת לעזור לפרוייקט שאני חלק ממנו, ובגלל שהיא יודעת כמה חשוב לי שהוא יצליח ויתקדם.

אם זה בזמן שהיא מקדישה על מנת לעזור בעריכה (תודו שלא ידעתם, אבל היא עורכת קבועה של דברי הפתיחה של כל גליון, למרות שברב המקרים הם ג'יבריש בשבילה...), ואם זה בתמיכה מלאה בי כמוביל הפרויקט (ויש לא מעט צורך בתמיכה כשמדובר בפרויקט שכזה...), אם זה בעוד לילה לבן שהחלטתי לעשות על מנת לעמוד בלו"ז החודשי, ואם זה בעוד לא מעט דברים.

אז ארִיָה שלי, תודה רבה לך, התרומה שלך למגזין (ולחיים שלי בכלל...) היא לא עניין של מה-בכך ואינה נלקחת כמובן מאליו! בחרתי להקדיש דווקא את הגליון הזה מפני שהחודש, אנו חוגגים 5 שנות נישואין, והלוואי, יקירתי, שימשכו לנצח.

וכמובן, לפני הכל, ברצוננו להודות לכל מי שהשקיע ונתן מזמנו האישי ובזכותו הגיליון פורסם, תודה רבה **לליאור אופנהיים**, תודה רבה **ליניב בלמס**, תודה רבה **לעו"ד יהונתן קלינגר** ותודה רבה **לישראל (Sro) חורז'בסקי**!

קריאה מהנה!

ניר אדר ואפיק קסטיאל.

תוכן עניינים

How to turn your KVM into a raging Key-Logging Monster, חלק א' - שסה בי את הבינארי הטוב ביותר שלך!

מאת ליאור אופנהיים ויניב בלמס

הקדמה

סדרת מאמרים זו היא תוצר מחקר שבוצע על ידי הכותבים כחלק מעבודתם בחברת Check Point Software Technologies.

בואו נודה בזה, key-logger-ים הם מגניבים. ממש מגניבים. אפשר למצוא אותם היום בכל פינה, במחשב שלכם, בכבלים שלכם, אפילו במכונת הקפה שלכם. חלקם לגיטימיים (או שלפחות ככה אומרים :), אבל רובם לא. בכל אופן, נראה שהתחום הזה כבר חרוש לגמרי, מה אפשר לחדש כאן? מה כבר אפשר להמציא?

אז זהו, שככה גם אנחנו חשבנו, עד שבוקר בהיר אחד שמנו לב לקופסא ששוכנת לה בנוחות על השולחן, ממש מתחת למסך, וליד כוס הקפה המלוכלכת מאתמול. לקופסא הזאת קוראים KVM. למי שבמקרה לא מכיר, KVM הוא קיצור של Keyboard, Video, Mouse וכל ייעודו בחיים הוא לחבר שני מחשבים (או יותר) לאותו סט של מקלדת, עכבר ומסך. ממש פשוט.

ל-KVM-ים יש היסטוריה מכובדת בעולם המחשבים. בעבר הרחוק, KVM היה פשוט מעגל אלקטרוני שאיפשר לחבר באופן מכני את העכבר המקלדת והמסך לפורט A או לפורט B, תלוי לאיזה מצב סובבת את המתג. (ולכן הוא גם נקרא בלשון העם: 'A/B Switch').

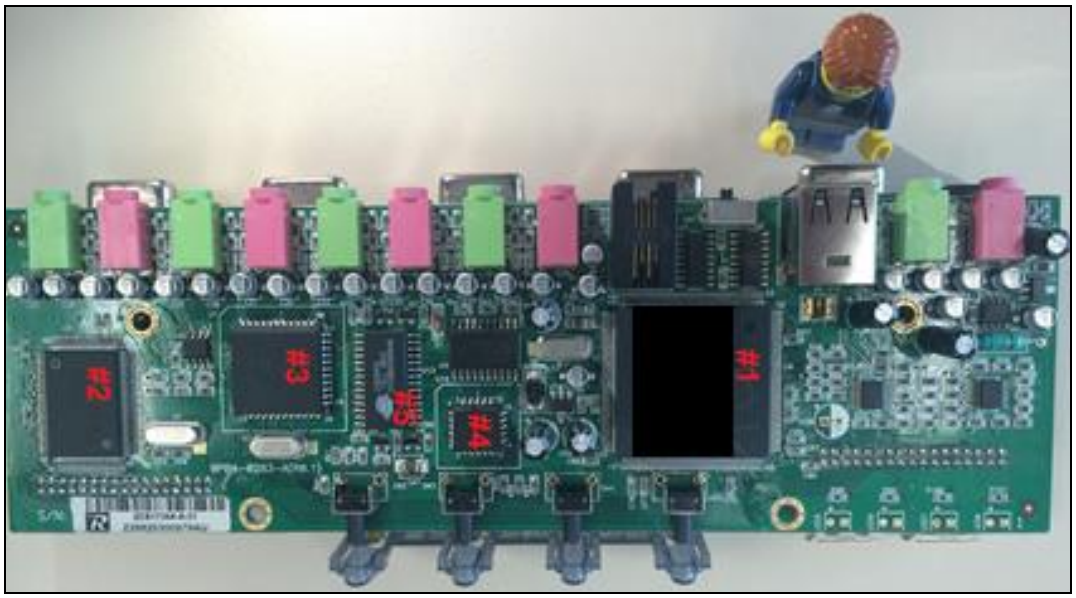
עם השנים ועם התפתחות הטכנולוגיה גם ה-KVMים נעשו הרבה יותר מתוחכמים. היום אפשר למצוא KVMים עם ממשקי קונפיגורציה שמוצגים על מסך המחשב, אפשרות להחליף את הפורטים דרך המקלדת, ואפילו ממשקי web. כמה נוח!

טוב - חשבנו לעצמנו - אז אם KVMים מודרניים הם כאלו מתוחכמים, בטח יש להם מעבד, ואם יש להם מעבד, בטח גם יש להם גם זכרון, ואם יש להם זכרון, בטח אפשר לשתול בו key-logger איכשהו. תחשבו על זה לרגע, key-logger שמותקן על KVM הוא (כמעט) בלתי ניתן לגילוי. אין עקבות על המחשב, כי הקוד

של ה-key-logger לא נמצא עליו, ואין צורך בחיבור שום חומרה חשודה נוספת. המשתמש יכול לאתחל את המחשב, הוא יכול לפרמט אותו, הוא יכול אפילו להחליף אותו במחשב חדש לגמרי, אבל כל עוד ה-KVM שם, כך גם ה-Key-logger שלנו. וכבונוס, בגלל שה-KVM נמצא בצומת של כמה מחשבים, אולי נוכל גם להקליט את ההקלדות של כל המחשבים המחוברים אליו. ואולי, רק אולי, אם נתפלל מספיק חזק לאלוהי ה-KVM, יהיה ניתן להשתמש ב-KVM כערוץ גישור בין שתי רשתות שמבודדות זו מזו ומחוברות בינהן רק דרכו. אבל עוד נגיע לזה בהמשך....

שמחים ומאושרים, עלינו על בגדי עבודה, הכנו אספקה כבדה של אלכוהול והתחלנו לעבוד.

משימה ראשונה - כדי להבין איך לעזאזל להכניס קוד שלנו לתוך ה-KVM, אנחנו צריכים קודם להבין איך בכלל הוא בנוי ואיך הוא פועל. כנראה שיש הרבה דרכים לענות על השאלה הזו, אבל הדרך האהובה עלינו כוללת מברג פיליפס וקצת אלימות.



רושם ראשוני - וואו, מי שם כל כך הרבה אלקטרוניקה בקופסא אחת?! רושם שני - יש האומרים שדברים טובים באים בקופסאות קטנות, אבל במקרה שלנו אולי יהיה יותר נכון לומר - "דברים מעניינים מגיעים בצ'יפים גדולים".

אז כדי להבין מה הולך כאן אולי כדאי להתחיל קודם למפות את הצ'יפים הגדולים שבתמונה, ולנסות להבין מי הם, ומה הם עושים:

- **צ'יפ גדול #1** - גוגל העלה חרס. אין שום מידע פרט לשם היצרן המוטבע על גבי הצ'יפ, אז ככל הנראה מדובר בצ'יפ ייעודי. קופסא שחורה.

- **צ'יפ גדול #2** - אותה תוצאה כמו צ'יפ #1, רק הפעם בצורה של מלבן. עד עכשיו, לא התקדמות מזהירה...
- **צ'יפ גדול #3** - מעבד Winbond 8052. יש! מעולה. למי מכם שלא מכיר, 8052 הוא מעבד מאוד נפוץ בעולם ה-Embedded שמבוסס על ארכיטקטורת intel 8051 (כמו intel 8086, אבל שונה לגמרי). למעבד יש ROM מוטמע בתוכו אשר מכיל את הקוד המורץ (כלומר ה-firmware, להלן "קושחה").
- **צ'יפ גדול #4** - PLD מבית Atmel. גוגל מגלה לנו שPLD הם ראשי תיבות של Programmer Logic Device. סך הכל מדובר ברכיב שניתן לצרוב עליו מעגלים דיגיטליים "לבקשתך", כך שהצ'יפ מבצע לוגיקה מסויימת, שכרגע, אין לנו מושג מהי.
- **צ'יפ גדול #5** - SRAM מבית Lyontek. או במילים אחרות - זיכרון.

למי שבמקרה דילג על הקטע הטכני המתיש לעיל, הנה תקציר - מסתבר שיש מעבד embedded נפוץ בתוך ה-kvm שלנו, ויש לנו הרגשה שהוא בעצם האחראי על הלוגיקה הפנימית של ה-KVM (מין ניחוש מושכל שכזה).

כל שנותר לנו לעשות עכשיו הוא לנסות להשיג את הקושחה של הצ'יפ, לנתח אותה ואז אולי נוכל לטעון למכשיר קושחה חדשה משלנו, ולהתחיל להשתשתע עם המכשיר.

למזלנו, אתר היצרן של ה-KVM שלנו מאפשר הורדת עידכוני קושחה בקלות. טכנית, העדכון עצמו מתבצע דרך כבל סיריאלי המחבר בין המחשב ל- KVM . נחסוך מכם את הניתוח של תוכנת העדכון, ורק נגיד שאחרי כמה מנגנוני הגנה (ובזכות כלי העזר המדהים ל-IDA - DIE) הצלחנו לחלץ מהזיכרון את הקושחה.

במבט חטוף ובעין בלתי מזויינת נראה שהקושחה שחילצנו דחוסה או מוצפנת באופן כלשהו. ערכי האנטרופיה שלה די גבוהים, דבר שתומך בהנחה הזו. גם כל ניסיונות פתיחה שלה ב-IDA (כ-8051 או ככל דבר אחר) עלו בתוהו. אז במקום לצלול לעומק הבינארי ולנסות להבין מה הולך כאן, פשוט לקחנו את הקושחה המחולצת ועם חיוך טיפשי הרצנו אותה ב-binwalk[1] כדי לזהות את סוג הקושחה ו\או מאפיינים בינאריים מעניינים אחרים.

