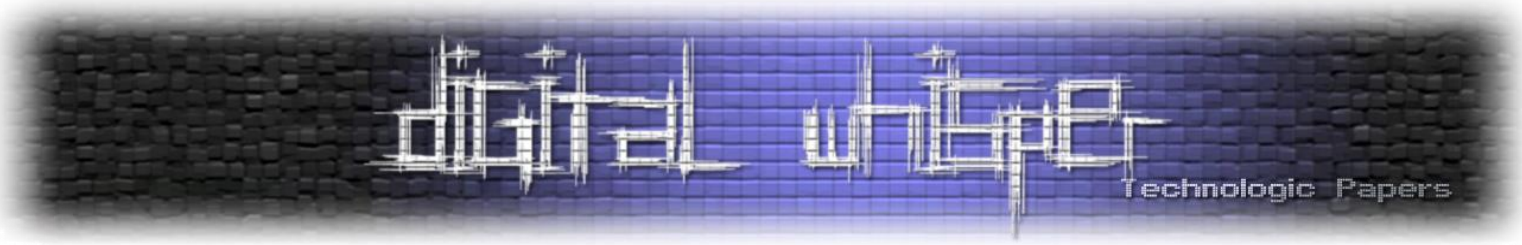


Digital Whisper

גליון 76, אוקטובר 2016

מערכת המגזין:

מייסדים:	אפיק קסטיאל, ניר אדר
מוביל הפרויקט:	אפיק קסטיאל
עורכים:	אפיק קסטיאל, ניר אדר
כתבים:	יורי סלובודיאניוק, אביחי כהן, יובל סיני ורועי חי.

יש לראות בכל האמור במגזין Digital Whisper מידע כללי בלבד. כל פעולה שנעשית על פי המידע והפרטים האמורים במגזין Digital Whisper הינה על אחריות הקורא בלבד. בשום מקרה בעלי Digital Whisper ו/או הכותבים השונים אינם אחראים בשום צורה ואופן לתוצאות השימוש במידע המובא במגזין. עשיית שימוש במידע המובא במגזין הינה על אחריותו של הקורא בלבד.

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

דבר העורכים

ברוכים הבאים לגליון ה-76 של DigitalWhisper!

הפעם, אין לנו יותר מדי מה להגיד, ולכן ננצל את הבמה הזאת, ולאחל לכם, קוראים יקרים שלנו - שנה טובה:

שתיהיה לכולנו שנת טובה, שנה של הרבה מחקר מעניין, שנה מרובה ביצירתיות, שנה בטוחה (או פרוצה? תלוי באיזה צבע החולצה שלכם...), שנה מלאה בחולשות מגניבות (אבל לא במוצרים שלכם כמובן), שנה של הרבה קוד - ובעיקר קוד פתוח, שנה בלי המילה "סייבר", שנה של ביטים ושל הצלחות בכל פרוייקט שתקבלו, שנה של עשייה, לימוד וידע. או בקיצור -

שנה טובה!

וכמובן, לפני שנצלול לחלק הבאמת מעניין של המגזין, ברצוננו להודות לי מי שבזכותו אוסף הביטים הזה מונח לכם על שולחן העבודה. תודה רבה ל**יורי סלובודיאניוק**, תודה רבה ל**אביחי כהן**, תודה רבה ל**יובל סיני** ותודה רבה ל**רועי חי**!

קריאה מהנה!

ניר אדר ואפיק קסטיאל.

תוכן עניינים

דבר העורכים	2
תוכן עניינים	3
איך לא מומלץ לנהל את ה-Firewall שלך	4
Deception To Catch Them All	16
מבוא ל-Transportation Cyber Security	30
הזלגת זיכרון ב-Nexus 5x דרך USB	47

איך לא מומלץ לנהל את ה-Firewall שלך

מאת יורי סלובודיאניוק

הקדמה

במהלך 10 שנות עבודה יום-יומית עם ה-Firewall של CheckPoint ראיתי לא מעט תקלות שונות ומשונות, בגרסאות שונות של Firewall-ים ובטופולוגיות רשת שונות - אך המכנה המשותף היה שכמעט בכל המקרים מנהלי הרשת היו חוזרים על אותן הטעויות שוב ושוב. המאמר הבא מסכם את התקלות השכיחות ביותר שנגרמות ע"י מנהלי Firewall-ים ובא לעזור למנוע תקלות כאלה.

מחיקת אובייקט שנמצא בשימוש

השגיאה הנ"ל אופיינית במיוחד לאנשי System בסביבת Windows - כאלה שמאשרים כל פעולה שמוצגת כאזהרה (Warning) ולא כשגיאה. מכל הטעויות, זאת יכולה להיות **הקטלנית ביותר** לקריירה שלכם. CheckPoint מאפשרת למחוק אובייקט שנמצא בשימוש - אך נותנת אזהרה שגם מראה איפה בדיוק האובייקט בשימוש.

ההמלצה שלי - לא למחוק אובייקטים שנמצא בשימוש **לעולם** - אלא לעבור על כל מקומות שבהם האובייקט בשימוש ולהוציא אותו משם בהפעלת היגיון כמובן ורק אחרי זה למחוק אותו.

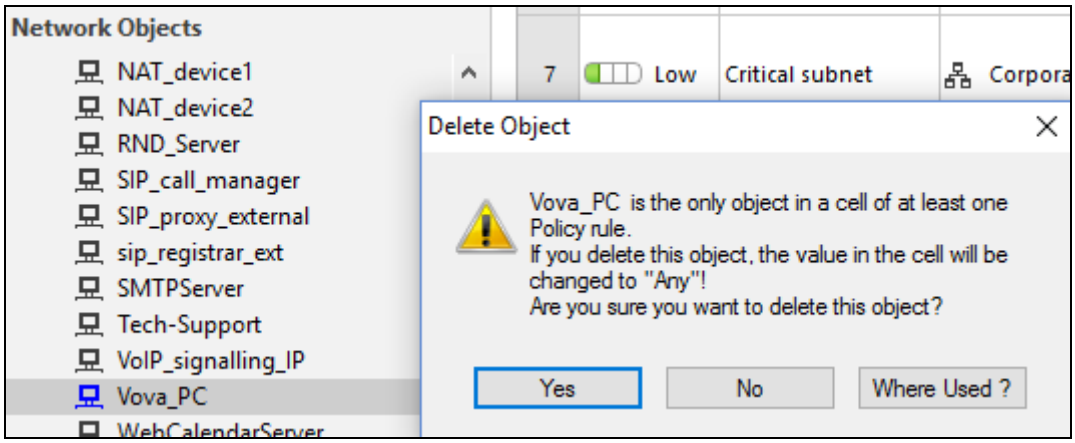
על מנת להמחיש את חשיבות העניין, אתן דוגמא מהחיים: יצא לי לטפל בלקוח שהתלונן על העובדה שכל העבודה של הארגון באינטרנט מאוד איטית ולא יציבה. אחרי כמה בדיקות נראה היה ש-Firewall שלו טוחן את הקו תמסורת של הארגון גם כאשר הרשת הפנימית מנותקת פיזית לגמרי.

אחרי חיטוטים בלוגים של ה-Firewall הסתבר שהוא נפרץ והפורצים הפכו אותו לשרת לינוקס לאחסון סרטים / תוכנות גנובות. מעבר לזה - הם השתמשו בו על מנת להריץ סורק / פורץ אוטומטי של שרתי SSH באינטרנט. מאחר וה-Firewall שומר ב-Management Log (מה שבעבר היה נקרא "Audit") שלו את כל הפעולות ניהול שבוצעו, לא היה קשה לאתר מה קרה...

אז איך הפורצים הגיעו ל-Firewall? אחד מחוקי האבטחה שהיו בו היה החוק הבא:

No.	Hits	Name	Source	Destination	VPN	Service	Action	Track
	Limit Access to Gateways Rule (Rule 1)							
1	Low		Vova_PC	Corporate-gw Management	Any Traffic	Any	accept	Log

כאשר Vova_PC (שם בדוי) - הוא מחשב פנימי ברשת מאחורי ה-Firewall. לאחר בירור, הסתבר שאחד ממנהלי רשת בארגון החליט "לעשות קצת ניקיון" והחליט לסדר את עץ האובייקטים. מסתבר שהוא מצא את האובייקט Vova_PC - אובייקט אשר היה שייך לאחד ממנהלי הרשת הקודמים וכבר לא עבד יותר בחברה. "יופי, צריך למחוק אותו" חשב המנהל וככה עשה. ה-Firewall-ים נתן לו אזהרה ברורה, אך הוא בחר להתעלם - ולחץ Yes. כך נראת האזהרה שהוא קיבל:



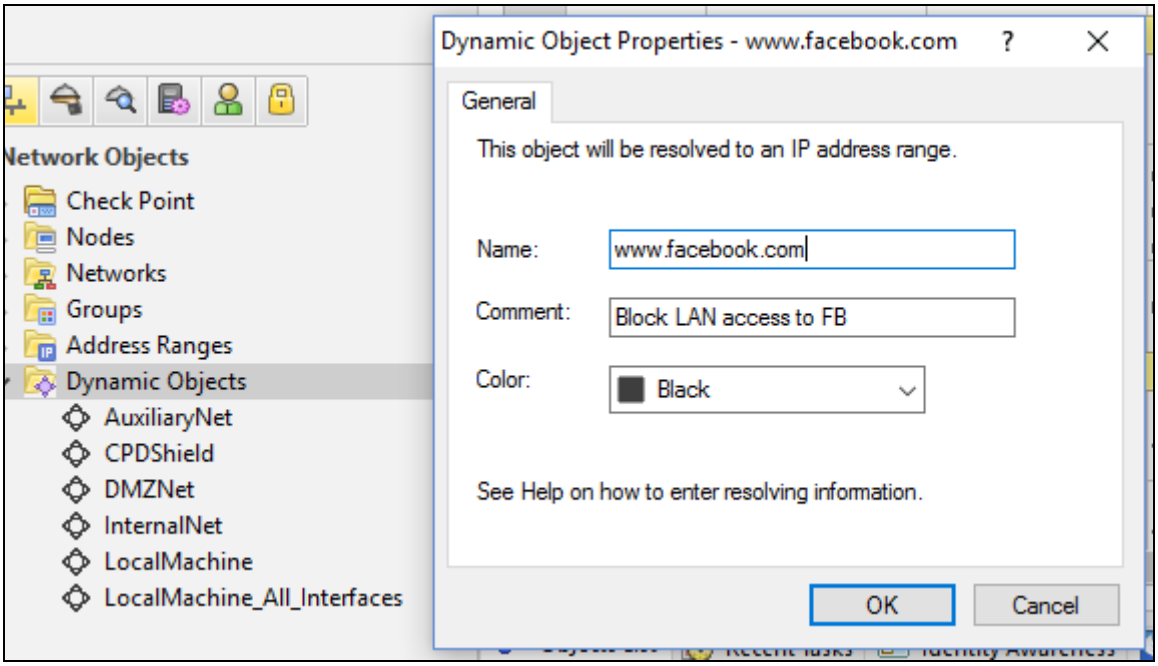
שהפך את כלל האבטחה הנ"ל ל:

No.	Hits	Name	Source	Destination	VPN	Service	Action	Track
	Limit Access to Gateways Rule (Rule 1)							
1	Low		Any	Corporate-gw Management	Any Traffic	Any	accept	Log

או במילים אחרות - פתח גישת ניהול ל-Firewall מכל מקום בעולם. במקרה הזה - גם לא עזר שמשתמש ה-SSH עם הרשאות root היה בשם admin עם סיסמה מעולה qwe123... בפחות משעה פרצו ל-Firewall קבוצת האקרים מרומניה, העלו סקריפטי bash אוטומטיים והמשיכו משם. למזל הארגון הפורצים לא הבינו לאן הם הגיעו ולא המשיכו הלאה לרשת הפנימית, אלא פשוט ניצלו את הרכיב כשרת לינוקס להפצת Warez וסורק SSH של שרתים באינטרנט.

שימוש ב-Dynamic Object לחסימת גישה לאתרי Web

זאת גם שגיאה הרסנית ל-Firewall. חוזרת על עצמה לרב באותה סיטואציה - מנהל Firewall נדרש לחסום גישה למשאב כלשהו באינטרנט ואין לו כתובת IP קבועה (למשל: לחסום גישה ל-facebook.com או ל-youtube.com). המכשול הוא שה-Firewall של CheckPoint יודע לעשות זאת רק עם רכיב ייעודי שנקרא URL/Application filtering ודורש רישיון המתאים לכך. הרישיון כמובן עולה כסף נוסף, ומה לעשות אם אין רישיון? ממשיכים לחפש בפיירוול עד שמוצאים ב-SamartDashboard, קטגוריה שנקראת Dynamic Objects ויש באובייקטים האלה אפשרות להגדיר משאב לפי שם ולא לפי כתובת IP! נראה שזה בדיוק מה שצריכים, ועוד בחינם... בלי לחשוד מגדירים אובייקט כזה לפי דוגמה:



ומשתמשים בו בכללי אבטחה, כמו למשל פה:

מקנפגים, עושים התקנת Security Policy ו-... במקרה הגרוע אנחנו מאבדים גישה לפיירוול והארגון מאבד גישה לאינטרנט. במקרה הטוב - החיבוריות נהיה איטית להחריד עד לרמה שפשוט לא ניתן לעבוד...

אז מה בעצם קרה? עבור כל אובייקט כזה בכללי אבטחה, עבור **כל פאקטה** שיכולה להתאים לכלל הזה ה-Firewall הולך לאינטרנט ומתשאל שרתי DNS מה כתובת IP הנוכחית של אובייקט (במקרה הזה של www.facebook.com). אם יש מספיק תעבורה דרך ה-Firewall (וזה כמעט תמיד המצב) אנחנו מגיעים

לבעיה קריטית של איטיות, בגלל הבדיקה הנ"ל, ה-Firewall נכנס לעומס של 100% CPU והוא ייתקע או יקרוס. העניין שאתחול של ה-Firewall לא יעזור הרבה כי הוא יעלה בחזרה עם אותו כלל משבית...

אז מה הפתרון? אם ה-Firewall עדיין מגיב: להסיר את החוק שמשתמש באובייקט הדינמי. אך אם זה לא אפשרי ונאלצים לנתק את הרשת הפנימית אל מנת להוריד מהעומס, אין ברירה אלא להתחבר ל-Firewall עם כבל קונסול, ולהסיר את **מדיניות אבטחה כולה** עם פקודה **fw unloadlocal**, חשוב שתשימו לב: הפעולה הנ"ך תפתח גישת ניהול מכל מקום ברשת ואולי אף מחוצה לה!, לכן חשוב לעשות זאת רק לאחר ניתוק ה-Firewall מהאינטרנט ורק על מנת להסיר את הכלל הבעיתי ולהתקין את המדיניות מחדש.

ההמלצה שלי פה היא: **אל תשתמשו באובייקט דינמי**, בכלל. בכל השנים שלי עם CheckPoint אולי פעם או פעמיים נאלצתי להשתמש בהם, אז בקיצור - תשכחו שהוא קיים.

אי-בדיקת מקום פנוי בדיסק הקשיח לפני ביצוע פעולת Debugging

העניין הזה הוא אחד החביבים עליי: אם הגעתם למסכנה שחייבים להיכנס למצב Debug כדי להבין / לפתור בעיה כלשהי - דבר ראשון זה לבדוק האם אין בעיה של מקום פנוי בכונן של ה-Firewall. חוסר מקום בדיסק יכול לגרום לאין סוף תופעות מוזרות והזויות, לדוגמא:

- אי-היכולת להתקין מדיניות אבטחה
- אי-היכולת לטעון לוגים ב-SmartViewLog, או גרוע יותר - ניתן לטעון לוגים אך הם ריקים.
- לא ניתן להתחבר לשרת SmartCenter.
- לא ניתן לעדכן חוקי Application Control / AV / IPS
- ועוד ועוד...

מה שבעייתי בכל התקלות האלה הוא שלמרות שהן נובעות מחוסר מקום בדיסק זה שאף פעם לא נקבל עליהם הודעת שגיאה מתאימה. תמיד יש איזו שגיאה פנימית של CheckPoint עם מספר כלשהו ומלל שרק מטעה... במקרה הזה צריך לזכור שבסופו של דבר Firewall זה שרת לינוקס לכל דבר. כדי לבצע את התפקיד שלו הוא מוריד למשל קבצים מאתר של CheckPoint, אם זה קובץ tar הוא יאלץ לפתוח אותו בתיקיה זמנית. השרת כל הזמן פותח / מוחק / יוצר / מצפין / שולח ל-SmartCenter קבצים (ולא לשכוח בלינוקס כל דבר הוא קובץ - כולל sockets וכו'). כל פעולה כזו דורשת מקום פנוי בדיסק.

אז המלצה שלי פה - תבדקו מקום פנוי בדיסק בכל בעיה הזויה. תבדקו במיוחד את המחיצה "/" שמערכת הפעלה מותקנת בה. לפעולה תקינה מניסיוני מומלץ שיהיה שם לפחות 1GB - 0.5 פנוי.

כדי לבדוק נכנסים ב-expert mode דרך ssh ומריצים df -h:

```
smartcenterr77> expert
Enter expert password:

Warning! All configuration should be done through clish
You are in expert mode now.

[Expert@smartcenterr77:0]#
[Expert@smartcenterr77:0]# df -h
Filesystem            Size  Used Avail Use% Mounted on
/dev/mapper/vg_splat-lv_current
                      7.8G  4.6G  2.9G  62% /
/dev/hda1             289M   24M  251M   9% /boot
tmpfs                 980M     0  980M   0% /dev/shm
/dev/mapper/vg_splat-lv_log
                      3.0G  524M  2.3G  19% /var/log
[Expert@smartcenterr77:0]#
```

שימוש בסיסמאות ניהול קלות לפריצה

דבר כל כך בסיסי שבטח תשאלו - סיסמאות קלות? ועוד של Firewall? לא הגיוני שמישהו יעשה זאת... עם זאת, תתפלאו לדעת עד כמה הרבה זה קורה בשטח... בדרך כלל זה מתחיל באינטגרטור המתקין Firewall חדש / משדרג אחד הקיים ושתמיד בלחץ של עוד 3 התקנות באותו יום. וכנראה שמפני שצריך להשתמש בסיסמת ניהול גם של SSH וגם של SmartDashboard כמה וכמה פעמים בהתקנה וקנפוג ראשוני - רבים מאותם מתקינים מקלים על עצמם ובוחרים סיסמאות כגון 1q2w3e \ 123456 \ qwe123 וכו' כדי לחסוך זמן בהקלדה ואומרים לעצמם - "אין בעיה, אחרי שנסיים, אשנה את כל הסיסמאות לקשות יותר", וכמובן שוכחים לעשות זאת...

ראיתי Firewall-ים שהתקינו אותם עוד בגרסה R55 עם סיסמא קלה, ו-10 שנים לאחר מכן - שדרגו אותן מבלי לשנות סיסמא כי פחדו לאבד גישה או לגעת במשהו שעובד שנים. אז המלצה שלי פה:

- לשנות בהתקנה (CheckPoint אגב, מציעה את האופציה הזאת בתפריט ההתקנה) את שם משתמש הניהול admin למשהו אחר - אל תפחדו, לא יקרה שום דבר.
- אם המשתמש כבר קיים וחוששים למחוק אותו - תשנו סיסמא שלו למשהו מסורבל וארוך, תשמרו את הסיסמא במקום שבו אתם שומרים את הסיסמאות ואל תשתמשו ב-admin אף פעם. פשוט תצרו משתמשים נוספים לכל מנהל Firewall - אם יש לכם כמה כאלה.

לשכוח לבטל האצה בפיירוול כשמבצעים Debugging

זאת שגיאה שכיחה שקרתה גם לי לא פעם ואפילו לתמיכה של CheckPoint. כשנמצאים תחת לחץ של תקלה לא פלא ששוכחים פרטים קטנים כאלה... בעבר זה לא היה כל כך חשוב, אך היום 99% מה-Firewall-ים (גם שרתי UTM וגם Open Servers) מגיעים עם יכולת האצת חומרה, מה שנקרא "SecureXL" בתיעוד של CheckPoint. התכונה זאת מורה לרכיב להרים ל-CPU רק את החבילה הראשונה של כל Connection, ואז אם נעשה "fw monitor" נראה רק את החבילה הראשונה של החיבור (במקרה של TCP SYN) ולא נראה את ההמשך.

דבר ראשון - בימינו, כאשר עושים בכניסה ל-Firewall הקלטה עם סניפר לטובת Debugging זה לבדוק אם מופעל SecureXL ואם כן - לבטל אותו זמנית ולהחזיר אחר כך. תשימו לב: יש שתי דרכים לבטל את האופציה הנ"ל. הראשונה היא דרך תפריט ה-cpconfig - אל תעשו זאת, מפני שהיא תבטל את ההאצה באופן קבוע (פעולה שגם דורשת אתחול של ה-Firewall).

האופציה השניה היא בעזרת פקודות ב-SSH:

- ראשית - לבדוק האם אופציה זו מופעלת בכלל:

```
fwaccel stat
```

- לאחר מכן, מבטלים:

```
fwaccel off
```

- אחרי סיום ה-Debugging, מפעילים בחזרה:

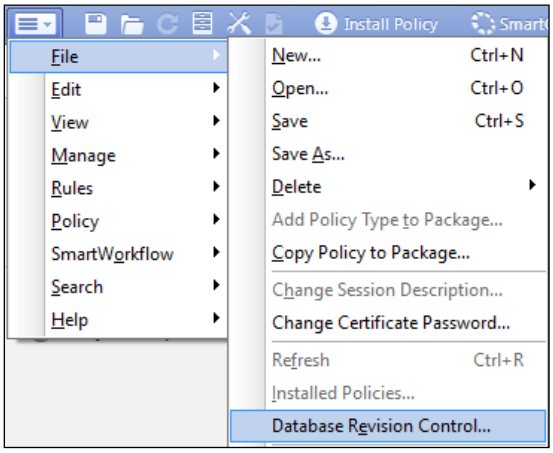
```
fwaccel on
```

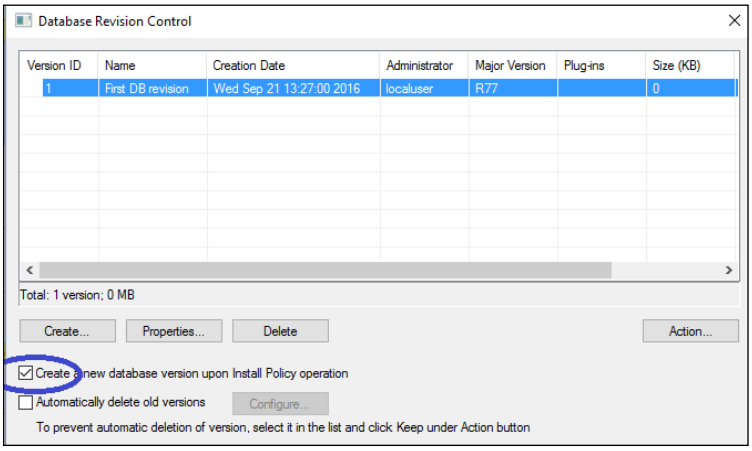
ביטול האצה כמובן יעביר את כל התקשורת ל-CPU ובך יעמיס את הפיירוול, אז תבדקו קודם שה-Firewall לא עמוס מדי לפני כן ויעמוד בהגדלת העומס או לחילופין - להפחית קודם לכן את העומס עליו.

אי-שימוש ב"ביטוח" נגד טעויות קינפוג - Database Revision Control

CheckPoint מאז ומתמיד הציעה אפשרות לשמור כגיבוי את הקונפיגורציה הנוכחית, השמירה כוללת את האובייקטים והכללים לפני התקנת מדיניות אבטחה. להפתעתי אולי רק ב-10%-15% מכלל ה-Firewall-ים שראיתי מפעילים את התכונה הזאת - וחבל. האופציה הזאת תוכל להציל את המצב אם נמחק אובייקט או חוק מורכב בטעות. עושים שינוי כלשהו שגרם לבעיות ברשת ולא בטוחים איזה שינוי בדיוק? ביצעתם מספר שינויים במקביל כאשר ה-Firewall מנוהל בו-זמנית ע"י כמה מנהלים ולא בטוחים מה בדיוק נהרס? תהליך שחזור גרסת מדיניות אבטחה דורש כמה קליקים בודדים...

הטענה היחידה הגיונית נגד גיבוי כזה היא שאם הוא מופעל להתבצע אוטומטית אז בכל התקנת מדיניות אבטחה, נוצר קובץ חדש שתופס מקום בדיסק של ה-Firewall (ב-SmartCenter ליתר דיוק), אבל גם פה אפשר לשים V על "Automatically delete old versions" וזה ימנע בזבוז מקום. אז המלצה שלי, גם לא למנהלים מתחילים - להפעיל Database Revision Control. עושים זאת כך:



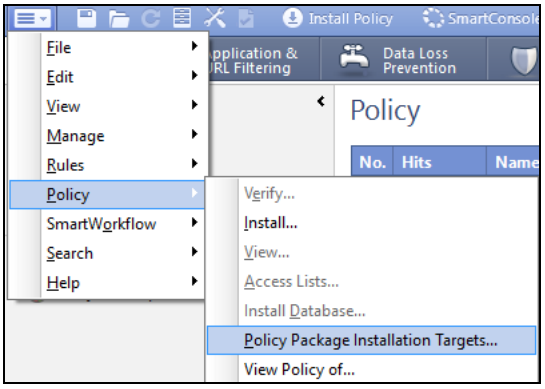


כשתרצו לשחזר קונפיגורציה - פשוט תבחרו את הגרסה מהתאריך הנדרש ותלחצו על כפתור: Action > Restore Version.

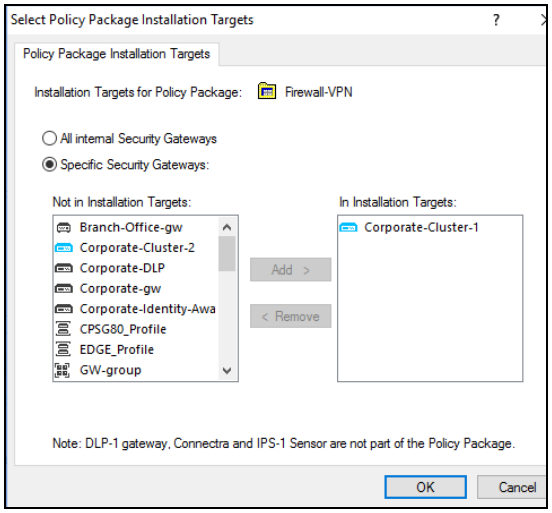
התקנת מדיניות אבטחה על Firewall הלא נכון

תקלה זו יכולה לקרות כאשר ה-SmartCenter מנהל כמה Firewall-ים במקביל או בשרת ניהול עם כמה חבילות מדיניות אבטחה עבור ה-Firewall-ים השונים שהוא מנהל. ה-SmartDashboard נפתח על מדיונות שהמשתמש הקודם סגר. לעיתים קרובות (במיוחד כשיש לחץ) קורה שפותחים SmartDashboard, מקנפגים כלל בלי לשים לב שעבדנו על מדיניות של Firewall אחר לגמרי...

CheckPoint לא מאמתת איזו מדיניות מתקינים לאיזה Firewall. ולכן, כאשר מתקינים מדיניות שמשתמשת באובייקטים וכללים לא רלוונטיים ל-Firewall, ברב רובם של המקרים הדבר יגרום להשבתה שלו ותפגע בכל התעבורה שעוברת דרכו. עם זאת - לא קשה להתאושש מתקלה שכזו - פשוט לבצע התקנה נוספת, אך הפעם עם המדיניות הנכונה. עם זאת, תמיד עדיף להמנע מבעיות מאשר לפתור אותן... CheckPoint מאפשרת לנו לקבוע מראש איזו מדיניות תותקן באיזה פיירוול. עושים את זה ככה:



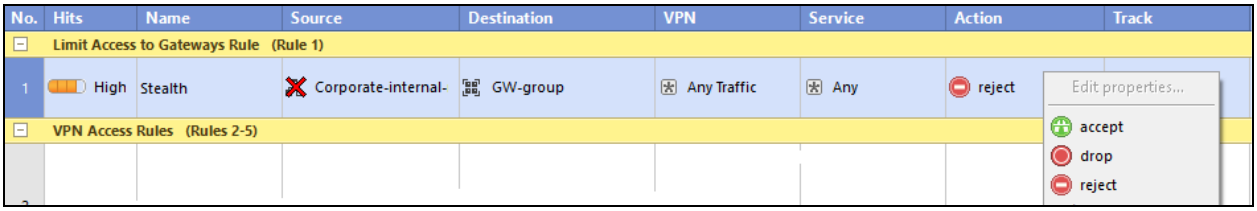
ולאחר מכן בוחרים לאיזה Firewall המדיניות הנוכחית הפתוחה ב-SmartDashboard תותקן:



כאן מדיניות שפתוחה כרגע וחלק מחבילה Firewall-VPN תותקן רק בפיירוול Corporate-Cluster-1 בעתיד בלי שתעשו בשביל זה משהו.

הפעלת כללי אבטחה עם פעולת Reject במקום Drop

קורה כשלא מבינים מהו הבדל ולכן קל להתבלבל. הדבר פשוט מאוד: Reject לא רק חוסם ניסיון תקשורת אלא גם שולח ליוזם התקשורת תגובה על כך (לדוגמה TCP RST), אופציה זו סתם מעמיסה על Firewall וגם נותנת אינדיקציה למישהו שמבצע סריקה מבחוץ שהוא אכן נחסם ע"י פיירוול.
היום אני לא מכיר שום סיבה להשתמש ב-Reject הזה...