אבל מהר מאוד החיוך הזחוח נמחק מפנינו כשראינו ש-binwalk לא באמת הצליח לזהות את הקושחה, את שיטת הדחיסה, או בעצם כלום, פשוט שום דבר! 0 תוצאות!

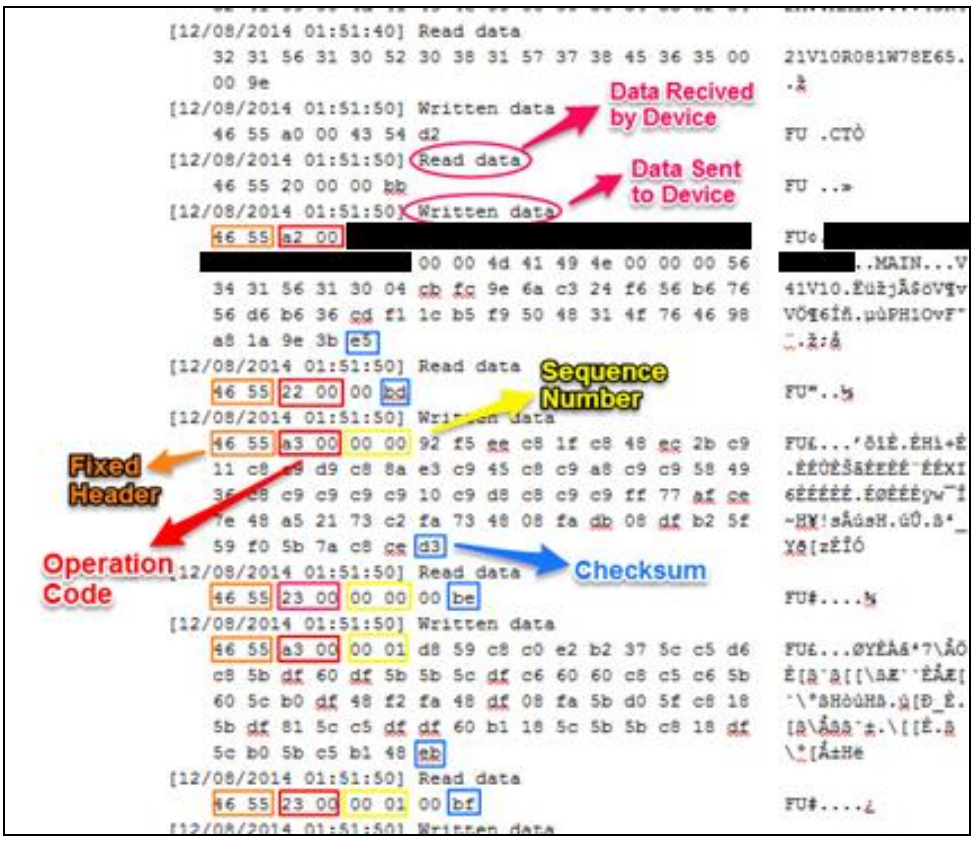
מה עושים?! הנחת העבודה שלנו היא שהקושחה צריכה להפתח מתישהו בתוך תוכנת העדכון, ואז להשלח בצורתה הפתוחה על גבי הכבל הסיראלי הישר אל תוך המכשיר. אם כך, אולי ננסה להיות קצת יותר יצירתיים ובמקום לחקור את תוכנת העידכון, ננסה להסניף את התעבורה שנשלחת ומתקבלת בפורט

[1] Binwalk - כלי נפוץ לזיהוי מגוון רחב של קושחות ידועות, שיטות דחיסה ושאר ירקות.

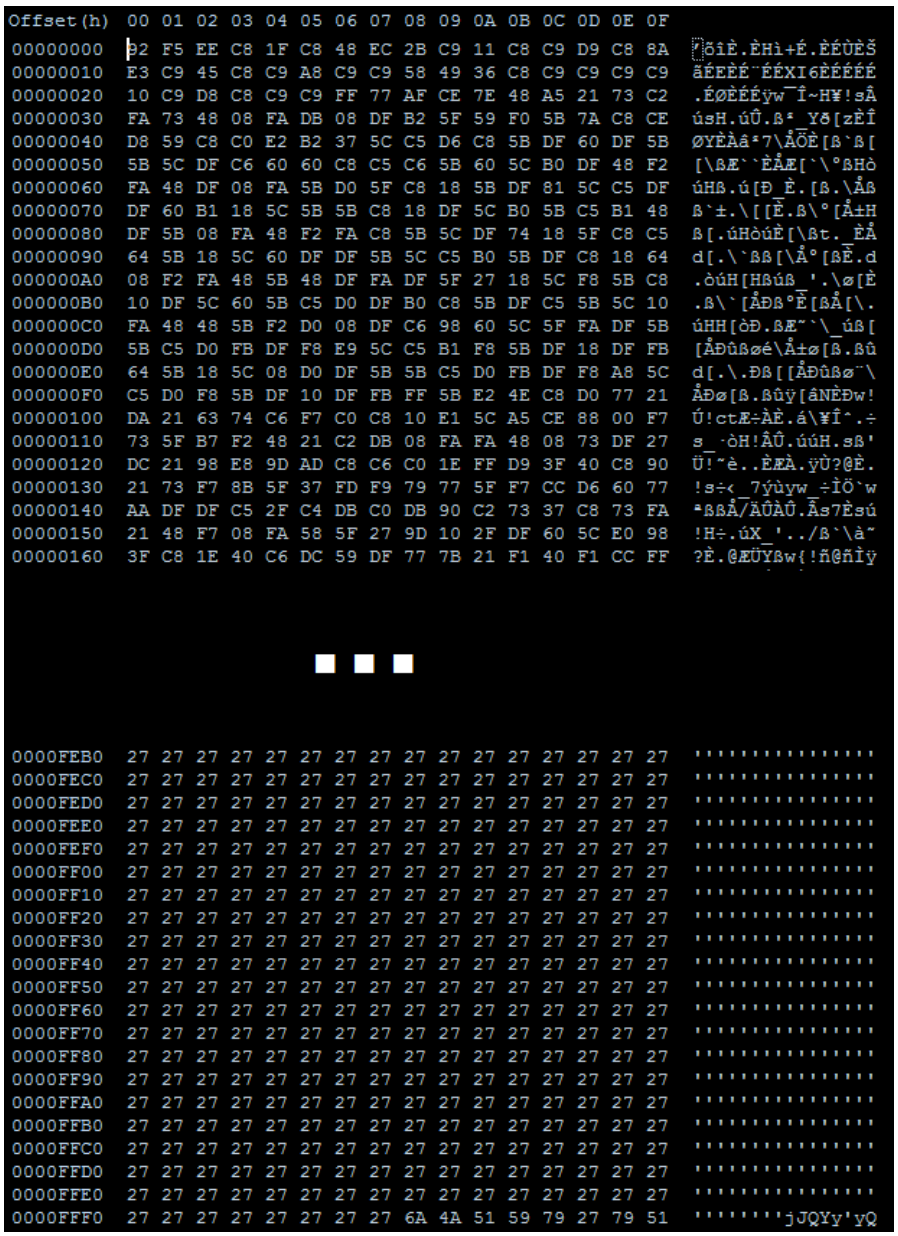
הסיריאלי. כך בעצם נוכל לחלץ את הקושחה, לאחר שנפתחה ו\או פוענחה, מבלי לדעת את אלגוריתם הדחיסה/הצפנה! נשמע כמו רעיון טוב.

אז הורדנו איזה תוכנה גנרית להסנפה של הפורט הסיריאלי, הרצנו את העידכון והתחלנו להסניף. מהר מאוד ראינו שעל גבי הפורט הסיריאלי מועבר איזשהו פורטוקול, שכפי שאמרנו כנראה מכיל בתוכו את הקושחה המקורית.

כמה כוסות קפה אחרי, והפרוטוקול הסיריאלי נותח באופן מלא:



כמה כוסיות ווסקי אחרי, ויש לנו כלי פייתוני לחילוץ המידע מתוך הפרוטוקול:



```
Offset(h)  00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F

00000000  92 F5 EE C8 1F C8 48 EC 2B C9 11 C8 C9 D9 C8 8A  'õîÈ.ÈHì+É.ÈÉÙÈŠ
00000010  E3 C9 45 C8 C9 A8 C9 C9 58 49 36 C8 C9 C9 C9 C9  ãÉEÈÉ¨ÉÉXI6ÈÉÉÉÉ
00000020  10 C9 D8 C8 C9 C9 FF 77 AF CE 7E 48 A5 21 73 C2  .ÉØÈÉÉÿw¯Î~H¥!sÂ
00000030  FA 73 48 08 FA DB 08 DF B2 5F 59 F0 5B 7A C8 CE  úsH.úÛ.ß²_Yð[zÈÎ
00000040  D8 59 C8 C0 E2 B2 37 5C C5 D6 C8 5B DF 60 DF 5B  ØYÈÀâ²7\ÅÖÈ[ß`ß[
00000050  5B 5C DF C6 60 60 C8 C5 C6 5B 60 5C B0 DF 48 F2  [\ßÆ``ÈÅÆ[`\°ßHò
00000060  FA 48 DF 08 FA 5B D0 5F C8 18 5B DF 81 5C C5 DF  úHß.ú[Ð_È.[ß.\Åß
00000070  DF 60 B1 18 5C 5B 5B C8 18 DF 5C B0 5B C5 B1 48  ß`±.\[[È.ß\°[Å±H
00000080  DF 5B 08 FA 48 F2 FA C8 5B 5C DF 74 18 5F C8 C5  ß[.úHòúÈ[\ßt._ÈÅ
00000090  64 5B 18 5C 60 DF DF 5B 5C C5 B0 5B DF C8 18 64  d[.\`ßß[\Å°[ßÈ.d
000000A0  08 F2 FA 48 5B 48 DF FA DF 5F 27 18 5C F8 5B C8  .òúH[Hßúß_'.\ø[È
000000B0  10 DF 5C 60 5B C5 D0 DF B0 C8 5B DF C5 5B 5C 10  .ß\`[ÅÐß°È[ßÅ[\.
000000C0  FA 48 48 5B F2 D0 08 DF C6 98 60 5C 5F FA DF 5B  úHH[òÐ.ßÆ˜`\_úß[
000000D0  5B C5 D0 FB DF F8 E9 5C C5 B1 F8 5B DF 18 DF FB  [ÅÐûßøé\Å±ø[ß.ßû
000000E0  64 5B 18 5C 08 D0 DF 5B 5B C5 D0 FB DF F8 A8 5C  d[.\.Ðß[[ÅÐûßø¨\
000000F0  C5 D0 F8 5B DF 10 DF FB FF 5B E2 4E C8 D0 77 21  ÅÐø[ß.ßûÿ[âNÈÐw!
00000100  DA 21 63 74 C6 F7 C0 C8 10 E1 5C A5 CE 88 00 F7  Ú!ctÆ÷ÀÈ.á\¥Î^.÷
00000110  73 5F B7 F2 48 21 C2 DB 08 FA FA 48 08 73 DF 27  s_·òH!ÂÛ.úúH.sß'
00000120  DC 21 98 E8 9D AD C8 C6 C0 1E FF D9 3F 40 C8 90  Ü!˜è..ÈÆÀ.ÿÙ?@È.
00000130  21 73 F7 8B 5F 37 FD F9 79 77 5F F7 CC D6 60 77  !s÷‹_7ýùyw_÷ÌÖ`w
00000140  AA DF DF C5 2F C4 DB C0 DB 90 C2 73 37 C8 73 FA  ªßßÅ/ÄÛÀÛ.Âs7Ès ú
00000150  21 48 F7 08 FA 58 5F 27 9D 10 2F DF 60 5C E0 98  !H÷.úX_'../ß`\à˜
00000160  3F C8 1E 40 C6 DC 59 DF 77 7B 21 F1 40 F1 CC FF  ?È.@ÆÜYßw{!ñ@ñÌÿ

                          ■ ■ ■

0000FEB0  27 27 27 27 27 27 27 27 27 27 27 27 27 27 27 27  ''''''''''''''''
0000FEC0  27 27 27 27 27 27 27 27 27 27 27 27 27 27 27 27  ''''''''''''''''
0000FED0  27 27 27 27 27 27 27 27 27 27 27 27 27 27 27 27  ''''''''''''''''
0000FEE0  27 27 27 27 27 27 27 27 27 27 27 27 27 27 27 27  ''''''''''''''''
0000FEF0  27 27 27 27 27 27 27 27 27 27 27 27 27 27 27 27  ''''''''''''''''
0000FF00  27 27 27 27 27 27 27 27 27 27 27 27 27 27 27 27  ''''''''''''''''
0000FF10  27 27 27 27 27 27 27 27 27 27 27 27 27 27 27 27  ''''''''''''''''
0000FF20  27 27 27 27 27 27 27 27 27 27 27 27 27 27 27 27  ''''''''''''''''
0000FF30  27 27 27 27 27 27 27 27 27 27 27 27 27 27 27 27  ''''''''''''''''
0000FF40  27 27 27 27 27 27 27 27 27 27 27 27 27 27 27 27  ''''''''''''''''
0000FF50  27 27 27 27 27 27 27 27 27 27 27 27 27 27 27 27  ''''''''''''''''
0000FF60  27 27 27 27 27 27 27 27 27 27 27 27 27 27 27 27  ''''''''''''''''
0000FF70  27 27 27 27 27 27 27 27 27 27 27 27 27 27 27 27  ''''''''''''''''
0000FF80  27 27 27 27 27 27 27 27 27 27 27 27 27 27 27 27  ''''''''''''''''
0000FF90  27 27 27 27 27 27 27 27 27 27 27 27 27 27 27 27  ''''''''''''''''
0000FFA0  27 27 27 27 27 27 27 27 27 27 27 27 27 27 27 27  ''''''''''''''''
0000FFB0  27 27 27 27 27 27 27 27 27 27 27 27 27 27 27 27  ''''''''''''''''
0000FFC0  27 27 27 27 27 27 27 27 27 27 27 27 27 27 27 27  ''''''''''''''''
0000FFD0  27 27 27 27 27 27 27 27 27 27 27 27 27 27 27 27  ''''''''''''''''
0000FFE0  27 27 27 27 27 27 27 27 27 27 27 27 27 27 27 27  ''''''''''''''''
0000FFF0  27 27 27 27 27 27 27 27 6A 4A 51 59 79 27 79 51  ''''''''jJQYy'yQ
```