אתחול ה-Firewall כולו כאשר צריכים לאתחל רק את ה-SmartCenter בלבד

לעיתים קרובות מנהלי רשת לא שמים לב לכך שרכיב ה-Firewall עצמו ורכיב ניהול ה-Firewall (ה-SmartCenter) הם שתי תוכנות / מערכות נפרדות, אפילו כשהן מותקנות על אותו שרת פיזי.

כשנתקלים בבעיית SmartCenter כלשהי שמחייבת אתחול (במחשבים אין כמו אתחול טוב ☺) - עושים אתחול לכל השרת, העניין אולי פותר את בעיית ה-SmartCenter אבל גם מאתחל Firewall ומשבית את כל התעבורה שעוברת דרכו... אין שום צורך בכך - תשתמשו בפקודות האלה כדי לאתחל את רכיב הניהול בלבד, מבלי לפגוע בתפקוד Firewall עצמו:

- סגירת ה-SmartCenter:

```
cpwd_admin stop -name FWM -path "$FWDIR/bin/fw" -command " fw kill fwm"
```

- הפעלתו מחדש:

```
cpwd_admin start -name FWM -path "$FWDIR/bin/fwm" -command "fwm"
```

אי-שימוש ב-NTP כמקור לשעון פיירוול

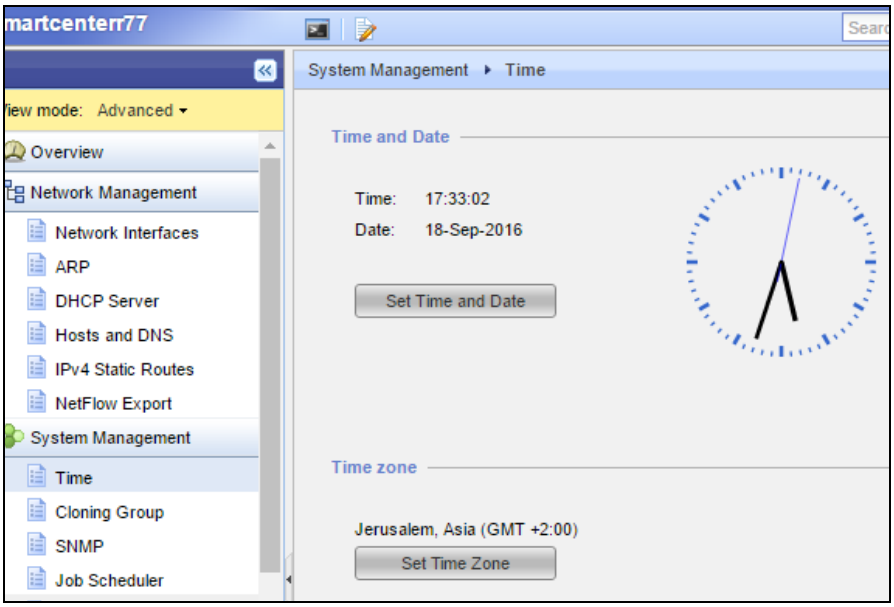
לא פעם הייתי חלק מהליך Debugging ארוך ומייגע שנבע מכך שהדברים ב-Log נראו לא הגיוניים, כל זה כדי שבסוף נבין ששעון ה-Firewall לא היה בכלל מכוון. ה-Firewall של CheckPoint יוצר לא מעט לוגים: לוגי אבטחה, לוגים של כל הרכיב הפנימיים שלו (Check Point daemons logs שסיומת שלהם elg.) והם מאוד עוזרים בעת פתרון בעיות. הלוגים שמגיעים ל-SmartCenter חתומים עם תאריך ושעה של מודול ה-Firewall שבו הם נוצרו, ואם השעון ה-Firewall לא מכוון זה פוגע באמינות הלוגים וגורם להם להטעות במקום לעזור.

מניסיון שלי - לא משנה איזה שרת, כולל השרתים היקרים והמתקדמים ביותר - השעון שלהם סוטה עם הזמן. ועם זה לא מספיק - הסטייה לא מתקיימת בצורה לינארית - מה שאומר שעם הזמן הסטייה גדלה בערך שאינו קבוע.

התוצאה מכך היא שאם אני מסתכל על לוגים של היום ורואה ששעון סוטה 10 דקות - אין לי דרך לדעת מה הייתה סטייה של שעון לפני חודש ועד כמה זמני הלוגים לא מדויקים... יש מקרים שבגלל הבעיה הזאת הלוגים פשוט לא שווים כלום.

איך מתקנים את זה? פשוט מאוד - מחברים את ה-Firewall לשרת NTP אמין (אפילו לכמה: יש תמיכה באחד ראשי ואחד משני).

אפשר לעשות את זה דרך Gaia:



או כמובן דרך ה-CLI:

```
smartcenterr77> set ntp server primary 13.13.13.1 version 2
smartcenterr77> set ntp server secondary 23.23.23.1 version 2
smartcenterr77> save config
```

אי-אימות גיבויים

העניין רלוונטי לא רק בעולם ה-Firewalling כמובן, אך עם ה-Firewall העניין קריטי במיוחד. CheckPoint מציע כמה דרכים לגבות את הקונפיגורציה של ה-Firewall: דרך Gaia, דרך CLI, לעשות זאת באופן יזום חד פעמי או באופן מתוזמן אוטומטית, הכי חשוב כמובן זה לגבות את ה-SmartCenter שמכיל את כל האובייקטים, הכללים, מסדי נתונים של ה-Firewall וכו', לגבות את המודול של ה-Firewall (ב- Distributed Installation) זה גם עניין מומלץ אך פחות קריטי מפני שהוא כולל רק את כתובות ה-IP של הממשקים.

במקרים רבים ה-SmartCenter מותקן על VmWare או תשתית וירטואליזציה מקבילה, ואז עניין הגיבוי הוא לא בעיה - פשוט לדאוג ל-Snapshots. אך, אם מבצעים את הגיבוי עם כלים של CheckPoint - אז חובה מדי פעם לנסות לשחזר Firewall מגיבוי שכזה. שוב, אני מדבר מניסיון - פנה אלינו לקוח ששרת ה-SmartCenter שלו לא עולה - שגיאה הקשורה לדיסק הקשיח. למזלו של הלקוח הוא הריץ באופן קבוע גיבוי אוטומטי מתוזמן - פעם בשבוע, אחרי סיום גיבוי ה-CheckPoint היה מעביר קובץ גיבוי לשרת ברשת דרך FTP. אגב מדובר במשרד ממשלתי מאוד גדול שלא היה חסר להם משאבים, וה-Firewall היה קריטי לעבודתו התקינה של ארגון.

הלקוח עם אינטגרטור שלו הביאו שרת חדש מאותו סוג, התקינו CheckPoint גולמי וניסו לשחזר את ה-SmartCenter מקובץ גיבוי. הם הריצו Upgrade Import וקיבלו שגיאה - הקובץ גיבוי אינו תקין, הריצו שחזור נוסף, הפעם עם קובץ גיבוי קצת ישן יותר - ושוב, אותה שגיאה. הם עברו על לפחות 20 קבצי גיבוי שהיו להם - אותו דבר, כל קבצי גיבוי יצאו פגומים! בסוף הם נאלצו להביא מישהו חיצוני ששחזר להם את הנתונים של ה-SmartCenter ישר מהדיסק הקשיח עצמו וככה ניצלו.

אז המלצה שלי - אם מריצים גיבוי בכלים של CheckPoint או עם סקריפטים משלכם, חייבים לאמת תקינותם. איך? פשוט מאוד - ע"י ביצוע שחזור.

חשוב שתבינו דבר אחד לגבי הליך הגיבוי: לא מדובר פה ב"קסם" של CheckPoint. בסופו של דבר, מה שהוא עושה זה להריץ מספר סקריפטים / תוכנות שאוספות קבצים, מאגדות אותם, מכווצת אותם לארכיון ובעזרת קליינט (למשל של FTP) מעלים לשרת שהוגדר. יש הרבה דברים שיכולים להסתבך בתהליך הזה, לדוגמא:

- לא היה מקום פנוי במחיצה המחזיקה את temp ותהלך ה-tar נכשל, מה שגרם לכך שחלק מהקבצים לא גובו.
- יכול להיות שרת ה-FTP בעיתי ופגע בשלמות הקבצים המועברים אליו.
- ועוד שלל סיבות נוספות.

סיכום

אם אתם מנהלי רשת, או חלק מצוות ה-IT ובמסגרת תפקידכם אתם מנהלים רכיב Firewall (לאו דווקא של חברת CheckPoint) אני בטוח שנתקלתם בלפחות חלק מהבעיות שהצגתי במהלך המאמר, או בבעיות דומות בעת תפעול ותחזוק ה-Firewall. חשוב מאוד להבין שמדובר בשרת לכל דבר. אם צריך אסכם בקצרה את הנקודות החשובות במאמר:

- אל תמחקו אובייקט אשר נמצא בשימוש באחד החוקים, ובכלליות - תקראו טוב טוב את השגיאות שאתם מקבלים.
- אל תשתמשו ב-Dynamic Object, ובכלליות - אל תשתמשו בפיצ'רים אם אתם לא בטוחים לחלוטין מה ההשפעה שלהם.
- חוסר במקום פנוי יכול להוביל לשלל בעיות הזויות, בכל פעם שמשהו נראה לא הגיוני - בדקו כמה מקום פנוי, ובכלליות - תדאגו תמיד שיהיה לכם לפחות 0.5 GB פנוי.
- תשתמשו בסיסמאות חזקות מאוד. אל תתעצלו. זה יכול להגמר באסון.
- בטלו את פונקציית ההאצה לפני ביצוע כל פעות Debugging הקשורה לרשת.
- בצעו גיבוי לקונפיגורציה של ה-Firewall באופן אוטומטי ופעם בכמה זמן - בדקו שאכן ניתן לבצע שחזור ממנה.
- הפעילו שרת NTP, יחסוך לכם כאב ראש לא קטן במידה ותכנסו ל-Debugg Session.

וזהו, תודה שקראתם את המאמר, אני מקווה מאוד שהפקתם תועלת ממנו. אשמח לתגובות / הערות / שאלות בקשר למאמר:

yuri@yurisk.info

יורי סלובודיאניוק,

FCNSP ,CCSE+ ,CCNP Security

LINKEDIN | BLOG

Deception to catch them all

מאת אביחי כהן

הקדמה

בעקבות הרווחיות אשר מתקפות סייבר מוצלחות מניבות, טריוויאלי יהיה לצפות שתחום ההגנה במימד זה ישגשג בצורה כזו או אחרת. בדרך כלל, בכל כמה ימים אנחנו שומעים על עוד חברה שנפרצה - בין אם בשביל גניבת מידע ובין אם מדובר בכל מניע אחר (כדוגמת Ransomware), המצב הנוכחי מביא לאחרונה חברות מכל הסוגים, למעין סוג של התעוררות בכל הנוגע לעולם שבין אם ירצו או לא - הם חלק ממנו, ובכך חשופים לאיומים מבחוץ. מה שמחייב אותן לבצע לחיפוש פתרונות מסוגים שונים כגון: Firewalls ,Content Filters, WaFs ,Honeypots וכו'.

ישנם הרבה שחקנים וותיקים בתחום הזה אשר מציעים פתרונות טובים, אך לאחרונה נכנסו שחקנים חדשים שלקחו את הרעיון של Honeypot מספר שלבים קדימה.

> מלכודת דבש (Honeypot) - "באופן כללי, מלכודת דבש מכילה נתונים אשר נראים כחלק מנתוני בסיס הנתונים או המערכת, ושלכאורה מכילים מידע בעל ערך לתוקפים, אך למעשה הם מבודדים ומנוטרים וחוסמים למעשה את התוקפים .ניתן להקביל זאת להצבת פיתיון משטרתי ולביצוע מעקב סמוי ,עד לתפיסת העבריין וענישתו." (ויקיפדיה).

חברות חדשות אלו הגיעו עם "מוטו" שונה לחלוטין ממה שהיה מקובל עד כה, מוטו שאומר בעצם שלא משנה כמה שכבות הגנה ארגון יטמיע ברשת שלו, פורץ מספיק מתוחכם יוכל לעקוף הגנות אלו ולחדור אל הרשת.

ולמען האמת? כולנו יודעים עד כמה זה נכון... במיוחד לאחר ההדלפה של "Shadow Brokers" שהכילה אקספלויטים שמנצלים 0-day ל-Firewalls מחברות כמו Fortinet, Cisco, Topsec וכו'. המוטו הנ"ל יושם למערכות שונות שהמטרה שלהן היא לא חסימת התוקף אלא שתילת מספיק מידע כוזב ברשת כגון: פרטי הזדהות, קבצים, לוגים וכו', שכאשר פורץ אכן יחדור לרשת הוא ישתמש במידע כוזב במקום מידע אמיתי וככה יחשוף את עצמו.

במאמר זה אפרט על פרויקט שכתבתי, אשר עושה שימוש באותן הטכניקות שחברות אלו משתמשות בהן על מנת לשתול מידע וניטור שלו. בנוסף, הטמעה של יכולות חדשות שהוספתי שלדעתי מוסיפות עוד רבדים נוספים המקשים על הפורץ לזהות בין מידע אמין לכוזב. (בעקבות כך שהפרויקט שלי נעשה בשביל

חברה מסוימת אני לא אוכל לפרסם את הקוד שלי אבל אשמח לעזור לכל מי שיבקש בבניית מערכת דומה או בכל שאלה אחרת ☺).

פסיכולוגיה של פורץ

על מנת באמת להבין את היתרונות של המערכת הזאת צריך קודם כל להבין את מהלך המחשבה של פורץ שכרגע השיג גישה לאחת התחנות ברשת הארגון.

המטרה של הפורץ היא להגיע למידע הרגיש (מסד נתונים, קבצים רגישים, וכו') ועל מנת להגיע למידע הרגיש עליו למפות את הרשת, לחפש מכונות מעניינות ולהתחיל להתקדם לכיוון המטרה, הוא ינסה להוציא מהתחנה פרטי הזדהות מול ה-AD (אולי Domain Admin) ופרטי הזדהות מול שירותי רשת שונים בתוך הארגון, הוא ינסה להפעיל sniffer על התחנה על מנת להאזין לתעבורת הרשת ואולי "ללכוד" פרטי הזדהות נוספים (מכאן הגיעה המחשבה על המודול הנוסף בפרויקט שלי - AIS) ועוד ועוד...

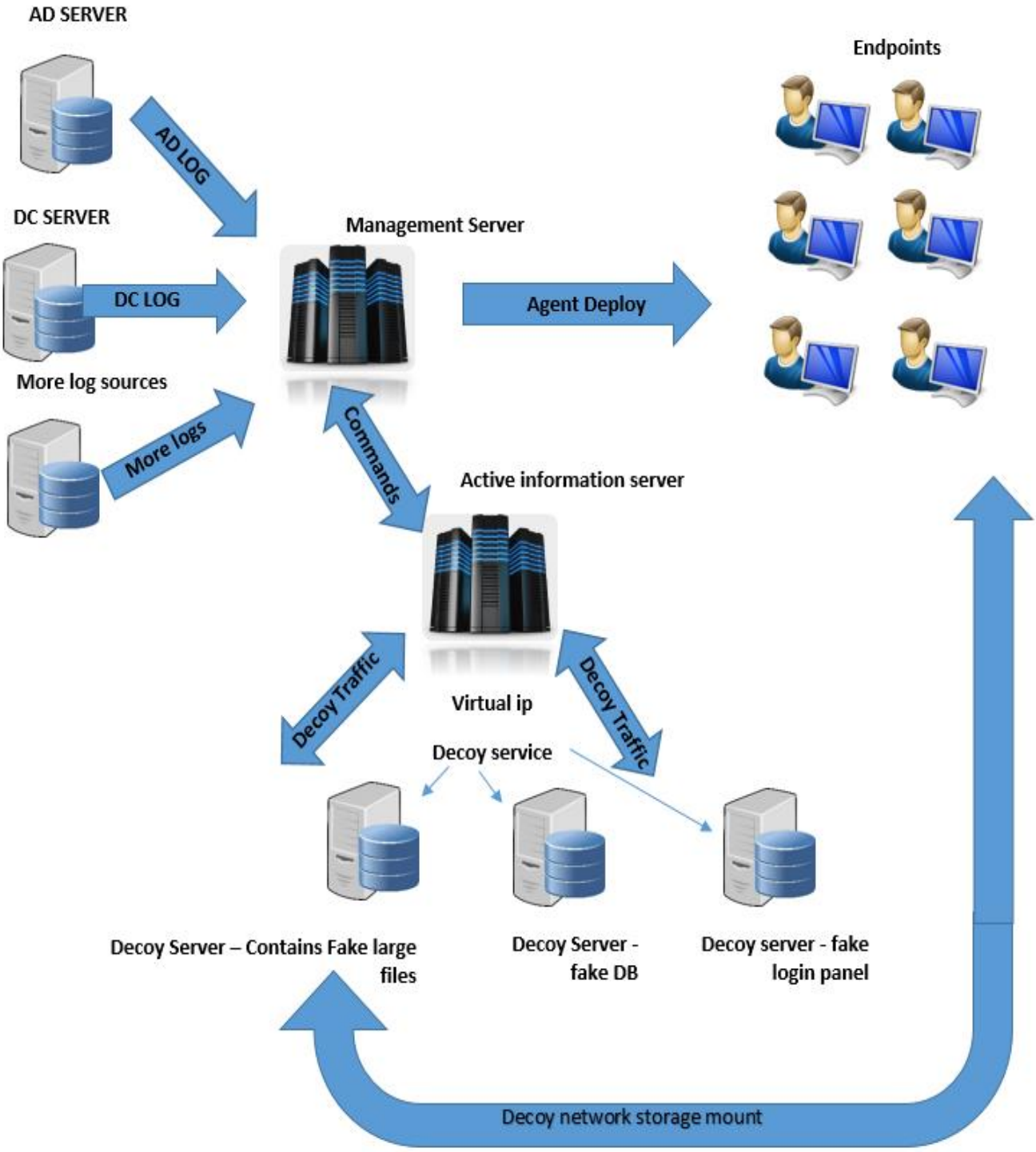
כמובן שלא שכחתי את כל ה-Ransomware למיניהן שהמטרה שלהן היא להצפין את הקבצים ולדרוש כופר בתמורה למפתח ההצפנה, וכאן נכנס ה-Decoy NFS שעליו ארחיב בהמשך.
אז אחרי הסבר קצר על מהלך המחשבה של פורץ ברשת הארגון נתקדם הלאה.

קצת מאחורי הקלעים

המערכת בנויה מ-3 מודולים עיקריים:

1. **Management Server** - שרת הניהול (הממשק משתמש נכתב ב-PHP וכל השאר ב-c) שתפקידו לספק את המידע (הכוזב) ל-Agent-ים ול-AIS (Active Information Server), לרכז ולנתח את הלוגים אשר מגיעים ממקורות שונים ברשת, וכמובן - ממשק משתמש נחמד שמציג למשתמש התראות אחרונות או שגיאות וכו' (ארחיב בהמשך).
2. **Agent** - (נכתב ב-c) קוד שירוץ באופן חד פעמי על התחנה אשר ישאב את המידע הכוזב משרת הניהול ויתחיל בשתילת המידע במיקומים שונים במערכת ההפעלה ובנוסף יצור mount ב-NFS לשרת Decoy שלנו, ולאחר מכן ימחק את עצמו.
3. **Active Information Server** - זהו מודול שהוספתי כרובד נוסף של מידע כוזב שכל מטרתו היא העברת מידע לא אמין ברשת ובכך יצירת "רעשים" בתעבורת הארגון (ארחיב בהמשך).

כך נראה תרשים של המערכת:



השתלת מידע

בעקבות האפשרויות האינסופיות של מידע שניתן להשתיל אני אתמקד באפשרויות המוצלחות ביותר:

פרטי הזדהות

NT cred:

כל מערכת Windows שומרת את פרטי ההזדהות שלכם **במידה והתחברתם לכוננו רשת בארגון**.
לדוגמא, אם נוריד את הכלי הזה - http://www.nirsoft.net/utils/network_password_recovery.html
ונריץ אותו נקבל (הדוגמא נלקחה ממכונת VM המריצה Win7)

Item Name	Type	User	Password
Domain:target=realdc	Domain Password	realuser	demopass

זאת אומרת שבמידה ופורץ מריץ את הכלי הזה או מקביל אליו על התחנה ומוציא את השם משתמש וסיסמא של ה-Domain Admin הוא יכול להתקדם ברשת כמעט ללא הפרעה. על מנת לשתול פרטים נשתמש בפקודה "cmdkey" באופן הזה:

```
c:\>cmdkey /add:realdc /user:fakeuser /pass:demopass
CMDKEY: Credential added successfully.
c:\>
```

כמובן שנכניס פרטים יותר "מזמינים", אבל לאחר הכנסת 5 משתמשים הפלט החדש יראה ככה:

Item Name	Type	User	Password
Domain:target=realdc	Domain Password	realuser	demopass
Domain:target=realdc2	Domain Password	fakeuser	demopass
Domain:target=realdc3	Domain Password	fakeuser1	demopass
Domain:target=realdc4	Domain Password	fakeuser2	demopass
Domain:target=realdc5	Domain Password	fakeuser3	demopass
Domain:target=realdc6	Domain Password	fakeuser4	demopass

כעת, מנקודת מבט של הפורץ הוא חושב שהוא מצא 5 פרטי הזדהות של משתמשים ברשת הארגון, אבל הוא אינו יודע שרק אחד מהמשתמשים הנ"ל הוא נכון, וברגע שהוא ינסה להזדהות עם הפרטים הלא נכונים ה-MS שלנו יקבל את הלוג מה-AD ויתריע כי המשתמש הכוזב ששתלנו ניסה להזדהות מול ה-AD, כמובן שהמידע יכלול את כתובת התחנה ועוד פרטים נוספים שיעזרו להבנת וגידור האירוע.

חשוב לציין שה-Agent כבר לא רץ על התחנה לאחר השתלת המידע הוא מוחק את עצמו, וגם כן חשוב לציין שרמת ה-False Positive פה היא כמעט אפסית הרי המידע הכוזב מושתל בתחנות קצה ולמשתמש אין שום סיבה לחטט במידע שהושתל ואם כן אז כנראה שתרצו לדעת מזה ☺.

במידה והפורץ הימר ובחר במשתמש אמיתי הסיכויים לתפוס אותו עדיין גבוהים מאוד כי כל תחנה שהוא יעבור ברשת הוא עדיין יצטרך להמר ככה שהסיכויים תמיד לטובתנו ☺.

Cached cred (salted double md4):

Windows שומר את פרטי ההזדהות מול ה-AD ומשתמשים לוקאלים בצורה מוצפנת בשם "NTHashs" (salted double md4) ומאחסן אותם ב-HKLM\Security Hive. ומיקום קובץ ה"כוורת" הוא :

%systemroot%\System32\config\SECURITY

בשביל לגשת ל"כוורת" נצטרך להריץ את ה-Registry Editor דרך חשבון System. על מנת לבצע זאת, נוכל להשתמש בכלי psexec על מנת לעשות זאת עם הפקודה:

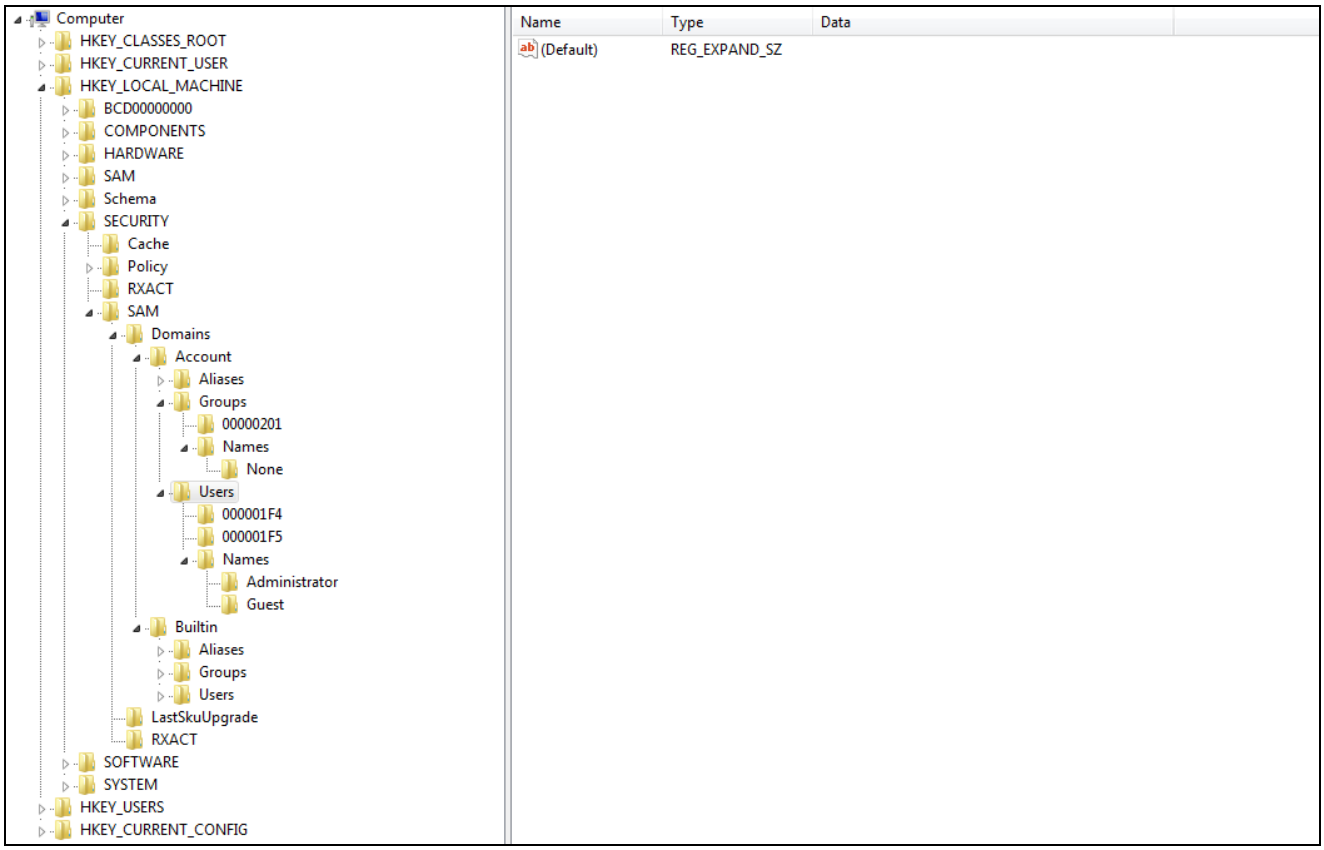
```
Psexec.exe -s -i regedit.exe
```

ניתן להוריד מכאן (http://technet.microsoft.com/en-us/sysinternals/bb896649.aspx).

ואז נוכל לראות את ה"כוורת":

```
HKLM\SECURITY
```

היא נראת כך:



מכאן אפשר להוסיף פרטים (מוצפנים) בצורה ידנית די בקלות. בשביל לראות את המידע בצורה יותר נוחה ומסודרת נשתמש בכלי שנקרא mimikatz שניתן להוריד מכאן:

http://blog.gentilkiwi.com/mimikatz

לאחר הרצת הכלי נשתמש בפקודה:

```
privilege::debug
```

בשביל לבדוק את ההרשאות, הפלט שאתם צריכים לקבל הוא:

```
Privilege '20' OK
```

לאחר מכן נריץ:

```
sekurlsa::logonpasswords
```

והפלט שתקבלו יהיה בסגנון הבא:

```
mimikatz # privilege::debug
Privilege '20' OK

mimikatz # sekurlsa::logonpasswords

Authentication Id : 0 ; 515764 (00000000:0007deb4)
Session           : Interactive from 2
User Name         : Gentil Kiwi
Domain            : vm-w7-ult-x
SID               : S-1-5-21-1982681256-1210654043-1600862990-1000
        msv :
         [00000003] Primary
         * Username : Gentil Kiwi
         * Domain   : vm-w7-ult-x
         * LM       : d0e9aee149655a6075e4540af1f22d3b
         * NTLM     : cc36cf7a8514893efccd332446158b1a
         * SHA1     : a299912f3dc7cf0023aef8e4361abfc03e9a8c30
        tspkg :
         * Username : Gentil Kiwi
         * Domain   : vm-w7-ult-x
         * Password : waza1234/
...
```

ולאחר הוספת הפרטים:

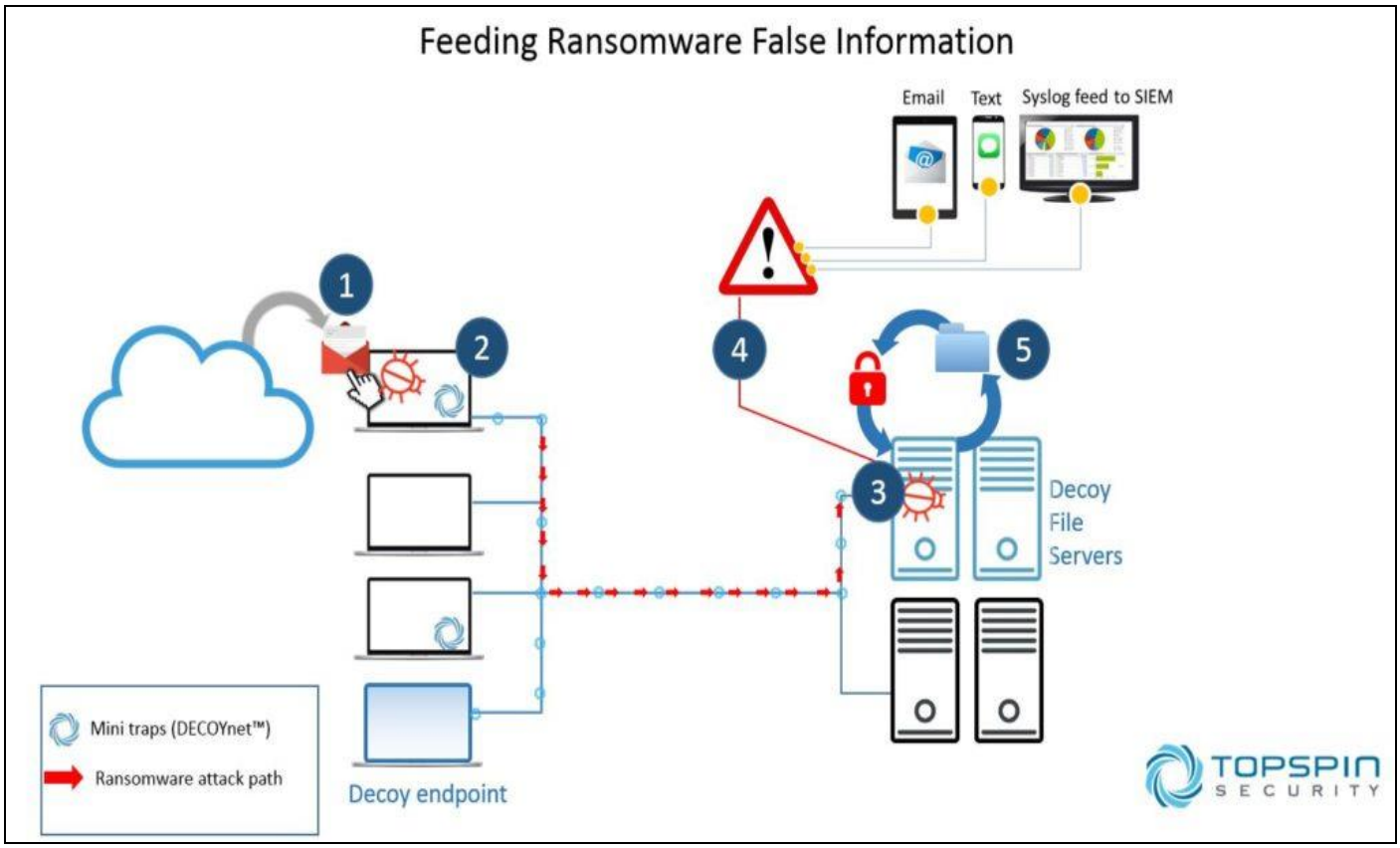
```
mimikatz # sekurlsa::logonpasswords

Authentication Id : 0 ; 515764 (00000000:0007deb4)
Session           : Interactive from 2
User Name         : Gentil Kiwi
Domain            : vm-w7-ult-x
SID               : S-1-5-21-1982681256-1210654043-1600862990-1000
        msv :
         [00000003] Primary
         * Username : Gentil Kiwi
         * Domain   : vm-w7-ult-x
         * LM       : d0e9aee149655a6075e4540af1f22d3b
         * NTLM     : cc36cf7a8514893efccd332446158b1a
         * SHA1     : a299912f3dc7cf0023aef8e4361abfc03e9a8c30
        tspkg :
         * Username : Gentil Kiwi
         * Domain   : vm-w7-ult-x
         * Password : waza1234/

         Authentication Id : 0 ; 9999 (00000000:0007deb4)
Session           : Interactive from 2
User Name         : fakeme
Domain            : vm-w7-ult-x
SID               : S-1-5-18
        msv :
         [00000003] Primary
         * Username : fakeme
         * Domain   : vm-w7-ult-x
         * LM       : d0e9aee149655a6075e4540af1f22d3b
         * NTLM     : cc36cf7a8514893efccd332446158b1a
         * SHA1     : a299912f3dc7cf0023aef8e4361abfc03e9a8c30
        tspkg :
         * Username : fakeme
         * Domain   : vm-w7-ult-x
         * Password : waza1234/

         Authentication Id : 0 ; 85455 (00000000:0007deb4)
Session           : Interactive from 2
User Name         : fakeme
Domain            : vm-w7-ult-x
SID               : S-1-5-19
        msv :
         [00000003] Primary
         * Username : fakeme
         * Domain   : vm-w7-ult-x
         * LM       : d0e9aee149655a6075e4540af1f22d3b
         * NTLM     : cc36cf7a8514893efccd332446158b1a
         * SHA1     : a299912f3dc7cf0023aef8e4361abfc03e9a8c30
        tspkg :
         * Username : fakeme
         * Domain   : vm-w7-ult-x
         * Password : waza1234/
```

כעת, במקום משתמש אחד יש לנו 3 שמתוכם 2 לא קיימים כלל, אך ברגע שהפורץ ינסה להזדהות איתם אנחנו נדע ☺.

קבצים

לאחר שה-MS עשה Deploy לכל ה-Agent-ים על כל התחנות, חוץ ממידע כוזב שמושתל ה-Agent-ים גם מקימים mount לשרת קבצים שלנו שמאחסן קבצי זבל גדולים ובמקביל מנטר את התעבורה בין התחנות אליו ומדווח ישירות לשרת MS שלנו.

במקרה ותחנה מסוימת נפגעה מ-Ransomware, הכופרה תמפה כוננים על התחנה ותתחיל להצפין קבצים. ברגע שהכופרה תתחיל להצפין קבצים בכונן הרשת המנוטר שלנו שרת ה-MS שלנו יזהה קצב כתיבה גבוהה ויתריע לנו, ומעבר לכף - נוכל אף להריץ קוד שיכבה את התחנה הנגועה.
לדוגמא הפתרון של OPSPIN (אותו הרעיון):



תעבורת רשת

כמו שציינתי קודם לכן, אחת הפעולות הכמעט וודאיות שפורץ יעשה לאחר שקיבל גישה לאחת מהתחנות ברשת היא הפעלת Sniffer על מנת לנסות לתפוס פרטי הזדהות ברשת או כל מידע מעניין אחר. וכאן נכנס שרת ה-AIS שלנו ו-2 שרתי ה-Decoy שלנו כמו שניתן לראות בתמונת ההיררכיה של המערכת שלנו.

המבנה מאחורי המודול הזה מאוד פשוט:

1. הקמת שרתי Decoy, לא כל כך רלוונטי איזה שירות הם יריצו (PHPMyAdmin, WebStorage, Interface, Fake Admin Panel).
2. הקוד שרץ על ה-AIS נכתב ב-C והוא מאוד פשוט: בכל פרק זמן מסוים התהליך מתחיל לבצע בקשות POST עם פרטי הזדהות דרך HTTP (ככה שלפורץ לא תהיה בעיה להבין אותם) לשרתי Decoy.

> על מנת ששרת ה-MS שלנו לא יתריע גם על הבקשות POST שמגיעות מה-AIS הוספתי ל-user-Agent שני רווחים בין שם הדפדפן למערכת ככה ששרת ה-MS שלנו יוכל להבדיל בין בקשה "לגיטימית" שמגיעה מה-AIS לבין בקשה שהפורץ מנסה להתחבר בעצמו לאחד משרתי ה-decoy שלנו.

בקשת POST לדוגמא שמגיעה מה-AIS לשירות (PHPMyAdmin):

```
POST /phpmyadmin/index.php HTTP/1.1
Host: 10.0.1.60
User-Agent: Mozilla/5.0  (Windows NT 6.1; Win64; x64; rv:50.0) Gecko/20100101 Firefox/50.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Referer: https://10.0.1.60/phpmyadmin/index.php
Cookie: pmaCookieVer=4; phpMyAdmin=qr66afs9e3tq49qkuu5ddg5b329aqOqv; pma_lang=en; pma_collation_connection=utf8mb4_unicode_ci; pma_console_height=92; pma_console_mode=collapse;
pma_console_config=%7B%22alwaysExpand%22%3Afalse%2C%22startHistory%22%3Afalse%2C%22currentQuery%22%3Atrue%2C%22enterExecutes%22%3Afalse%2C%22darkTheme%22%3Afalse%7D
DNT: 1
Connection: keep-alive
Upgrade-Insecure-Requests: 1
Content-Type: application/x-www-form-urlencoded
Content-Length: 100

pma_username=test&pma_password=1234&server=1&target=index.php&token=351baead555814f9f5ae66dfbde68563
```

האזור שמסומן בצהוב נועד להדגיש את ה-2 רווחים שציינתי מקודם, דרך Regexp ה-MS שלנו יכול להבדיל בין בקשה שמגיעה מה-AIS לבין כל בקשה אחרת. לדוגמא בקשה שתגרום להתראה:

```
POST /phpmyadmin/index.php HTTP/1.1
Host: 10.0.1.60
User-Agent: Mozilla/5.0 (Windows NT 6.1; Win64; x64; rv:50.0) Gecko/20100101 Firefox/50.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Referer: https://10.0.1.60/phpmyadmin/index.php
Cookie: pmaCookieVer=4; phpMyAdmin=qr66afs9e3tq49qkuu5ddg5b329aqOqv; pma_lang=en; pma_collation_connection=utf8mb4_unicode_ci; pma_console_height=92; pma_console_mode=collapse;
pma_console_config=%7B%22alwaysExpand%22%3Afalse%2C%22startHistory%22%3Afalse%2C%22currentQuery%22%3Atrue%2C%22enterExecutes%22%3Afalse%2C%22darkTheme%22%3Afalse%7D
DNT: 1
Connection: keep-alive
Upgrade-Insecure-Requests: 1
Content-Type: application/x-www-form-urlencoded
Content-Length: 100

pma_username=test&pma_password=1234&server=1&target=index.php&token=351baead555814f9f5ae66dfbde68563
```

שימו לב שההבדל מאוד קטן, וגם אם הפורץ יודע שקיימת מערכת כמו שלנו קשה מאוד לזהות את ההבדל בין ה-User-Agent של הבקשות.

מכונות רפאים

אחד הצעדים הראשונים שפורץ יעשה לאחר שחדר לארגון - זה מיפוי הרשת על מנת להתקדם לעבר המטרה, המשימה הזאת לא מסובכת בכלל אבל מה הפורץ יראה לאחר הטמעת מערכת "הונאה" כזאת? ובכן, במקום לראות 10 מכונות לגיטימיות הוא יראה 30 כאשר 20 מתוכן בכלל לא קיימות אבל ברגע שהוא יעשה צעד לכיוון אחת מהמכונות האלו המערכת תזהה ותתריע.

חשוב לציין שהמכונות האלו הן לא מכונות Honeypot, אלא להפך - הן כלל לא קיימות פיזית, הן סתם מידע כוזב שהושתל ברשת. רק לפורץ אשר נכנס לרשת הארגון תהיה סיבה לחפש מידע שכזה ולהשתמש בו, כל המשתמשים הלגיטימיים האחרים ברשת לא צריכים לדעת ולא מושפעים מהמידע הזה כלל וכלל.

על מנת ליצור מכונות "רפאים" נצטרך להקצות כתובות וירטואליות, בדוגמא הזאת השתמשתי במכונה שבשבילנו כרגע היא תהייה ה-AIS אשר מריצה CentOS. לדוגמא, בחרנו את הכתובת 192.168.1.4 אז נריץ:

```
Ifconfig eth0:1 192.168.1.4 netmask 255.255.255.0
```

פעולה זאת תיצור כתובות וירטואלית 192.168.1.4 ו-eth0 יטפל בכל בתעבורה שתגיע לכתובת זאת.

לאחר מכן נעדכן את ה-Routing Table:

```
Arping –q –U –c 3 –I eth0 192.168.1.4
```

```
eth0:1    Link encap:Ethernet  HWaddr 00:50:56:83:4E:E6
          inet addr:192.168.1.4  Bcast:192.168.1.255  Mask:255.255.255.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
```

כעת, תוכלו לראות שהכתובת "חיה" וכל מה שנותר לנו לעשות זה לקשור אותה לפורט מסוים, לדוגמא לפורט 22 שמאחוריו עומד שירות OpenSSH שנראה פגיע. בגלל שכל התהליך צריך להיות אוטומטי, נוח וקל לתפעול צריך לכתוב קוד שינהל את כל ההקצאות של הכתובות וקשירה שלהם לשירותים שונים, אך בשביל הדוגמא הנוכחית נקשור ידנית את שירותOpenSSH לכתובת שהקצנו בפורט 22. נערוך את הקובץ הבא:

/etc/ssh/sshd_config

נדלג ל:

```
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
ListenAddress *
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
```

ונוסיף:

```
ListenAddress 192.168.1.4
```

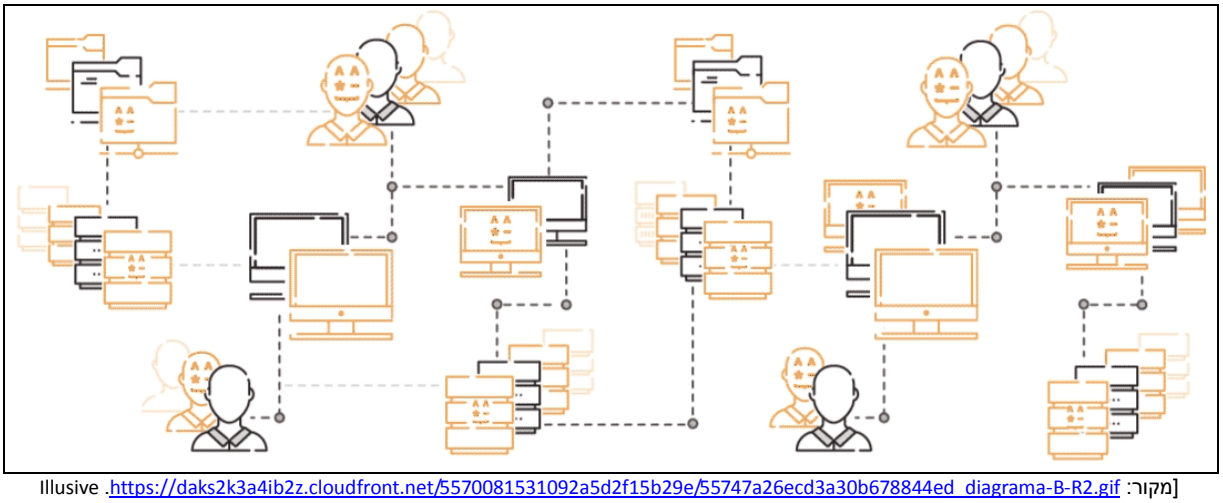
נשמור ונאתחל את השירות.

לאחר מכן, סריקת nmap תראה כך:

```
Nmap scan report for 192.168.1.4
Host is up (0.0082s latency).
Not shown: 999 closed ports
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 6.0p1 Debian 4 (protocol 2.0)
Service Info: OS: Linux; CPE: cpe:/o:linux:kernel
```

וברגע שהפורץ ינסה להתחבר או ליצור כל סוג של תעבורה לשירות שלנו נקבל התראה ☺. המטרה של הצפת הרשת במכונות רפאים היא לגרום לתוקף להמר בכל צעד שהוא יעשה בתוך רשת הארגון וברגע שהוא יהמר לא נכון (הסיכויים לטובתנו תמיד ☺) נתפוס אותו.

חשוב לציין: כל מכונות ה-Decoy שלנו שמריצות את השירותים שונים שהזכרתי מקודם הן גם מכונות רפאים.