מצוין! נראה שהקושחה לא דחוסה יותר - ערך האנטרופיה ירד משמעותית. כנראה שאנחנו בכיוון הנכון. חדי האבחנה בינכם גם כנראה ישימו לב לכך שהבית 0x27 חוזר על עצמו בסוף הקובץ. עוד סימן נהדר לכך שהמידע לא דחוס יותר.

אולי עכשיו נוכל לקבל תוצאות טובות יותר מ-binwalk? טוב, אז כנראה זה לא הולך להיות כל כך קל... binwalk ממשיך לסרב בתוקף לזהות את סוג הקושחה או כל דבר אחר, וכנ"ל IDA.

ניסינו לחזור על אותה הבדיקה עם גרסאות עידכון שונות וכצפוי קיבלנו בדיוק את אותן התוצאות. ההבדל הניכר היחידי בין התוצאות היה בבית הריפוד (זה עם הערך 0x27 בתמונה למעלה), ערך הריפוד משתנה בין כל גרסה.

אז למה קיימים ריפודים שונים בגרסאות שונות? הסיבה היחידה שיכולנו לחשוב עליה היא שמדובר בשיטת קידוד פשוטה ומטרתה היחידה היא למנוע מאיתנו לצפות בקוד האמיתי בקלות. ולכן, מכיוון שריפוד באפסים נראה יותר טוב בעין, נשמע לנו מאוד הגיוני לנסות לקסר (פועל: XOR) את כל הקובץ בבית האחרון הזה, וכך לאפס את סוף הקובץ, ובתקוה כך גם שאר הקובץ יהפוך למשהו יותר הגיוני.

אולי הפעם באמת הצלחנו? האם יש לנו עכשיו קוד קריא?

אז זהו, שלא.

אם נסתכל על חצי הכוס הריקה binwalk עדיין לא מחזיר שום תוצאה וגם IDA לא מביאה איזו בשורה מרעננת, אבל אם נסתכל על חצי הכוס המלאה, התגלה לנו משהו די מעניין.

כשמשווים את כל הגרסאות השונות, לאחר פעולת הקיסור, מתגלה דפוס תדירויות מאוד דומה בין הגרסאות. כלומר, אותם הבתים הופיעו מספר דומה של פעמים על פני כל הגרסאות באופן עקבי. זה כנראה אומר שעשינו צעד בכיוון הנכון, כל הגרסאות כתובות עכשיו באותה ה"שפה", כל מה שנשאר לנו להבין הוא איך לתרגם את השפה הזו לקוד בעל משמעות.

למרות שהשגנו התקדמות מסוימת, נותרו המון שאלות פתוחות והפיתרון עדיין לא נראה באופק. כנראה שזה הזמן הנכון לרדת לברזלים[2].

המשך יבוא...

נ.ב.

לאלו מכם שהגיעו עד לחלק הזה במאמר ואינם יכולים להתאפק, הלינק הבא מכיל את גרסת הקושחה במצבה המקורי. אתם מוזמנים לנסות את מזלכם וכישוריכם האישיים ולנסות להפוך אותה לקוד אמיתי (כן, זה לגמרי אפשרי). בהצלחה!

[2] **ברזל** - הוא יסוד כימי שסמלו הכימי Fe ומספרו האטומי 26. הוא גם כינוי נפוץ לשכבה הנמוכה ביותר האפשריתבמערכת כלשהי.

בנק המטרות שהממשלה מפרסמת

מאת עו"ד יהונתן קלינגר

הקדמה

חוק הגנת הפרטיות הישראלי הוא קצת מיושן, נחקק ב-1981 ושונה מעט מאוד מאז. מה שמצחיק עוד יותר הוא פרק ההגנה על פרטיות במאגרי מידע בחוק. למרות שההגדרה של מהו מאגר מידע השתנה מהיסוד מאז 1981, והשימושים ב-big-data עוד יותר, הרי שבשנת 1996 התחלפה ההגדרה של מהו מאגר מידע (שהיה בעבר "מרכז להחסנת מידע באמצעות מערכת עיבוד נתונים אוטומטית").

מה שלא השתנה בחוק מאז חקיקתו הוא עניין הרישום: כל מאגר מידע (עוד שניה נגיע להגדרה מהו מאגר מידע), חייב להרשם אצל "רשם מאגרי המידע" ולעדכן בטופס איזה פרטים הוא שומר, וכיצד הוא אוסף אותם. הפעולה הזו אינה רק טכנית אלא גם מהותית: היא נועדה לתת הכשר לדרך העבודה לפחות ברמה המקדמית.

בפועל, רוב האנשים שמנהלים מאגרי מידע לא רושמים את המאגר (ועוברים על החוק), אפילו משרדי ממשלה שחייבים לרשום את מאגרי המידע שלהם לא עושים כן.

אז מהו מאגר מידע? החוק (שתוקן ב-1996) מגדיר את המאגר כ"אוסף נתוני **מידע**, המוחזק באמצעי מגנטי או אופטי והמיועד לעיבוד ממוחשב" ומחריג מההגדרה "אוסף לשימוש אישי שאינו למטרות עסק" או "אוסף הכולל רק שם, מען ודרכי התקשרות, שכשלעצמו אינו יוצר איפיון שיש בו פגיעה בפרטיות לגבי בני האדם ששמותיהם כלולים בו, ובלבד שלבעל האוסף או לתאגיד בשליטתו אין אוסף נוסף".

מה זה "מידע" שנאגר באוסף? "נתונים על אישיותו של **אדם**, מעמדו האישי, צנעת אישותו, מצב בריאותו, מצבו הכלכלי, הכשרתו המקצועית, דעותיו ואמונתו". (אדם לא כולל תאגיד, כאמור בסעיף 3 לחוק הגנת הפרטיות) כלומר, בעוד שההגדרה ה"מחשבית" ל-Database היא פשוטה מאוד, ב"עורכדינית" צריך להבדיל בין "מסד נתונים" לבין "מאגר מידע" (כלומר Data מול Information). לדוגמא, ספרי טלפונים לכשעצמם, שלא מכילים נתונים על מעמד אישי, מצב בריאותי או אפיון כלשהוא כלל אינם מוגדרים כמאגר מידע בחוק, ולא רק שלא יהיו חייבים ברישום, גם לא בהכרח שיהיו חייבים באבטחה הראויה, ומותר יהיה (לכאורה, ובכפוף לסעיף 2 לחוק הגנת הפרטיות) לסחור בהם.

מכאן, נתחיל בסיפור שלנו.

כאמור, יש בישראל חובה לרשום את המאגרים, במיוחד אם המאגר משמש לדיוור ישיר או לפרסום. יש עוד זכות משמעותית והיא "זכות העיון"; סעיף 13 לחוק נותן לכל אחד את הזכות לעיין במאגרים שמחזיקים במידע עליו. העיון כפוף לאגרה, שנקבעה בתקנות הגנת הפרטיות, אבל היא של 20 שקלים (לא חדשים). כלומר, לכל אחד בפועל יש זכות לעיין במידע במשרדי מחזיק המאגר, או לקבל פלט.

עכשיו, איך תדע באיזה מאגר יש עלייך מידע?

על פי סעיף 12 לחוק הגנת הפרטיות, לכל אדם יש זכות לעיין בפנקס מאגרי המידע. מדובר על אותו פנקס שבו אנשים (שמכבדים את החוק) רשמו את המאגר שלהם בו. העיון בפנקס הוא עיון במסמך ממשלתי, ולכן הוא ללא תשלום.

לאחר שעיינת בפנקס מאגרי המידע, ניתן לפנות אחד אחד לבעלי המאגרים שמופיעים בו (כ-15,000 מאגרים) ולבקש לעיין במידע עלייך (או לקבל אישור שאין מידע כזה). העיון במידע עדיין לא מאפשר מחיקה שלו, אבל הוא כן מאפשר לדעת מי יודע עלייך מה: לדוגמא, אם במאגר של חברה מסוימת שמכרה לך ביטוח חיים יש מידע מסוים, ואתה רואה אותו גם בחברה שמשווקת מכשירים סלולריים, שנמצאת בבעלות של אותה חברת אחזקות, אז כנראה שהם החליפו מידע.

לכן, ביקשתי לפני כחודש את פנקס מאגרי המידע ממשרד המשפטים, ובצורה מפתיעה ויעילה אפילו קיבלתי אותו. בלינק הזה תוכלו לקבל גרסא אונליין של הפנקס.

אז מה השימושים האפשריים בפנקס, עכשיו שהוא נמצא בידינו?

1. **ניהול מערכת מבוזרת מבוססת המונים לאיתור ספאמרים ומטרידים**. לא אחת אנחנו מקבלים שיחות טלפון או מסרונים מטרידים מאנשים שאומרים שהם קיבלו את המידע שלנו בגלל X או Y. אם נניח מתקשרת נציגה מחברה סלולרית מסוימת ואומרת שיש לה מבצע לחברי קופת חולים כללית.

   אתה שואל את הנציגה מהיכן הפרטים שלך, והיא אומרת שהיא קיבלה אותם מקופת חולים.

   בשלב הזה, אתה יכול ללכת למערכת המאגר ולבדוק האם לקופת החולים יש מאגר, ואז לעיין במאגר הזה ולראות האם יש העברה של המידע שלך החוצה.

   ברגע שתגלה שקופת החולים העבירה מידע, תוכל לסמן גם את מספר הטלפון של המתקשר כספאמר, וגם את קופת החולים כמי שלא מכבדת את הפרטיות שלך ומוכרת את המידע למשווקים. תוכל גם, במקביל, לעיין בכל המאגרים האחרים ולראות מי מהחברות שאתה עושה איתן עסקים לא מכבדות את הפרטיות שלך.

   את המידע הזה, תוכנות כמו WeNoSpam יוכלו לנצל. למי שלא מכיר, WeNoSpam היא אפליקציה סלולרית שעובדת בצורה פשוטה מאוד: כאשר אתה מקבל שיחה ממספר שלא שמור ברשימת אנשי

הקשר שלך, היא שולחת שאילתא לשרת, בשאילתה היא שואלת "האם מספר הטלפון X הוא מספר של ספאמרים?". אם מתקבלת תשובה חיובית, היא חוסמת את השיחה (או המסרון).

אם אתה מדווח על מספר טלפון כמספר של ספאמרים, זה גם נרשם.

2. **איתור פרטי ספאמרים**. ביחד עם המידע מפנקס המאגרים, אפשר יהיה לא רק לזהות את הספאמר, אלא גם לאסוף את הפרטים שלו ולאפשר תביעה מהירה יותר. במאגר תוכלו לראות את פרטי השולח, כולל ח.פ וכדומה. אם, לדוגמא, קיבלתם SMS מחברה המציעה לך להשתתף בהגרלת לוטו, ובשיחה הצלחתם לדלות מספיק פרטים, תוכלו להשתמש בפנקס כדי לאמת חלק מהפרטים ולבקש לאחר מכן לעיין במידע שיש להם.

אחת הבעיות עם ספאם טלפוני היא שאנחנו לא יודעים מי השולח ומהיכן יש לו את הפרטים שלנו. ולכן, אחרי שקיבלנו הודעה טלפונית צריך לכתת רגלים כדי לתבוע.

ברוב המקרים הודעת הספאם תוביל ללינק אלמוני שיגיע לעמוד נחיתה, וגם כאשר נקבל שיחה מהספמארים ברגע שהם יבינו שמנסים להוציא מהם מידע הם ינתקו.

בפנקס המאגרים יש לנו את רשימת בעלי המאגר, וניתן יהיה להצליב את המספרים באמצעות Reverse Lookup.

368627	512963646	לוטו זהב מועדון מנויים בע"מ	לקוחות	ירושלים	25			9430412	כן	מאוש
368664	520036112	קרגל בע'מ	מאגר נתוני שכר	לוד			65	7110001		מותלה ח
368749	520036799	חברה לשיווק והספקה לבנין בע"מ	הנהלת חשבונות	תל אביב - יפו	לינקולן	20		6713412		מאוש
369364	511130221	ח.ל. גרינשטיין בע"מ	ח.ל גרינשטיין בע"מ	נתניה	דיזנגוף	47	0	4243801		מאוש
369578	500500376	קרנית קרן לפיצוי נפגעי תאונות דר	"משכורות עובדי חברת "אבנר	תל אביב - יפו			37505	6137401		מאוש
369730	511705535	ק.ו. פרוגרס בע"מ	ללא שם	בת ים	ספטל גיורא	64		5950737		מאוש
369753	520003781	שטראוס גרופ בע"מ	לקוחות שטראוס גרופ	פתח תקווה	הסיבים	49	149	4959504		מאוש
369907	61199	שירותי בריאות כללית	תמר- בי"ח כרמל	תל אביב - יפו	ארלוזורוב	115	16250	6209813		מאוש
370028	580178085	(ח.ס.ן - חולי סרטן נפגשים (ע"ר	חברי העמותה ונמניה	תל אביב - יפו	הירקון	169		6345325	כן	מותלה ח
370378	500277009	עירית עפולה	רישוי עסקים	עפולה			2016			מאוש

3. **"מה יודעים עליי"**. כזכור, על פי סעיף 13 לחוק לכל אחד את הזכות לעיין במאגרים שמחזיקים במידע עליו. מה הבעיה? אנחנו לא יודעים מי המאגרים שמחזיקים עלינו מידע. לכאורה, בעלות "סמלית" של שליחת מכתב רשום לכל אחד מהמאגרים, אפשר לטפל בבעיה הזו. אלא, שאם יש 15,000 מאגרים, ושליחת דואר רשום עולה (בקירוב) 10 ש"ח, מדובר על 150,000 ש"ח.

אבל, אפשר לרכז את הפניות. אתר מרוכז שיקום יכול לשלוח את כל הפניות לכל המאגרים מכל המבקשים החודשיים ולחסוך בעלות.

הפתרון? מרימים אתר אינטרנט שבו כל אחד שמעוניין לקבל את התיק האישי שלו מכל המאגרים נרשם. בסוף החודש מאגדים את כל הנרשמים, ובודקים: אם יש 1,500 נרשמים, אז בעלות סמלית

של 100 ש"ח לאחד אפשר לשלוח את החבילה הרשומה לכל בעלי המאגרים ולבקש שישלחו את התיק האישי לכל אחד מהאנשים שברשימה.

החסרון במצב כזה הוא יצירה של מאגר של מבקשים, שעשוי להיות יעד סייבר לאחר מכן. אבל החסרון הוא יחסית זניח בהתחשב בתועלת בעיון.

4. **ספר/מועדון לקוחות**. אם אתה עצמאי או חברה בתחום אבטחת המידע, ברור לך שלקוחות חדשים קשים להשגה, ולכן אתה צריך תמיד למצוא אל מי לפנות. כאן יש לך רשימה איכותית של חברות שמחזיקות מאגרי מידע וצריכות שירותים סביב אותם המאגרים: אבטחה שוטפת, ייעוץ, אופיטמיזציה וכדומה. מה טוב מזה כדי להושיב טלפנית שתתקשר אחד אחד לכל הגורמים לקביעת פגישה?

5. **בנק מטרות**. כמובן, שהמשתמש הזדוני לא ממש חייב להיות מלאך שינסה למכור שירותי אבטחת מידע. משתמש זדוני מספיק יכול להבין שפנקס המאגרים הוא בנק מטרות, שיכול לעזור לו לנסות לדלות מידע על אנשים בין אם באמצעות הנדסה חברתית ובין אם באמצעות פריצה למאגרים.

ומה אתה יכול לעשות בפועל עם המידע?

1. **קודם כל, אתה יכול לבקש להסיר את עצמך מחלק מהמאגרים**. סעיף 17ו לחוק הגנת הפרטיות מאפשר לך לבקש מחיקה ממאגרים שמשמשים לדיוור ישיר. לצערנו, הפסיקה לא קבעה שספאם הוא דיוור ישיר, אלא דיוור ישיר הוא "פניה אישית לאדם, בהתבסס על השתייכותו לקבוצת אוכלוסין, שנקבעה על פי איפיון אחד או יותר של בני אדם ששמותיהם כלולים במאגר מידע"; כלומר, כל פניה שעושה פרופיילינג. לכן, אתה יכול לבדוק איזה מהמאגרים מוגדר כמאגר לדיוור ישיר ולבקש להמחק.

2. מהמידע שאתה לא יכול להסיר, אתה יכול **לתקן**. סעיף 14 לחוק הגנת הפרטיות נותן לך את הזכות לתקן מידע לא נכון שמופיע במאגרים. זה אומר שאת כתובת המייל שלך ומספר הטלפון שלך אתה יכול לתקן לכתובת המייל שלך לצורך ספאם ולמספר הטלפון שלך לצורך ספאם, בהנחה שאתה באמת מחזיק כאלה. גולן טלקום, לדוגמא, מאפשרים לך להחזיק מספר טלפון וירטואלי רק לצורך כזה.

3. **מידע שאתה לא יכול להסיר ולא יכול לתקן (כי הוא נכון) אתה יכול ליידע**. הסכמה, על פי חוק הגנת הפרטיות, ניתנת לחזרה (09-1222 ורדי נ' גוטסמן, השתנה בערעור, ע"א 1697/11 א. גוטסמן אדריכלות בע"מ ואח' נ' אריה ורדי, וגם תא 6023/07 אפריאט נ' ידיעות אחרונות). כלומר, אתה תמיד יכול לפנות למי שלא מוכן למחוק ולומר לו "אדון נכבד, אני לא מסכים שתעביר את המידע עליי ואני לא מוכן שתצור איתי קשר", גם אם יצירת הקשר היא לא לפי חוק הספאם (סעיף 30א לחוק התקשורת).

מה אתה יכול לעשות אם הארגון שלך מופיע ברשימה?

ככל הנראה, שאם הארגון שלך מופיע ברשימה אז אתם מחזיקים מאגר מידע. זה לא אומר שאתם ספאמרים, זה לא אומר שאתם גנבים, זה לא אומר שום דבר חוץ מזה שאתם מחזיקים מאגר שחייב ברישום. עכשיו, יכול להיות שתחליטו, בתור ארגון, לערוך בדיקה שוב ולבדוק האם באמת צריך להחזיק את המאגר.

האם באמת כל מי שמחזיק מערכת הנהלת חשבונות צריך רישום?

חלק ניכר מהמאגרים ברשימה הם מאגרי שכר של מקומות עבודה. זה טוב אם אתה עובד לשעבר שרוצה לעיין במידע, אבל האם זה אומר שכל מי שמנהל חשבות שכר צריך לרשום מאגר? בפועל, לא בהכרח.

הדרישה לרישום מאגר היא בהתקיים אחד מהתנאים הבאים: מספר האנשים שמידע עליהם נמצא במאגר עולה על 10,000; יש במאגר מידע רגיש; המאגר כולל מידע על אנשים והמידע לא נמסר על ידיהם, מטעמם או בהסכמתם למאגר זה; המאגר הוא של גוף ציבורי כהגדרתו בסעיף 23; המאגר משמש לשירותי דיוור ישיר כאמור בסעיף 17ג (סעיף 8 לחוק הגנת הפרטיות).