[מקור: https://daks2k3a4ib2z.cloudfront.net/5570081531092a5d2f15b29e/55747a26ecd3a30b678844ed_diagrama-B-R2.gif. Illusive networks]

ניטור הרשת

אחרי שהשתלנו את כל המידע אנחנו צריכים לנטר אותו, אני ממליץ על חיבור של Port Mirroring והקמת Snort. למידע על כתיבת חוקים ל-Snort:

http://manual-Snort-org.s3-website-us-east-1.amazonaws.com/node27.html

כמובן שכל אחד והעדפה שלו, אפשר גם לנטר את ה-AD דרך ה-Event Viewer של המכונה וגם לנטר את הבקשות שנשלחות למכונות ה-Decoy שלנו (מכונות רפאים) מקומית מהמכונה, אך בגלל שזה מאוד מסורבל עדיף להשתמש ב-Snort שיזרוק לנו התראות לכל המודולים ביחד.

ה-Snort ירוץ על ה-MS שלנו ויחפש שימוש בפרטים ששתלנו או ניסיונות התחברות לשירותים הפיקטיביים שלנו, במידה וזוהה שימוש בפרטים אלו או התחברות לאחד השירותים שלנו תצא אלינו התראה (אפשרי לחבר למערכת SIEM די בקלות).

למשל על מנת לזהות ניסיונות התחברות לשירות ssh הפיקטיבי שלנו נשתמש בחוק:

```
# alert tcp any any -> 192.168.1.4 22 (msg:"login attempt to decoy";
flow:to_server,established; content:"SSH-";
depth:4; detection_filter:track by_src, count 1, seconds 60;
metadata:service ssh; classtype:misc-activity; sid:19559; rev:5;)
```

וברגע שיהיה ניסיון התחברות למכונת Decoy שלנו נקבל התראה ☺. אם נרצה שכל תעבורה בפורט 22 למכונת רפאים שלנו תגרום להתראה נשתמש ב:

```
alert tcp any any -> 192.168.1.4 22 (msg:"Traffic to decoy";
sid:10001337007;)
```

אך במידה ונשתמש בחוק זה יקרו הרבה מצבים בהם נקבל false positive, לכן עדיף ללכת על בטוח ולאתר רק ניסיונות וודאיים לגשת לשירות רפאים שלנו כמו החוק שהשתמשנו מקודם.

ה-Snort גם יכול לחפש תבנית מסוימת בתעבורת רשת וזה מצוין בשבילנו על מנת לזהות מתי בקשות לא לגיטימיות מגיעות לשירותי רפאים שלנו כמו PHPMyAdmin שציינתי מקודם, נעשה זאת ע"י שימוש בפרמטר content באופן הבא:

```
alert tcp any any -> PHPMyAdmin_ip 80 (content:" Mozilla/5.0 (";
content:"EFG"; http_raw_header;)
```

לאחר שנפעיל את החוק הנ"ל, Snort יחפש ב-headers של כל בקשה שתגיע לשירות שלנו את התבנית הבאה "Mozilla/5.0 (ובגלל שיש רווח אחד בין שם הדפדפן לפרטי המערכת כמו בכל בקשה רגילה ולא כמו בקשות שיוצאות מה-AIS שלנו, אנחנו נקבל התראה רק על בקשות לא לגיטימיות שמגיעות מה-AIS שלנו ☺.

למרות ש-Snort לדעתי הכי יעיל, אסביר בקצרה על הוצאת אירועים ישירות מה-Event Viewer של ה-AD על מנת לזהות שימוש בפרטי הזדהות ששתלנו. ה-"event id" שאנחנו מחפשים הוא 4625 שבעצם אומר ש"פרטי הזדהות אינם נכונים".

לאחר חיפוש של ה"event id" הספציפי הזה נקבל:

Audit Failure	22/09/2016 09:37:36	Microsoft Windows security auditing.	4625	Logon
Audit Failure	21/09/2016 11:52:22	Microsoft Windows security auditing.	4625	Logon
Audit Failure	21/09/2016 11:50:21	Microsoft Windows security auditing.	4625	Logon
Audit Failure	21/09/2016 11:48:20	Microsoft Windows security auditing.	4625	Logon
Audit Failure	21/09/2016 11:46:19	Microsoft Windows security auditing.	4625	Logon
Audit Failure	21/09/2016 11:42:17	Microsoft Windows security auditing.	4625	Logon
Audit Failure	21/09/2016 07:48:22	Microsoft Windows security auditing.	4625	Logon
Audit Failure	21/09/2016 07:46:21	Microsoft Windows security auditing.	4625	Logon
Audit Failure	21/09/2016 07:44:20	Microsoft Windows security auditing.	4625	Logon

אם נסתכל על הפרטים של אחת מהרשומות הנ"ל, נראה את הפרטים הבאים:

```
Failure Information:
        Failure Reason:         Unknown user name or bad password.
        Status:                 0xc000006d
        Sub Status:             0xc0000064

Process Information:
        Caller Process ID:  0x0
        Caller Process Name:    -
```

Log Name:	Security
Source:	Microsoft Windows security
Event ID:	4625
Level:	Information
User:	N/A
OpCode:	Info

כמובן שנרצה לקבל התראה רק על שימוש בפרטים ששתלנו ולא טעויות לגיטימיות של משתמשים ברשת, לכן אני ממליץ (למי שלא יודע לכתוב קוד) להשתמש ב-netwrix שניתן להוריד מכאן:
https://www.netwrix.com/

שבעצם לאחר הגדרה פשוטה יסנן רק את האירועים שמכילים את הפרטים המושתלים שלנו ויוציא לכם התראה למייל או יריץ תוכנית אחרת רק תבחרו.

לסיכום

עד כאן ראינו עד כמה גישה זאת יכולה להיות יעילה בהתמודדות מול האיומים השונים, במקום כל פתרונות ההגנה השונים שיש כיום אשר פועלים נגד הקוד הזדוני גישה זאת פועלת נגד הפורץ עצמו, זאת לחלוטין חשיבה מחוץ לקופסא.

בין כל היתרונות של מערכת כזאת חשוב שוב להזכיר את רמת ה-False Positive שכמעט ולא קיימת, הרי כל מערכות ה-SIEM היום למיניהן דורשות השקעה עצומה וממושכת על מנת להגיע לרמת False Positive נמוכה, במערכת הזאת False Positive יקרו רק במקרים מאוד נדירים.

כל הדרכים להשתלת מידע שהצגתי כאן נעשו בצורה ידנית, ברגע שיש API (אצלי כתוב ב-c) שלוקח את כל המידע מהמשתמש דרך הממשק (אצלי כתוב ב-PHP) ומתחיל לשתול אותו ברחבי הרשת דרך Deploy

של Agent-ים לתחנות השונות ברשת, ויצירת מכונות רפאים וקשירתן לשירותים שונים, הכל הופך יותר פשוט ונוח לתפעול.

המערכת שלי רצה כבר למעלה משבועיים וזיהתה באופן כמעט מידי Ransomware כבר פעמיים. אז נכון אין למערכת שלי ממשק משתמש מפואר או כל מיני פיצ'רים נוחים כמו למערכת מהמדף אבל זה לא אומר שהיא יעילה פחות ☺.

על המחבר

שמי אביחי כהן, אני מתעסק בעיקר במחקר חולשות, אני מאוד שמח לתרום לסצינה בארץ בעיקר מפני שלמדתי רבות מהמאמרים במגזין.
לכל שאלה ניתן לפנות אלי דרך כתובת האימייל: MR.B3ND3R@GMAIL.COM.

מבוא ל-Transportation Cyber Security

מאת יובל סיני

מבוא

מערכות המחשוב בתחבורה (Transportation Computer Systems) הן נדבך חשוב בחיי היום יום של מרבית הציבור, אם לא כולו. מערכות מחשוב אלו מושתתות בעיקרן על משפחת מערכות תפעוליות מסוג ICS / SCADA (Industrical Control System / Supervisory Control and Data acquisition) - המנטרות ומבקרות את התשתית הדיגיטלית המשמשת לניהול התשתית הפיזית.

בין מערכות המחשוב בתחבורה השכיחות ניתן למנות את מערכת הרמזורים והאיתותים (Signaling) האחראית בין השאר לניתוב תעבורה בכבישים ומסילות הרכבת, ומתן תיעדוף לתחבורה הציבורית בהתאם לצורך. מערכת בקרה אווירית(Air Traffic Control) אשר אחראית לניתוב תעבורה במרחב האווירי (Air Space) ופועלת בשילוב עם מערכת נחיתת מכשירים (Instrument Landing System) לשם יישום תהליכי אוטומציה של שלב הנחיתה וההמראה. מערכת בטיחות (Safety System) הקיימת כמעט בכל אמצעי תחבורה ציבורית, ואשר מטרתה לזהות ולהגיב במקרה של אירוע בטיחות, כדוגמת התפרצות שריפה או אבדן שליטה של הנהג. מערכת בקרת סביבה (Heating, Ventilating and Air conditioning) אשר מטרתה להתאים את התנאים הסביבתיים לצרכי קהל הלקוחות (אנשים) ואמצעי התחבורה. מערכת תשלומים (Payment System), וזאת כדוגמת מערכת הרב-קו הקיימת מזה תקופה במדינת ישראל, ואשר יש לה נקודות ממשק רבות, כדוגמת נקודות מכירה (Point of Sales) פיזיות ואינטרנטיות.

בשנים האחרונות ניתן לזהות מגמה להטמעת מערכות ברכבים פרטיים וציבוריים אשר מטרתן לאפשר תקשורת דו-כיוונית בין הנהג ו/או אמצעי התחבורה לסנסורים ומרכזי מידע שונים. וכך לדוגמא, אמצעי ניווט המובנים ברכבים מסתמכים באופן ניכר על טכנולוגית ה-GPS (Global Positioning System) אשר מאפשרת אף ניווט אוטונומי לעבר היעד. דוגמא אחרת הינה שימוש בפתרונות ניווט מתקדמים מבוססי רשתות חברתיות (Social Networks) ו"חוכמת ההמונים" (Crowdsourcing), וזאת כדוגמת Waze.

עוד יצוין כי כבר כיום מוטמעות מערכות מבוססות SIM 3G/4G ברכבי היוקרה המעדכנות, בנוגע לתקלות ברכב, רדיו מבוסס אינטרנט הכולל מערכת ניווט ושליטה, מערכות איכון ובחינת התנהגות נהג, כדוגמת Pointer IQ ו-Ituran Starlink, יכולות מעקב אחר התנהגות הולכי רגל בתחבורה ציבורית לדוגמא בהתפרצות לכביש בצמתים עמוסים ואף נעילת רכב/אוטובוס במקרה של סכנה. אפליקציות למובייל

המתקשרות ב-Bluetooth לכלי הרכב, אשר בתורו מדווח למרכז השירות (ישירות או בעקיפין) על תקלה פוטנציאלית או מבצע דיאגנוסטיקה באון-ליין (באמצעות תשאול ממשק J533[1] או ממשק אחר).

"האינטרנט של הדברים" (Internet of Things) הביא לעולם את יכולת אמצעי התחבורה לתקשר עם הסביבה באופן דו-כיווני[2], ובכך לשפר את חווית הנהיגה ורמת הבטיחות. יכולות בינה מלאכותיות (Artificial Intelligence) מקנות כיום לאמצעי התחבורה יכולת פעילות אוטונומית. שדרוגים אוטומטיים של רכיבי התוכנה מרחוק, וזאת כדוגמת שימוש בטכנולוגיית FOTA (Firmware Over The Air) ובטכנולוגיית SOTA (Software Over The Air). אף ציודי הבדיקה המסורתיים עברו שינוי דרסטי, והם כוללים כיום התממשקות דיגיטלית ליעד הנבדק, והן ל"שירותי ענן" (Cloud Services) המספקים מידע, מעקב היסטורי אחר הרכב והנהג ועדכוני תוכנה.

עם זאת, למרות היתרונות הגלומים במערכות המחשוב בתחבורה ובטכנולוגיות השונות, אין הן חסינות בפני איומי אבטחה מידע שכיחים.

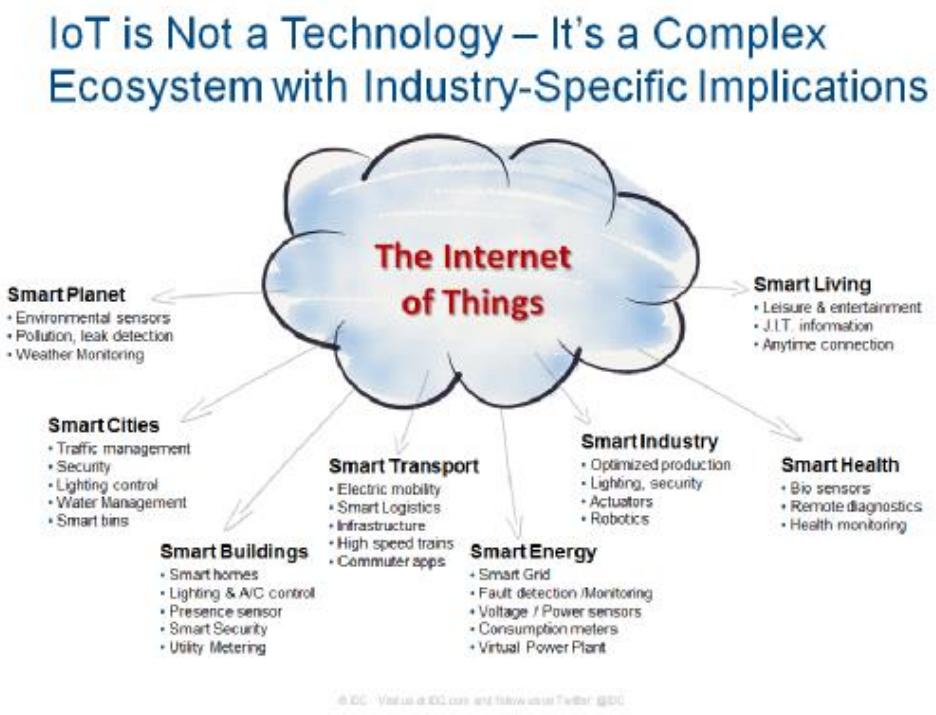
מאמר זה מציג את האיומים השכיחים ביחס למערכות המחשוב בתחבורה, וזאת על מנת להבנות בסיס ידע אשר יוכל לסייע לקורא להכיר את עולם זה. בנוסף, המאמר סוקר מספר המלצות ראשוניות לשם התמודדות עם האיומים השונים. עם זאת, אין מאמר זה מתיימר להציג את כל עולם הידע בנושא, לא שכן הצגה פרטנית נושא כזה או אחר.