מבחינתנו, כאשר מקום העבודה מכיל פחות מ-10,000 עובדים, וכאשר העובדים הסכימו לאגירת המידע בהיותם עובדים, התנאי הוא שיש מידע רגיש. ומהו מידע רגיש על פי חוק הגנת הפרטיות? על פי סעיף 7, מידע רגיש הוא "נתונים על אישיותו של אדם, צנעת אישותו, מצב בריאותו, מצבו הכלכלי, דעותיו ואמונתו" (או מידע שנקבע על ידי שר המשפטים שהוא מידע רגיש).

כל עוד מעסיק ינקוט בגישה מינימאליסטית ויסרב לשמור מידע רפואי על המחלות של העובדים (מה שהתקבל לאחרונה כחובה), ולא ישמור מידע על הרקע הכלכלי או אמונות שלהם (בפועל: אם הוא לא יעקוב אחריהם במקום העבודה), ומאגר המידע שלו כמעביד יכלול רק את הנוכחות של העובד ואת השכר, אז ככל הנראה שאפשר יהיה להמנע מהרישום. במצב כזה, מעסיקים רבים יוכלו להמחק מפנקס המאגרים ויחסכו חובות רבות (אך לא את החובה לאבטח את המידע).

לסיכום

חבל שעד היום לא פורסם פנקס מאגרי המידע ברבים. יש לפרסום שלו תועלת רבה גם לשקיפות השלטונית, אבל גם ליישומים ושימושים אזרחיים נוספים. אני מקווה שאני לא אצטרך לפנות בבקשות גילוי רבעוניות אלא אצליח לעודד את המערכת לפרסם את המאגר בצורה ייזומה. לבינתיים, יש לכם מאגר פתוח לשימוש.

HTTP/2 - הבנה וניטור של תקשורת העתיד

נכתב ע"י ישראל (Sro) חורז'בסקי, CTO, AppSec Labs)

פרולוג

כל המשפחה התיישבה רגל על רגל והסתובבה לכיוון הקיר. המסך ירד והמקרן החל לפעול. לשבת בבית קפה שהוקצה במיוחד ליום שלם של עבודה משותפת ומהנה, להזמין קצת נשנוש ושתייה ובמשך שעה שלמה לשמוע רק הרצאות על נושאים טכנולוגיים חדשים שהיו ארוכים מכדי להיכנס למייל. תענוג.

המאמר מכיל את אחד הנושאים שהוצגו ב-Tech talk הפנימי של משפחת AppSec בחודש האחרון, והוא מכיל 2 חלקים: הסבר על מה שהשתנה ב-http/2, ואיך לראות תעבורה של http/2 (רמז: לא באמצעות Burp/fiddler).

פינת היסטוריה - SPDY is dead

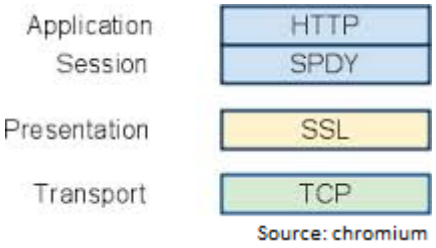
מי שלא מתלהב מהיסטוריה, יכול לעבור לעמוד הבא ולקרוא ישירות על HTTP/2. למי שכן, נעשה מעט סדר כרונולוגי.

התקן של פרוטוקול HTTP/1.1 התפרסם בשנת 1997, ומאז לא השתנה עד השנה. במובן הזה, HTTP2 (ליתר דיוק HTTP/2) הוא אחד השינויים הכי גדולים בפרוטוקול שהיו בעשרים השנים האחרונות.

בשנים שחלפו מאז שהוגדר התקן של HTTP 1.1, עולם האינטרנט השתנה משמעותית, וכיום כל דף אינטרנט שעולה בדפדפן שולח המון בקשות לשרת (ajax, css, images, ico, js, cors ועוד). כך שמעבר לעובדה שהפרוטוקול טקסטואלי - מה שיוצר בקשות גדולות (וזה בלי להגיד מילה על ה-abuse שעושים לקוקיז כשמכניסים אליהם כמויות של מידע...), כיוון שיש הרבה חיבורים מול השרת נוצר בדפדפן "תור" של בקשות. הדפדפן מגביל את כמות החיבורים פר שרת כדי לא ליצור עומס, וכשיש כמות חיבורים מוגבלת, גם כשמגדירים Connection: keep-alive, בקשה חדשה תישלח רק כשהתשובה של הקודמת מסתיימת להגיע.

בקיצור, העסק עובד לאט וצריך לעשות משהו.

זה מה שחשבו בגוגל, ולכן הקימו צוות מחקר שיצר את פרוטוקול SPDY (קוראים את זה: ספידי) שביצע מספר אופטימיזציות על שיטת התעבורה של HTTP. אחת מהן לדוגמא היא ה-Server push, שבה השרת יכול לשלוח לדפדפן קבצים לקאש, עוד לפני שהדפדפן מבקש ממנו. וכך כשהדפדפן יצטרך את הקבצים הללו, הם כבר ישכנו אצלו לבטח במטמון.



התקן של גוגל היה כל כך טוב, שהוא תפס מהר מאוד וכולם תמכו בו דפדפנים ושרתים כאחד, וכשהחליטו להגדיר את תקן HTTP/2 כולם המליצו לבסס אותו על SPDY. למה להגדיר את HTTP/2 ולא להמשיך עם SPDY? כיוון שתקן צריך להיות מוגדר ע"י גוף עצמאי ולא חברה מסחרית שהיא צד במשחק. ויאמר לשבחה של גוגל שהיא כבר הודיעה שבתחילת 2016 תבטל את התמיכה ב-SPDY בכרום, כדי להתיישר לסטנדרט.

וכך הגענו ל-HTTP2.

קדימה ישראל, תפניק אותנו ב-HTTP/2

בשמחה. מה שחשוב זה להבין את העקרונות והמטרות של HTTP/2 ואז נבין את כל מה שנעשה שם. העקרונות שעמדו בעת התכנון היו:

- להשאיר את מבנה הפרוטוקול של HTTP/1.1
- לשפר את הביצועים, כך שדף שעולה בדפדפן, יוכל להיטען מהר יותר
- ללמוד מטעויות העבר, בעיקר בתחום ה-Security

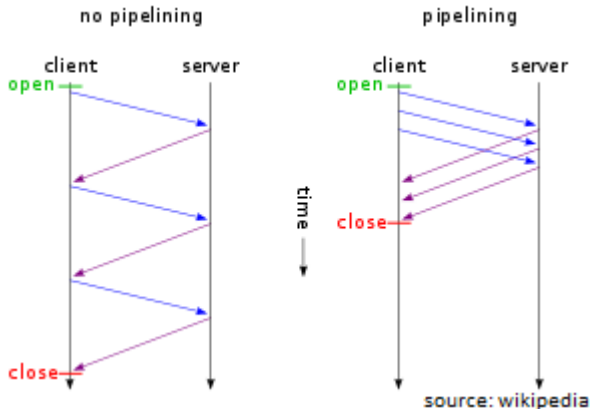
כדי לאפשר זאת, השתמשו במספר טכניקות, חלקם ברמה הטקסטואלית של הפרוטוקול וחלקם ברמת ה-Network.

להשאיר את מבנה הפרוטוקול של HTTP/1.1

ברמת האפליקציה, כל העיבוד של המידע, מבנה הבקשות, נותר זהה. כל מה שאנחנו מכירים על הכותרים (Headers), עוגיות (Cookies), מתודות (Get/Post ודומיו). גם הדחיסה הקלה שנראה בהמשך היא רק ברמת התעבורה, עבור הקוד בשרת האפליקציה הכל שקוף ונותר זהה.

Piplining over single TCP connection

בחיבור סינכרוני קלאסי, כנשלח בקשה (Request) - נקבל תשובה (Response). כשנשלח 2 בקשות, נקבל 2 תשובות. איך נדע איזו תשובה שייכת לאיזו בקשה? המודל הטבעי הוא FIFO. התשובה הראשונה שייכת לבקשה הראשונה. והתשובה השניה לבקשה השניה. מה שאומר שגם אם התשובה של הבקשה השניה אמורה הייתה להגיע מהר יותר, היא תחכה עד לתשובה של הבקשה הראשונה. ההמתנה מכונה HOL Blocking.



כדי להימנע מ"היתקעות" שכזו, צריך לעבור לתקשורת א-סינכרונית. הצורה הטבעית לעשות זאת היא להצמיד לכל בקשה איזשהו ID רץ. והשרת, כשהוא מחזיר תשובה מצמיד את ה-ID של הבקשה לתשובה. וכך גם אם נקבל את התשובה של הבקשה השניה קודם, נדע שהיא שייכת לבקשה השניה, לפי ה-ID. ב-HTTP זה מכונה http piplining. מדובר בהבדלי ביצועים משמעותיים מאוד.

המדהים זה שניתן לממש piplining גם על HTTP/1.1. ויותר מכך - מרבית הדפדפנים והשרתים תומכים בזה. אממה, כיוון שיש שרתים ש"התערבבו" מ-piplining, הדפדפנים מגיעים עם הפיצ'ר כשהוא Disabled.

בכל מקרה, ברגע שהחלטנו לממש piplining, אין עוד צורך בכמה חיבורים מול אותו שרת, אמור מעתה - HTTP/2 מבצע Single TCP connection מול השרת ומתקשר מעליו באמצעות ריבוב (Multiplexing). מבחינת HTTP/2, מסתכלים על הבקשה כמכילה 2 פריימים, פריים של ה-Headers ופריים של ה-Content. כך בבקשה וכך בתשובה, הפריימים רצים על החיבור בצורה שוטפת. המשמעות של הריבוב, היא שמספר פריימים יכולים לעבור (כמעט) באותו זמן והם לא מתבלבלים או מפריעים אחד לשני.

HPACK - Header compression

כדי לייעל את התעבורה, היינו מצפים שהדיפולט יהיה איזשהו כיווץ לתוכן הפריימים. הן על התוכן (מה שהיה מקובל לבצע באמצעות gzip) והן על ה-Headers. אך היות ובשנים האחרונות נפרצו מספר פרוטוקולים (ביניהם SPDY) שביצעו דחיסה למידע, גם כאשר התעבורה הייתה מוצפנת (ע"ע Crime, Breach) הוחלט לא לבצע דחיסה אלא אינדוקס.

קיימות 2 טבלאות. טבלה אחת סטטית, לכל מיני חלקים בפרוטוקול שהם נפוצים (Method GET, Method POST, Status code 200, Status code 404 ועוד). וטבלה אחת דינמית, שניתנת להגדרה פר קונקשן (כזכור, כל התקשורת של חלון הדפדפן עם שרת מסויים היא בחיבור יחיד).

תקשורת בינארית

אמנם התוכן הטקסטואלי של HTTP לא עובר המרה לבינארי, אבל כל יתר המידע של HTTP2 (טבלאות hpack, priority של פריימים וכד') עובר המרה. וגם התוכן הטקסטואלי עובר אנקפסולציה לתוך פריימים שהם מעטפת בינארית.

הסיבות העיקריות לשימוש בפרוטוקול בינארי הן שהוא קצר יותר. שזה אחד מהיעדים של HTTP2. והסיבה והסיבה השניה היא שהוא קל יותר לניתוח. הוא מוגדר מאוד ואין בו כל מיני שטיקים שקיימים בפרוטוקול טקסטואלי (ירידת שורה זה \n או \r\n. רווח ניתן לייצוג גם עם רווח, גם עם פלוס וגם עם %20 ודומיהם).

TLS also on HTTP

זה דבר שלא נקבע ברמת הפרוטוקול, אבל קרה דה-פקטו. פיירפוקס וכרום תומכים ב-HTTP2 רק מעל TLS, כך שגם אם בשורת הכתובת כתוב HTTP:// עדיין התקשורת תהיה מוצפנת. אם שני הדפדפנים הללו דורשים תקשורת מוצפנת, ההצפנה נהפכת להכרחית De Facto, ואין לשרתים טעם לתמוך ב-HTTP2 ללא TLS.

שנים שאני מחכה שיגיע הרגע הזה, והנה הוא בא. סוף סוף כל האתרים יהיו בתווך מוצפן, ומתקפות MITM יצטרכו להיות הרבה יותר חכמות.

Server side push

זה פיצ'ר שהוא בהחלט שונה ומעיד על חשיבה מקורית. הרי אנחנו בשרת, יודעים איזה קבצים נדרשים עבור הדפדפן לטעון את הדף. אז למה שנוריד לדפדפן רשימה של Resources ואז נחכה לקבל ממנו בקשות? אפשר במקביל, יחד עם רשימת ה-Resources להזרים עוד קבצים/Responses מהשרת. הדפדפן יכניס את הקבצים לקאש (מטמון), וכשיצטרך יטען אותם משם. דוגמת קוד Server side ב-NodeJS ש"דוחפת" קבצים לקליינט, ניתן לראות כאן.

מבחינת אבטחה, מיד קופצת לנו המחשבה על העמסה של הקאש של הדפדפן. אך זו בעיה שקיימת בלי קשר לשאלה אם הטריגר להורדת הקבצים בא מהדפדפן בעקבות דף html שהוא קיבל מהשרת או שהשרת שלח מיד לדפדפן את כל הקבצים. בכל מקרה הדפדפן צריך לאכוף איזושהי הגבלה.

לקריאה נוספת בעניין הפרוטוקול

וודאי יש לכם בראש שאלות כמו למה הוא נקרא HTTP/2 ולא HTTP/2.0? האם חושבים כבר על HTTP/3? האם זה ייתמך בקליינטים שהם לא דפדפנים? וכל מיני נקודות נוספות, אם כן כדאי לכם לקרוא את ה-FAQ הקליל של http2.github.io.

איזה שרתי Web תומכים ב-HTTP/2?

במפתיע, מרבית השרתים בגרסאות עדכניות כבר תומכים. IIS, Apache, Nginx, Jetty ועוד. חלקם בצורה מלאה, חלקם ע"י הפעלה של מודול וחלקם ברמת Experiment (רשימה מלאה). כמו שניתן לצפות, נראה שרתי פרוקסי/CDN-ים קדמיים (ע"ע אקאמאי) שתומכים HTTP2 ומאחורה שרתים שלאו דווקא תומכים.

תכל'ס, ישראל, בוא תראה לי איך זה נראה בפועל

כאנשי אבטחה וכמפתחים שמתעניינים באבטחה, מעבר לידע התיאורטי מעניין אותנו לנטר בפועל את התעבורה של HTTP2. יכול להיות שהדפדפן פונה ל-Endpoints אחרים באתר שלא הכרנו. אולי הם מכילים בעיות אבטחה שלא קיימות ב-Endpoints הישנים? אולי הוא בכלל שולח מידע לדומיין אחר?

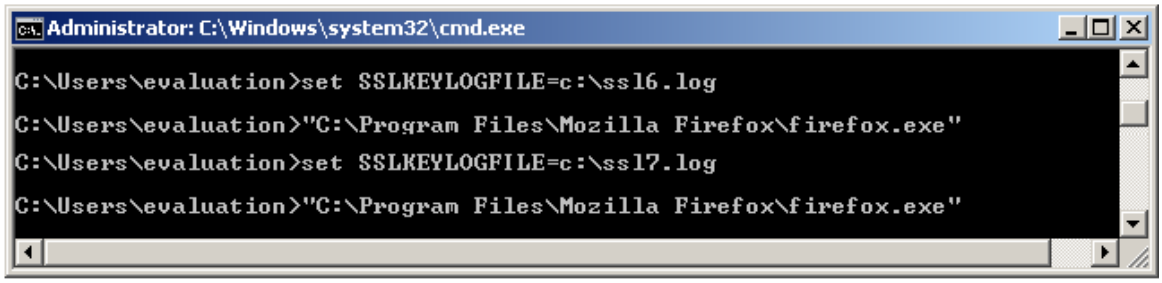
לכן מפתיע מאוד שכיום, כבר מספר חודשים אחרי שהפרוטוקול קיים בשטח, אתרים כמו גוגל ואקאמי משתמשים בו כבר (גם מתוך אתרים אחרים החיבור הוא ב-HTTP2). עדיין, אם תשימו Burp, Fiddler או כל כלי נפוץ אחר בין הדפדפן לשרת בניסיון לתפוס את התעבורה, זה לא יעבוד. הכלים הללו נכון לשעת כתיבת שורות אלו לא תומכים בפרוטוקול HTTP/2 (בפורום של Burp, למשל, נכתב ש[לא ידוע מתי הוא יתמוך]), וההתנהגות של האתר תהיה כמו בדפדפן שלא תומך HTTP2.

אז איך בכל זאת ניתן לראות את התעבורה? התשובה המפתיעה היא Wireshark. כאן אתם אמורים לקפוץ "הי, אבל אמרת שב-HTTP2 התעבורה תמיד מוצפנת!". אכן, היא מוצפנת, אבל פיצ'ר שמרבית האנשים לא מכירים מאפשר לפענח גם תעבורה מוצפנת דרך Wireshark. הביטו וראו.

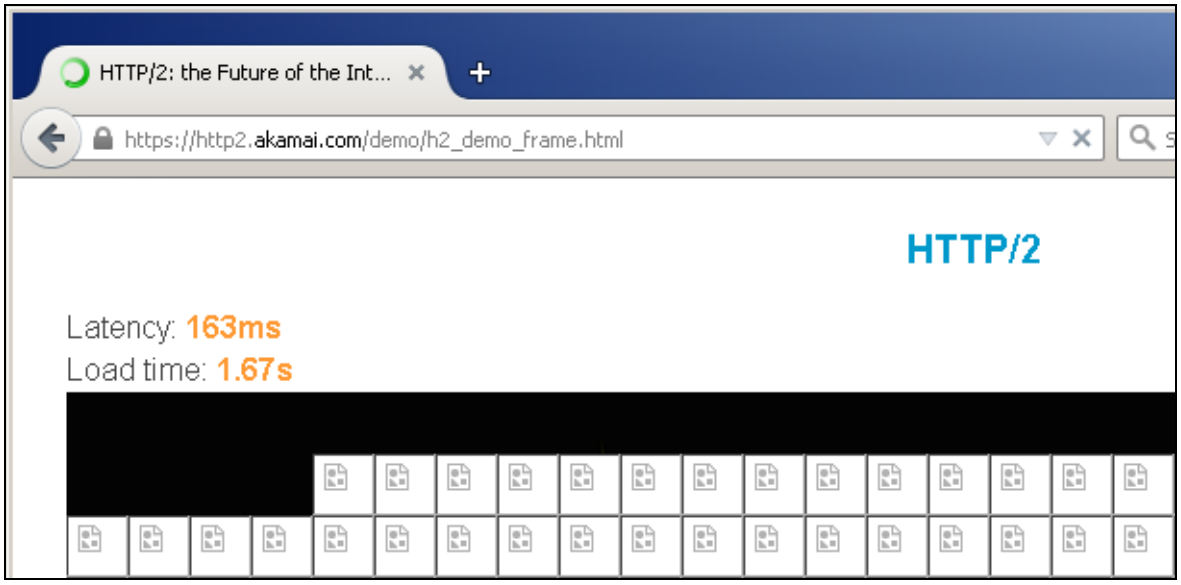
הערה: מכאן עד לסוף המאמר יהיו הרבה תצלומי-מסך, כדי שלא תבזבזו את השעות שאני השקעתי בחיפוש למה דברים לא עובדים לי...

דבר ראשון, אנחנו רוצים גרסת Wireshark שיודעת לנתח תעבורה של HTTP2, אז תורידו Wireshark בגרסת Development release. העדכני כיום (וממנו התצלומים) הוא 1.99.8. הוא נראה מעט שונה מהגרסה הרגילה. הורדנו? נפעיל אותו ונגיד לו להתחיל להאזין.

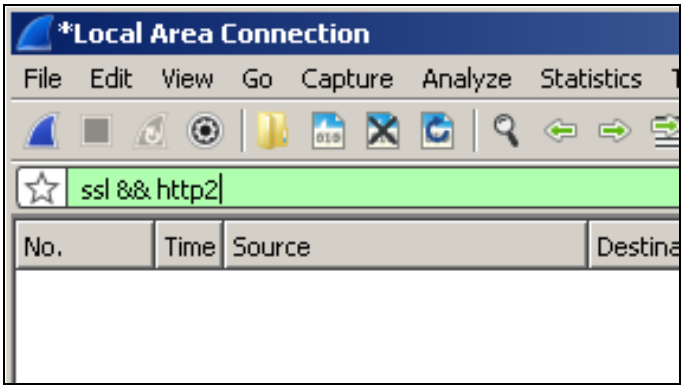
השלב הבא, יהיה להגיד לדפדפן להכניס לקובץ לוג מיוחד, את כל המידע הסודי שהוא מייצר בשלב יצירת חיבור מוצפן מול השרת, כדי ש-Wireshark יידע לקרוא אותו (לינק לפירוט טכני). נעשה זאת ע"י הגדרה של משתנה הסביבה SSLKEYLOGFILE לכתובת הקובץ (אין צורך ליצור את הקובץ מראש), ואז נפעיל את פיירפוקס/כרום.