[1] כינוי חלופי - CANBUS Diagnostics Interface

[2] כינויים שכיחים לטכנולוגיות מסוג אלו: V2V (Vehicle-to-Vehicle), V2I (Vehicle-to-Infrastructure) V2F (Vehicle-to-Field), V2P (Vehicle-to-Pedestrian))

"האינטרנט של הדברים" (Internet of Things) על קצה המזלג

"האינטרנט של הדברים" מהווה מערכת אקולוגית (Ecosystem) אשר כוללת תחת מטרייתה שורה של מימושים טכנולוגיים ותהליכים עסקיים (Business Process). התרשים הבא מציג חלק מהמימושים השכיחים:

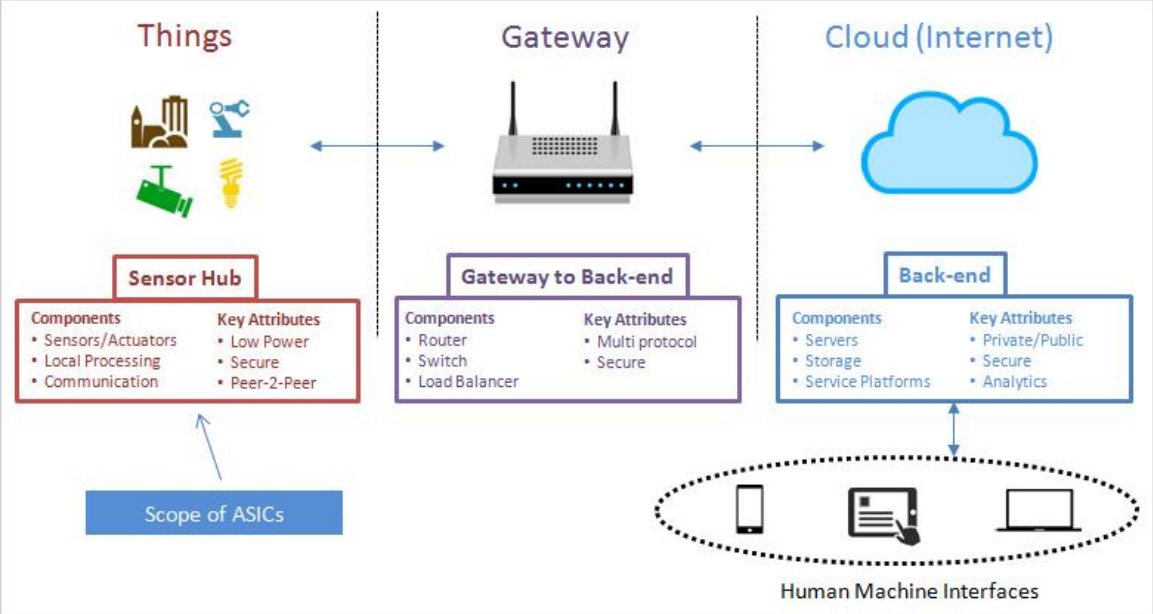


3

בהתאם, ניתן לזהות כי "האינטרנט של הדברים" כולל בחובו אף את נושא מערכות המחשוב בתחבורה, והאינטגרציה מול מערכות משיקות נוספות, אשר מהוות תשתית למערכת אקולוגיות נוספות, כדוגמת "ערים חכמות" (Smart Cities).

התרשים הבא מציג את מאפייני התקשורת השכיחים בעת עבודה עם "האינטרנט של הדברים", וזאת החל משלב קבלה/שליחה של נתונים באמצעות יחידות קצה (כדוגמת סנסורים), שכבת התווך התקשורתי (כדוגמת תשתית סלולרית LTE או תשתית תקשורת אחרת) וכלה בשכבת השירותים, אשר מבוססת בד"כ על תשתית ענן (Cloud Based).

[3] מקור: https://www.vmware.com/ciovantage/article/3-essentials-for-your-iot-toolkit



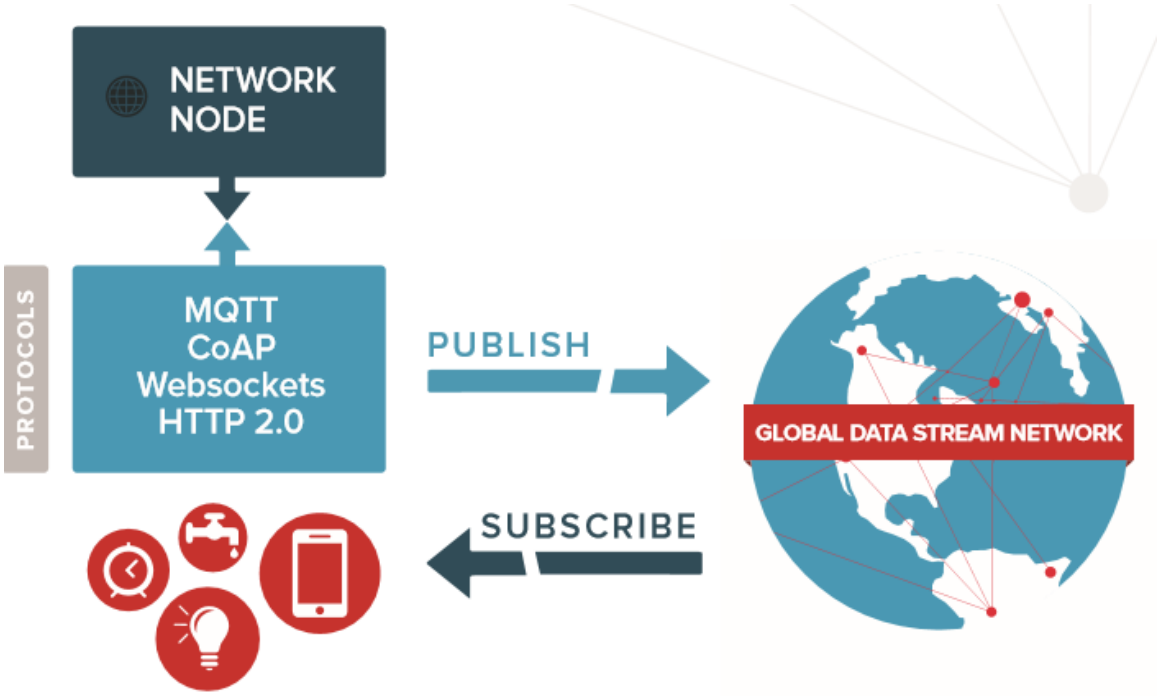
בהתאם ניתן לזהות מספר סוגיות אבטחתיות מהותיות בנדון, כדוגמת שטח התקיפה (Attack Surface) הפוטנציאלי הנרחב של מערך "האינטרנט של הדברים", התלות הענפה בזמינות מדיום התקשורת, ומגבלות שונות במאפייני יחידות הקצה, אשר מונעים החלה של אמצעי הגנה מקובלים. וכך לדוגמא, עקב הצורך בצמצום צריכת החשמל ע"י יחידות הקצה (כדוגמת סנסורים), יחידות הקצה אינן תומכות כבחירת מחדל באלגוריתמי הצפנה שכיחים, ומקובל לראות כי יחידות הקצה אינן כוללות תמיכה בהצפנה כלל, או לחילופין כוללות תמיכה בהצפנה מסוג Light-Weight Data Encryption, אשר נחשבת חלשה יחסית. סוגיה זו זוכה ליתר חשיבות עקב העובדה כי שיטת הזיהוי השכיחה בין "מכונה למכונה" (Machine to Machine - M2M) לדוגמא, מבוססת על זיהוי מבוסס הצפנה או שימוש בזיהוי המבוסס על תשתית קרטוגרפית "מוחלשת" (Lightweight Cryptography).

דוגמא אחרת היא המורכבות הגבוהה של הפרוטוקולים בהם נעשה שימוש, וזאת כדוגמת Google Wave Federation Protocol, המנסים למזג שורה של עולמות תוכן שונים (כדוגמת מסרים המידיים (Instant Messaging), דואר אלקטרוני, עריכת מסמכים, שיחות קוליות ווידיאו) דבר המגדיל את הסבירות לכשל לוגי ו/או תכנותי, אשר יצרו חולשות אשר יהיה ניתן לנצלן לרעה.

כמו כן, בתרשים הבא ניתן לראות כי "האינטרנט של הדברים" כולל שימוש במגוון פרוטוקולים, טכנולוגיות וממשקים, לרבות שימוש במודל משיכה/דחיפה (Pull\Push), וזאת בדומה ליכולות המובנות כיום ב- HTML 5.x.

עם זאת המגוון הרב של הפרוטוקולים, הטכנולוגיות והממשקים מעלה לדיון את האפשרות לקיומן של בעיות תאימות בין יצרני פתרונות שונים, וכן את העובדה כי למרות קיומם של סטנדרטים בתחומים

נוספים, הם לא מספקים מענה הוליסטי, ואף מכילים חולשות אבטחה הידועות כבר כיום. במאמר מוסגר יצוין כי סוגיה זו מזכירה בעיות אבטחת המידע אשר התגלו החל מתחילת עידן האינטרנט, לרבות העובדה כי חלק ניכר מהסטנדרטים לא כללו התייחסות פרטנית לנושא אבטחת המידע, ולאור זאת ניתן לאתר אף כיום חולשות המובנות במוצרי מחשוב שונים, אשר מקורם בסטנדרטים לא שלמים.



4

לשם ההמחשה של חשיבות "האינטרנט של הדברים" בעולם התחבורה, ניתן להציג שורה של מימושים שכיחים, וזאת כדוגמת העברת חיווי סטאטוס תקינות צמיג הרכב לנהג באמצעות ממשק Bluetooth או RF (Radio Frequency), שליטה מלאה או חלקית על כל הרכב באמצעות טלפון חכם או אביזר מחשוב לביש, דיווח מיקום באופן אוטומטי למרכז ביטחון, אשר מסוגל אף להשבית את כלי הרכב מרחוק במקרה של חשד לגניבה, חיבור אמצעי התחבורה הציבוריים למערכת שליטה ובקרה מרכזית, אשר מאפשרת איתור חריגות והתמודדות עם עומסים חריגים, מתן תיעדוף לתעבורה ציבורית ע"י שינוי אקטיבי של פרופיל ההתנהגות של הרמזורים בצמתים, חיוב לקוח בתשלום בגין שימוש בתחבורה ציבורית על סמך הזדהות מבוססת מפתח פרטי (Private Key) המאוחסן באביזר מחשוב לביש או התקן נלווה, ודיווח "זמן אמת" מבוסס תקשורת סלולרית ביחס למיקום והסטאטוס של משאיות משא. רכבות בעולם לדוגמא באנגליה, צרפת, גרמניה ואף בישראל עוברות לשימוש ב-"ניהוג"; קרי ניתן לבצע שליטה מלאה על הקטר ממרכז הבקרה באמצעות שימוש בהתקני שליטה מבוססי GSM-R ללא צורך בנהג אמיתי.

[4] מקור: A New Approach to IoT Security - 5 Key Requirements to Securing IoT Communications, PubNud, נדלה ב-18.09.2016

במאמר מוסגר, מר ארז מטולה, מייסד חברת AppSec Labs העביר בכנס OWASP ישראל 2016 הרצאה מעניינת בנושא Hacking The IoT (Internet of Things) PenTesting RF Operated Devices. מצגת ההרצאה זמינה להורדה מהלינק הבא.

לסיום פרק זה, נציין כי לאור העובדה כי כל יצרן עשוי לייצר פתרון מסוג "האינטרנט של הדברים", אין כל ערבון כי הפתרון המוצע יענה לדרישות אבטחת מידע נאותות, לא שכן לדרישת תאימות בין פתרונות שונים. לאור העובדה מערכות מחשוב בתחבורה הולכות לכיוון אימוץ נרחב של עולם "האינטרנט של הדברים", סביר להניח כי הסוגיות השונות ישפיעו אף מסגרת האיומים בעולם זה.

איומים שכיחים במערכות המחשוב בתחבורה

בשנת 2014 שתי סטודנטיות מהטכניון בנו, במסגרת פרוייקט בפקולטה למדעי המחשב, מערכת הגורמת לתכנת הניווט הפופולארית "Waze" לדווח על פקקים מדומים. באמצעות התכנה שבנו, הצליחו הסטודנטיות ליצור פקק שנמשך שעות וגרמו לנהגים לסטות מדרכם. המנחים שלהן הודיעו על המתקפה ל-"Waze", פירטו את הדרך שבה בוצעה ונענו על ידי סגן הנשיא לתפעול כי החברה בודקת דרכים למנוע מתקפות כאלה."[5]

הקלות שבה הצליחו הסטודנטיות לממש את המתקפה מהווה דוגמא להשלכות רוחביות של תקיפת סייבר על תשתית תחבורה קריטית, אשר תוכל לפגוע בשגרת היום של הציבור, לגרום נזק כלכלי ניכר ואף לגרום לפגיעה בחיי-אדם. בנוסף, רק בחודש אפריל 2016 התברר שניתן לעקוב אחר רכבים ואנשים באמצעות ניצול חולשות במערכת של Waze על-ידי התחזות לרכבים מדומים אשר עוקבים אחר נהגים אחרים סביבם ללא הפרעה.

סיבוני ואסף חילקו במאמרם "קווים מנחים לאסטרטגיה לאומית במרחב הסייבר" את ההתקפות במרחב הקיברנטי לשלוש קטגוריות, וזאת בהתאם למטרת ההתקפה, וכנגזרת, לעתים, במתווה ובכלים:

א. תקיפה לצורך פגיעה, הרס ומחיקה (Computer Network Attack - CNA) - תקיפה שתכליתה גרימת נזק למחשב/רשת ומניעה של המשך תפקודם התקין. הנזק יכול להיות ברמת השבתה לפרק זמן מוגבל, לדוגמה: מתקפה מסוג מניעת שירות (Denial of Service), או שינוי חזות לאתר (Defacing) ואף מחיקה של מידע, השבתה של המחשב ושיתוק תהליכים נתמכי מחשב בארגון הנתקף בהתקפת עומק (Advanced Persistent Threat - APT).

ב. תקיפה לצורך הפקת מידע/ריגול (Computer Network Exploiting - CNE) - תקיפה שתכליתה איסוף מידע. המידע יכול להיות טכנולוגי - על מבנה הרשת והמחשבים - לצורך מימוש מאוחר יותר של תקיפת CNA, או איסוף נתונים לצורך מימוש פעילות אקטיבית עתידית (כדוגמת איסוף נתוני כרטיסי

[5] מקור: http://www.technion.ac.il/2014/03/%D7%A1%D7%98%D7%95%D7%93%D7%A0%D7%98%D7%99%D7%95%D7%AA-%D7%91%D7%98%D7%9B%D7%A0%D7%99%D7%95%D7%9F-%D7%9E%D7%99%D7%9E%D7%A9%D7%95-%D7%9E%D7%AA%D7%A7%D7%A4%D7%AA-%D7%A1%D7%99%D7%99%D7%91%D7%A8-%D7%A2/ , נדלה ב-18.09.2016

אשראי או נתוני זהות של משתמשי דואר אלקטרוני), והוא יכול להיות איסוף מידע תוכני (גניבה של מידע מסחרי, מחקר ופיתוח או סודות צבא ומדינה).