בדפדפן שנפתח נגלוש לדף שמתקשר ב-HTTP2 (לינק):

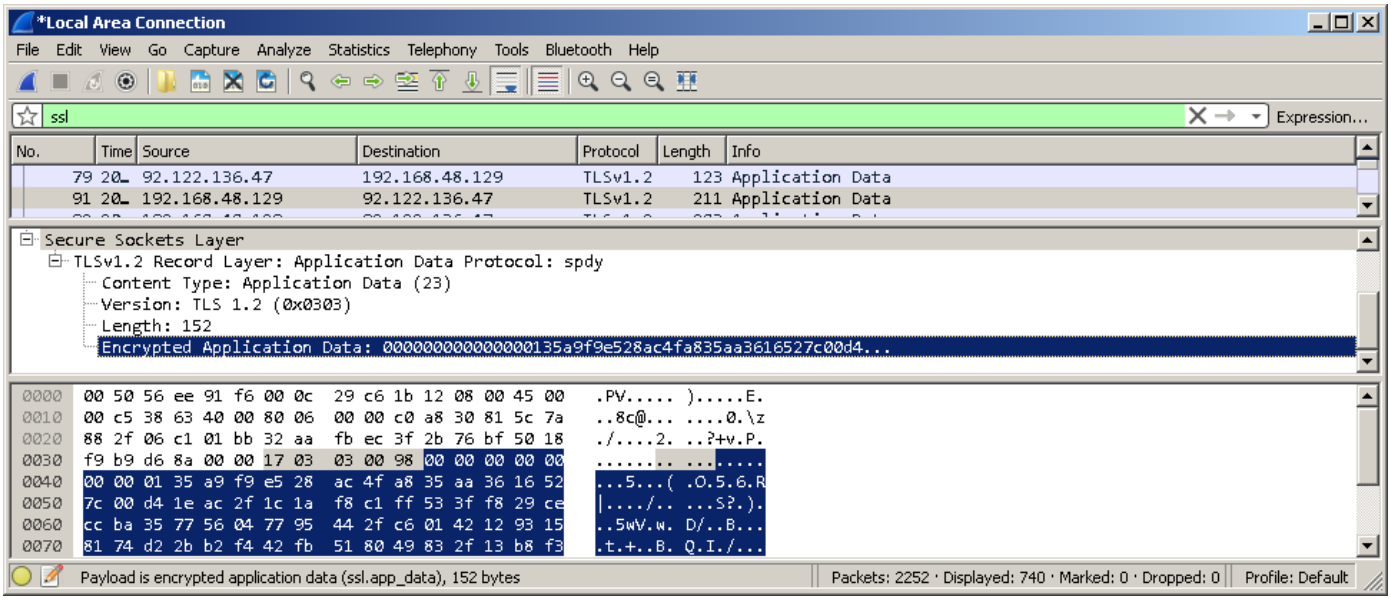


ב-Wireshark אנחנו אמורים לראות כעת תעבורה הולכת וגדלה. רק כדי "להוכיח" שהתעבורה של HTTP2 מוצפנת, נריץ פילטר על ssl && http2:



הסיבה שהתוצאה היא חלון ריק, היא שברגע שהתקשורת מוצפנת, ל-Wireshark אין אפשרות לדעת איזה פרוטוקול יש מעל ה-SSL.

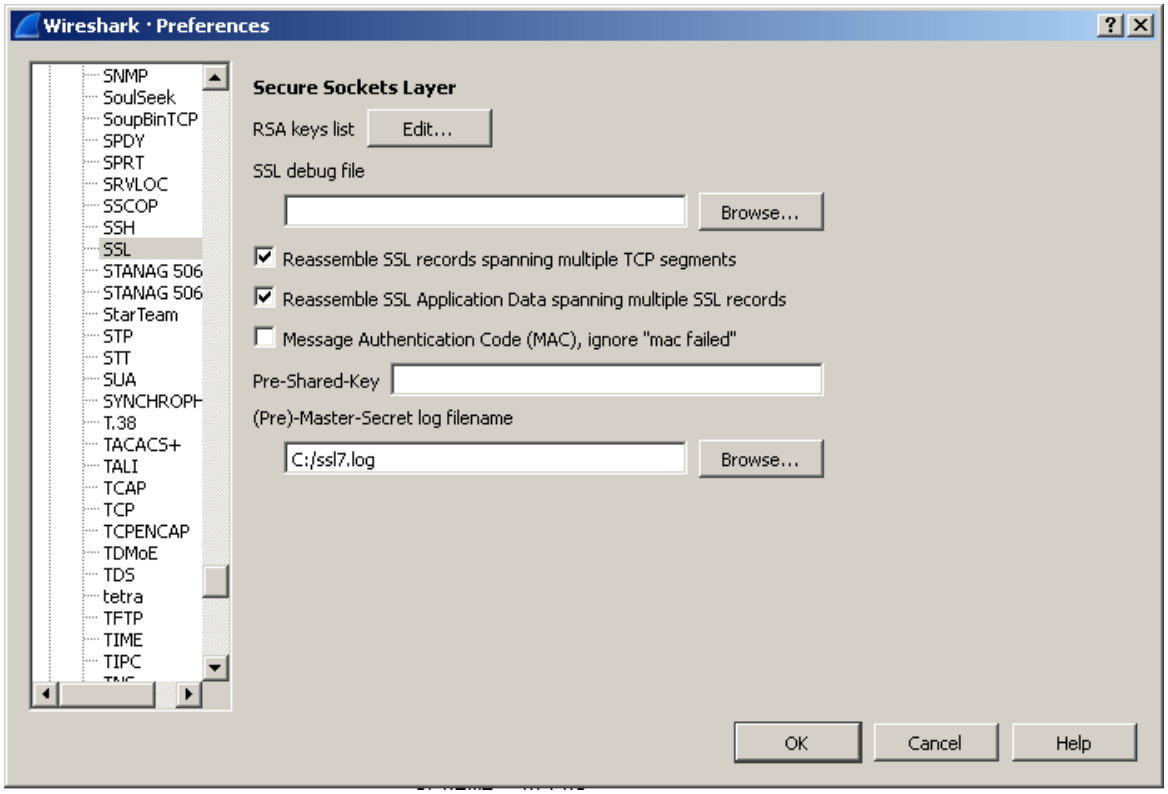
אפשר גם להריץ סינון על SSL. ללכת לאיזו שורה שה-Info שלה הוא Application Data, ולראות שהתוכן שלו הוא Encrypted application data:



הסיבה שאני מפרט את כל זה, היא שלפעמים בטסטים משהו לא עובד, ואתם לא יודעים אם הבעיה בפענוח, או שבכלל לא תפסתם טראפיק נכון, ואולי פיענחתם ואתם לא מבינים את זה... אז ככה תדעו להשוות באיזה שלב אתם נמצאים.

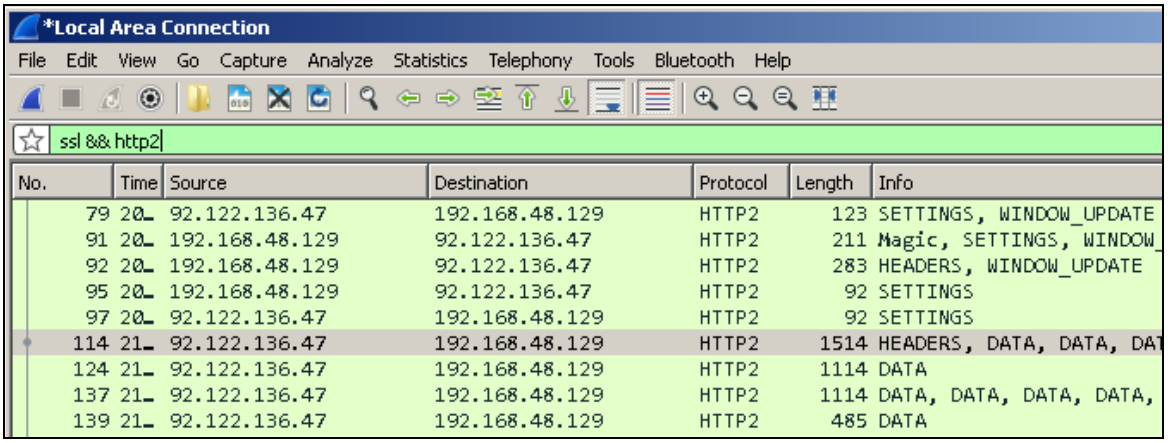
כעת ניכנס לתפריט Edit, ושם נבחר את הפריט התחתון ביותר, Preferences. בחלון שנפתח, במלבן השמאלי נפתח את Protocols, ובו נבחר את SSL.

כעת בצד ימין נגדיר את ה-Pre-Master-Secret log filename לקובץ שהגדרנו במשתנה SSLKEYLOGFILE:



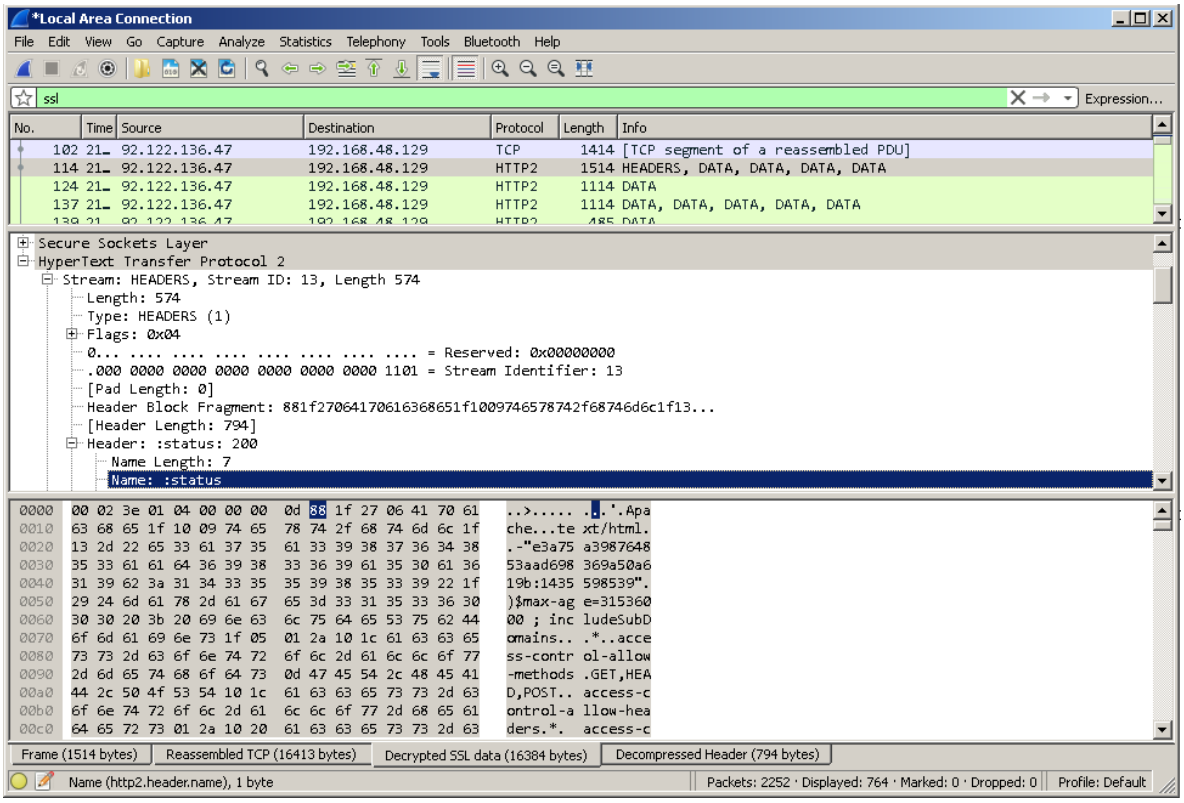
נלחץ OK, וזהו. כעת הכל אמור להיות מפוענח.

נריץ שוב את הפילטר של קודם ssl && http2 והפעם נקבל תוצאות:



כמו שאפשר לראות, בעמודת ה-Info מעבר ל-Headers ו-Data, יש עוד דברים שעוברים בתחילת החיבור. זוכרים את ה-Hpack שמאפשר להעביר טבלה דינמית? אז ככה יש כמה דברים שעוברים, ואז מתחיל לעבור המידע עצמו ב-2 סוגי Frames. אחד של Headers ואחד של ה-Data. בדיוק כמו שקראתם מקודם.

כעת נבחר שורה שמכילה Headers ונעיף בתוכה מבט:



הדבר הראשון שכדאי לראות זה שלמטה ממש מעל שורת ה-Status bar, יש טאבים תחתיים. הטאבים הללו לא היו לפני הפענוח של התעבורה. תחזרו שני דפים אחורנית לתצלום של Wireshark ותראו שאין שם את הטאבים.

שמתי את הפוקוס על הטאב Decrypted SSL data, ניתן לראות במלבן האמצעי שמדגיש בכחול את הטקסט: Name: :status וכפי שאפשר לראות 2 שורות מעליו מדובר בקוד 200. בכמה בתים לדעתכם נראה את זה במלבן התחתון? כפי שניתן לראות זה מיוצג בבית אחד. זו הדגמה חיה ל-Hpack שהזכרנו. 4 בתים אחריו ניתן לראות את ההידר Apache. איפה שם ההידר (Server)? גם הוא מועבר בבית יחיד. בדיקות מראות שהדחיסה הזו בהינתן מספר בקשות, חזקה הרבה יותר מ-Gzip!

נחזור למלבן האמצעי, שמתם לב לשורה המסומנת? Name: :status. מה זה הנקודתיים לפני המילה סטטוס? ובכן, ב-HTTP2 כשרואים את זה, סימן שהטקסט לא הגיעה כך אלא הגיעה באינדוקס Hpack

ומה שאנחנו רואים הוא הפענוח שלו. אם נרצה לראות את כל הבקשה במלבן התחתון מפוענחת בצורה חלקה, נלך לטאב Decompressed Header ושם נראה את הבקשה כמו בקשת HTTP רגילה.

מה שחסר שם זה הצלבה מלאה לעומת המלבן האמצעי. כשתשחקו עם זה טיפה תבינו וגם תלמדו עוד כמה דברים שאין טעם כעת להאריך בהם (לדוג' בפועל נעשה שימוש בפסאדו-הידר authority במקום בהידר המוכר host).

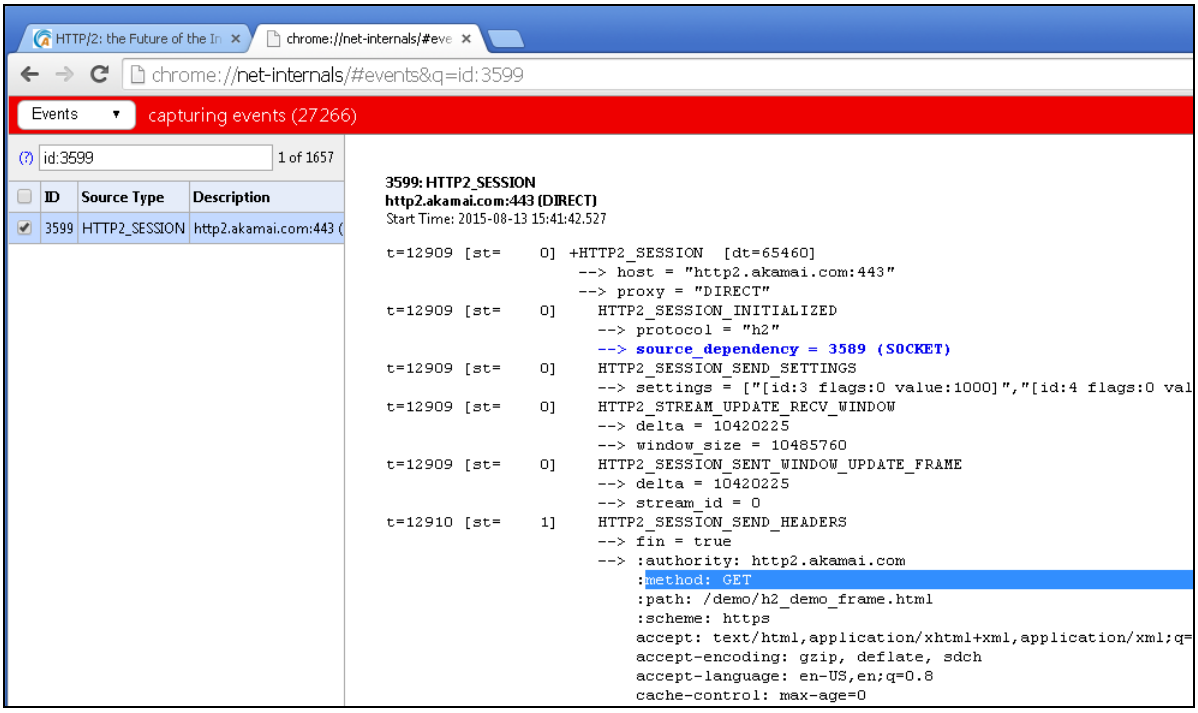
עד כאן בעניין Wireshark. אני רק אגיד ש-Wireshark יודע לפענח את המידע ה-SSLי ל-HTTP2 רק אם הוא היה בהאזנה (capture) מתחילת החיבור. כך שאם ביצעתם recapture או חלילה פתחתם את wireshark אחרי שגלשתם בדפדפן לכתובת היעד והדפדפן כבר התחבר פעם אחת לשרת. Wireshark לא יידע לפענח את התעבורה.

הצורה הקצרה לפתור את זה, לבצע capture ב-wireshark ואז לסגור ולפתוח מחדש את הדפדפן. אם רוצים לבצע "full restart", צריך לסגור את הדפדפן ו-Wireshark. לחזור לחלון CMD. להפנות את משתנה הסביבה SSLKEYLOGFILE לקובץ אחר (לא למחוק את הקובץ הקודם אחרת wireshark יקרוס כשהוא יתחיל להאזין, כי הוא עדיין מפנה לקובץ הישן), להפעיל שוב את Wireshark ולבצע Capture, להפעיל את הדפדפן מתוך ה-CMD שוב. וזהו.. לכן בתמונה הראשונה של חלון ה-CMD (3-4 דפים אחורנית) ניתן לראות שפעם הפניתי לקובץ ssl6.log ופעם לקובץ ssl7.log.

כלי נוסף שמאפשר לצפות בתעבורה, הוא לא אחר מאשר הדפדפן כרום בכבודו ובעצמו. מגיע איתו tool מובנה בשם net-internals שיושב בכתובת chrome://net-internals. אגב טיפ בעניין כרום, אם אתם לא זוכרים את הכתובות הפנימיות שלו, תמיד כשתתחילו לכתוב chrome:// אחת האופציות היא chrome-urls ששם יש את כל הכתובות.

נחזור לענייננו, גולשים ל-chrome://net-internals ולמעלה בצד שמאל בוחרים http2. כעת כשתגלשו מטאבים אחרים בכרום לאתר שמשתמש ב-http2 תראו "ססנים". בתמונה ניתן לראות שבחרתי ססן ונפתח חלון בצד ימין שמראה את התקשורת של הססן.

בתחילה יש את החלקים ה-"Network"ים של HTTP2, למטה ניתן כבר לראות הידרים של בקשה שנשלחים לשרת. שימו לב ל-method: אתם כבר יודעים למה יש שם נקודתיים...



לגבי עריכה של הטראפיק, לצערי כרגע אין כלי שמאפשר את זה, אז תצטרכו לבצע קומבינות (תוסף לדפדפן שמבצע hooks לפונקציות JS שמוציאות את הבקשות, תוסף ל-Wireshark, או כל דבר שאתם חושבים עליו).

אפילוג

ההרצאה הסתיימה, סשן קצר של שאלות ותשובות, ותוך כדי שהמשתתפים מגלגלים במוחם רעיונות לתקיפה של הפרוטוקול, טכניקות הגנה וכל מיני דברים שהאקרים חושבים עליהם, המרצה הבא נעמד ומחבר את המחשב שלו למקרן. טק-טוק מכיל בדרך כלל 2-4 סשנים. בהמשך גם הוצגה מתקפה שמאפשרת Code execution בשרתים. כאמור, שעה של נחת.

על הכותב ומשפחת אפסק

ישראל חורז'בסקי, CTO באפסק. בין השאר גם מרצה ויועץ אבטחת מידע בתחום Application Security.

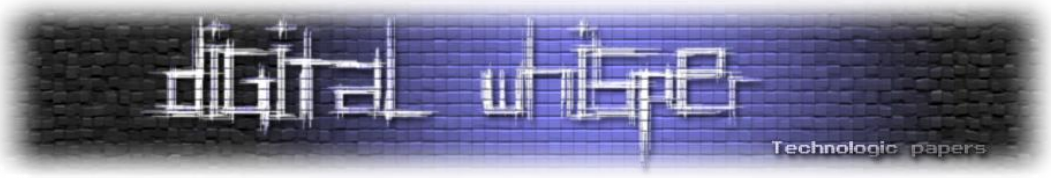
החדשות הטובות הן - אפסק מגייסת עובדים! הן Juniors והן Experts. יש 2 סיבות למה לעבוד באפסק: 1. זו חברה שנמצאת בטופ של הטכנולוגיה ושל המתקפות האפליקטיביות, ומשקיעה כל העת כדי להישאר שם. Period! 2. היחס לעובדים הוא אישי, עושים הכל כדי לבוא לקראתם, אנחנו משקיעים בעובדים, נותנים ואפי' דורשים כל העת להתקדם.

אז אם אתה רוצה להתקדם - מקומך איתנו, שלח קו"ח ל-israel@appsec-labs.com.

שלכם

ישראל חורז'בסקי

סמנכ"ל טכנולוגיות (CTO), AppSec Labs



דברי סיכום

בזאת אנחנו סוגרים את הגליון ה-64 של Digital Whisper, אנו מאוד מקווים כי נהנתם מהגליון והכי חשוב- למדתם ממנו. כמו בגליונות הקודמים, גם הפעם הושקעו הרבה מחשבה, יצירתיות, עבודה קשה ושעות שינה אבודות כדי להביא לכם את הגליון.

אנחנו מחפשים כתבים, מאיירים, עורכים ואנשים המעוניינים לעזור ולתרום לגליונות הבאים. אם אתם רוצים לעזור לנו ולהשתתף במגזין Digital Whisper - צרו קשר!

ניתן לשלוח כתבות וכל פניה אחרת דרך עמוד "צור קשר" באתר שלנו, או לשלוח אותן לדואר האלקטרוני שלנו, בכתובת editor@digitalwhisper.co.il.

על מנת לקרוא גליונות נוספים, ליצור עימנו קשר ולהצטרף לקהילה שלנו, אנא בקרו באתר המגזין:

www.DigitalWhisper.co.il

"Talkin' bout a revolution sounds like a whisper"

הגליון הבא ייצא ביום האחרון של חודש ספטמבר 2015.

אפיק קסטיאל,

ניר אדר,

31.08.2015