ג. תקיפה לצורכי השפעה, פסיכולוגית בעיקרה (Computer Network Influence - CNI) - תקיפות מסוג זה נועדו לטעת את התחושה של חוסר ביטחון, חוסר שליטה, פגיעה בריבונות, פגיעה מוראלית בציבור וחוסר יכולת להגן על אורח החיים הנורמטיבי. תקיפות כאלה יהיו בדרך כלל מוגבלות בזמן, ולא יגרמו נזק ממשי זולת התחושות הללו."[6]

לשם נוחות ההצגה, מאמר זה יאמץ את גישתם של סיבוני ואסף בעת הצגת האיומים השכיחים במערכות המחשוב בתחבורה.

תקיפה לצורך פגיעה, הרס ומחיקה (Computer Network Attack - CNA)

בהתאם לפרסומים הקיימים עד כה ניתן להסיק כי הבעיה המהותית ובעלת דרגת הסיכון הגבוהה ביותר ברמה הציבורית בשלב זה הינה רמת החשיפה הגבוהה של מערכות המחשוב בתחבורה לאיומים מקטגוריה זו.

באמצעות שיבוש פעילות של מערכת מחשוב בתחבורה מרכזית לדוגמא, גורם עוין עשוי לגרום לשורה של "תקלות", אשר מטרתן לגרום לנזק כלכלי ואף לפגיעה בחיי אדם. תקיפה שכיחה מסוג זו מוכרת אף בכינוי "ירוק צולב", המאופיינת במצב שבו גורם עוין מצליח לגרום לקיומה של תעבורה אמצעי תחבורה בכיוונים מנוגדים, וזאת במטרה לייצר התנגשות הדדית. בהינתן כי תקיפה מסוג זו מצליחה, התוקף עשוי אף להשבית צירי מפתח למשך זמן ארוך, ליצור תאונת שרשרת רבת נפגעים, ואף למנוע את האפשרות להגעת גורמי חירום והצלה לאתר האירוע. כמו כן, כפי שהודגם אף באופן מעשי ע"י צוות החוקרים מהטכניון, ניתן לייצר דיסאינפורמציה אשר תגרום לשיבוש תנועה, ואף לעיתים לפגיעה ברכוש ונפש.

דוגמא נוספת לתקיפה אשר הוצגה ב-Black Hat[7] כללה השתלטות עוינת על כלי רכב, וזאת באמצעות מחשב נייד מרוחק אשר קיבל שליטה על ה-CAN bus[8] (Controller Area Network), דבר אשר התאפשר בין השאר עקב נגישותו של ממשק דיאגנוסטיקה (שלא אובטח באופן נאות) לגישה מרחוק. דוגמא אחרת לתקיפה שהודגמה כללה העברת שדרים כוזבים לסנסורי בטיחות הרכב החיצוניים, אשר בתורם העבירו דיווח כוזב למערכות הבטיחות של הרכב כי ישנה סכנת התנגשות, דבר אשר חייב הפעלת רוטינת חירום. במקרה של תקיפה אמיתית סביר להניח כי גורם עוין היה מסוגל לגרום לתאונת דרכים רבת נפגעים, גם אם הוא היה מצליח לשטות במערכת המחשוב של רכב בודד.

---

[6] http://www.inss.org.il/uploadImages/systemFiles/memo149.pdf , נדלה ב-18.9.2016

[7] בכנס Black Hat USA 2016 הועבר קורס פריצה מעשי (Hands On) לכלי רכב ע"י חברת CanBusHack, Inc.

[8] "פרוטוקול תקשורת מתוקנן שתוכנן לאפשר למיקרו-בקרים והתקנים אחרים לתקשר האחד עם השני, וזאת כדוגמת קישור בין מערכת ההגה והסנסורים השונים". מקור: https://he.wikipedia.org/wiki/CAN-bus, נדלה ב-20.9.2016

תקיפה אחרת מבוססת על פגיעה בזמינות תשתית ה-GPS (Global Positioning System), לרבות העברת שדרי GPS כוזבים[9], וזאת על מנת להשבית ו/או להטעות את מערכות הניווט עליהן אמצעי התחבורה נסמך. עוד יצוין כי הודגם לא פעם כי ניתן לטעון Malware למערכת המחשוב של כלי רכב, וזאת באמצעות ניצול ממשקי כלי רכב שונים, כדוגמת קורא card SD, ממשק USB ו-Bluetooth. לפיכך יתכן כי בעתיד גורמים עוינים יעשו שימוש בתקיפות מבוססות "תוכנת כופר" (Ransomware), וזאת במטרה לסחוט את בעל הרכב ו/או נוסעי הרכב.

כדוגמא אחרונה אציין את העובדה כי תקיפה של מערכות מחשוב בתחבורה עשויה להוות שלב הכנה לתקיפה פיזית מסורתית, כאשר באמצעות תקיפה זו הגורם העוין עשוי למנוע או לעכב את יכולת הכינוס של כוחות הביטחון בחירום.

תקיפה לצורך הפקת מידע/ריגול (Computer Network Exploiting - CNE)

סביר להניח כי שכיחות תקיפה מסוג זו נמוכה יחסית, בסייג למקרים שבהם יעשה שימוש באמצעי תשלום, וזאת כדוגמת תקיפת תשתית נקודות מכירה (Point of Sales) או כשלב מקדים למימוש תקיפה מקטגוריה אחרת. וכך לדוגמא, התגלה בעבר כי חלק מיצרני הרכבים ציידו בעבר את שלט הרכב במנגנון הזדהות בעל חולשה אינהרנטית, אשר באמצעות Replay Attack או מתקפה אחרת, מקנה לגורם עוין אפשרות לקבל גישה לרכב. יצוין כי החולשה הנ"ל לא הסתיימה בפגיעות רכבים מיצרן אחד בלבד, אלא התגלה כי רכבים אשר נבנו ע"י יצרנים שונים, חשופים אף הם לפגיעות דומה, לרבות העובדה כי היצרנים הטמיעו ברכבים קודי אבטחה זהים (ולא חד-ערכיים/אישיים בשילוב מנגנון אבטחה מקובל, כדוגמת Challenge\Response).

סוגיה נוספת שאינה זוכה להתייחסות ענינית ומעמיקה בקטגוריה זו באופן ישיר, הינה סוגיית הפרטיות (Privacy), לרבות איסוף מידע מזהה אישי (Personally identifiable information - PII) למטרות השגת הישגים כלכליים, פרסום ממוקד, סחר במידע אישי או סחיטה של נשוא המידע. יצוין כי מזה תקופה מתקיים במדינת ישראל פיילוט אשר מטרתו לבחון את התנהגות הנהגים, אם למטרות מיסוי ואם למטרות אחרות. כמו כן, חלק מחברות הביטוח מציעות כיום חבילות ביטוח המותנות בהתקנת תוכנת מעקב אשר התנהגות נהג הרכב. עם זאת, בחברה הישראלית לא התקיים עד כה דיון ציבורי מקיף בנושא, דבר אשר מאפשר חופש פעולה רב למדי לגורמים השונים, וכל זאת ללא ביקורת ציבורית נאותה.

תקיפה לצורכי השפעה, פסיכולוגית בעיקרה (Computer Network Influence - CNI)

באמצעות תקיפה זו ארגון טרור (סייבר טרור) או גורם המתחרה בשוק עשוי להשיג יתרון תודעתי, אשר יאפשר לו למנף את מטרותיו, אם למטרות כלכליות, פוליטיות-חברתיות או למטרה אחרת. לשם מימוש

[9] וקטור תקיפה זה נוצל ע"י אירן בשנת 2011 לשם חטיפת מזל"ט אמריקאי. התקרית מוכרת בכינוי Iran–U.S. RQ-170 incident

תקיפה מצומצמת אין התוקף נדרש למשאבים רבים, ואף חלק ניכר מאמצעי התקיפה נגישים בשוק החופשי.
כמו כן, קטגוריה נוספת שאינה נופלת בחלוקה הנ"ל הינה כשלים מובנים, אשר עשויים לנבוע מכשל טבעי ו/או כשל אנושי של רכיב מחשוב או אחר.

המלצות ראשוניות להגנה על מערכות המחשוב בתחבורה

מודעות צרכנים

התפיסה המקובלת אצל יצרנים רבים הינה כי אבטחת מידע לא מהווה גורם אשר משפר מכירות, ולפיכך הם נמנעים מלהשקיע תקורות נאותות בתחום. לאור זאת, הזרז העיקרי לשיפור רמת אבטחת המידע של מערכות המחשוב בתחבורה הינו שיפור תודעת הצרכנים, אשר בתורם ידרשו הן מהיצרנים, והן מגורמי הרגולציה השונים לנקוט בצעדים הנדרשים לשם צמצום מסגרת הסיכון למינימום סביר.

החלת רגולציה בתחום מערכות המחשוב בתחבורה

בדומה לרגולציית ה-PCI והנחיות הפיקוח על הבנקים המוכרות מעולם הפיננסים, סביר להניח כי רגולציה דומה תושת על עולם התחבורה. ניסיונות למיסוד רגולציה מסוג זו מתבצעים מזה תקופה ע"י U.S NHTSA (National Highway Traffic Safety Administration). כמו כן, ברית יצרני המכוניות (Alliance of Automobile Manufacturers) פרסמה ביולי 2016 מסגרת עבודה (Framework) לנושא Cybersecurity, אם כי ראוי לציין כי מדובר ברגולציה על בסיס התנדבותי, ולפיכך לא ברור מהי רמת הישימות שלה בפועל. בנוסף כיום פועלות מספר קבוצות של חברות שונות בתחום התעופה - לדוגמא ICAO & CANSO אשר הפנימו את הסיכון הגלום בסייבר לתחום התעופה ונתנו לו קדימות וחשיבות גבוהה ובכך מנסות לכונן רגולציה ומסגרות הגנה מסודרות בעולם. בארה"ב ה-TSA (Transport Secuirty Administration) מוביל מספר קבוצות עבודה הכוללות אף את ישראל לטובת פיתוח מסגרות הגנת סייבר על כלל תחומי התחבורה הציבורית ההמונית - לרבות מטוסים, רכבות, מנהרות, שדות תעופה ועוד.

אבטחה פיזית

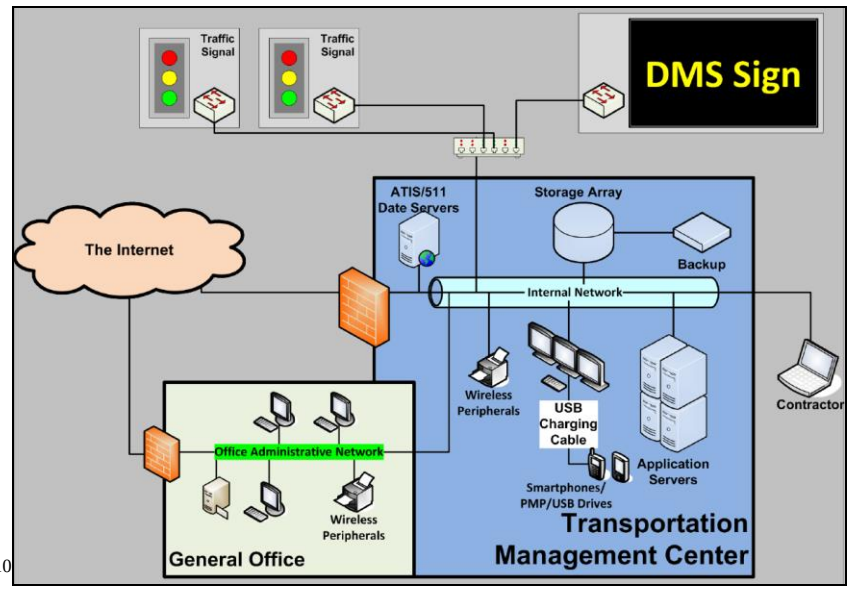
חלק לא קטן מהרכיבים אשר משרתים את מערכות המחשוב בתחבורה פעילים באזורים שונים, דבר אשר מחייב תפירת פתרון אבטחה מתאים. פתרונות מקובלים הינם; מידור פיזי של הרכיב, כך שגורמים זרים לא יוכלו לגשת אליו, שימוש באמצעי מניעת שימוש (Anti Tampering) שונים אשר מטרתם להקשות על קבלת גישה פיזית לרכיב, שימוש במערכת מצלמות במעגל סגור (Closed-Circuit Television) וסיורים יזומים של צוותי ביטחון.

אימוץ מתודות מעולם ה-Cybersecurity הקיים

ברמה העקרונית, מערכות המחשוב בתחבורה אינן שונות באופן מהותי ממערכות מחשוב מוכרות, כדוגמת SCADA (Supervisory Control and Data Acquisition) ו-DCS (Distributed Control System). לאור זאת, ניתן לאמץ מתודות מעולם ה-Cybersecurity הקיים - לרבות NIST CSF Cyber Security Framework, וכדוגמת פיתוח מאובטח ובניית מודל איומים (Threat Modeling) ברמת המיקרו (כדוגמת מודל איומים לכל Electronic Control Unit או לכל ממשק) והמאקרו (מודל איומים ייחודי לכל אמצעי התחבורה), ביצוע בדיקות חוסן, בידול לוגי ופיזי של סביבות בהתאם לדרגת רגישות, ובכך לשפר את רמת אבטחת המידע של הפתרונות הקיימים בתחום.

להלן דוגמא למספר טעויות שכיחות בעת הקמת פתרונות לשידור אותות דיגיטליים (Digital Messaging Solutions) \ מערכת איתות, וזאת כדוגמת:

- חיבור רשת המנהלתית המחוברת לאינטרנט ישירות לרשת מערכת האיתות.
- מתן גישה לספקים לרשת הייצור האחראית למערכת האיתות ללא פיקוח, ואף התרה של הכנסת מחשב נייד אישי של הספק למתחם העבודה.
- מתן אפשרות לעובדים לחבר לעמדות העבודה ברשת מערכת האיתות את הטלפונים הסלולריים שלהם. התוצאה הברורה מאליה היא כי מדובר בווקטור תקיפה שכיח.
- שימוש בתשתית אלחוט שאינה מאובטחת דיה.
- העדר סגמנטציה רשתית בהתאם לעקרונות "הצורך לדעת" (Need to know) ומתן "הרשאות נמוכות" (Least Privilege).
- העדר רכיב מידור (כדוגמת NGFW או SCADA FW) אשר ימנע אפשרות להעברת כל סוג שדר/תעבורה, לרבות מניעת העברת שדר זדוני אשר מקורו מרמזור (לדוגמא), לעבר ליבת מערכת האיתות.
- העדר פתרונות לזיהוי אנומליה ברבדים השונים ברשת (משתמשים, תקשורת, אפליקציה וכו').
- העדר אתר DRP (Disaster Recovery Plan) וקיומה של מערכת שרידה.



[10]

הטמעת פתרונות הגנה ברמה חומרתית

לאור העובדה כי מערכות המחשוב בתחבורה מוטמעות באמצעי התחבורה הסופי, ממשק המשתמש (טלפון סלולרי או ממשק אחר), סנסורים פנימיים וחיצוניים ואף בשרתים ושירותים נלווים, נדרש לעיתים לספק אבטחה חומרתית, אשר תצמצם את האפשרות לביצוע שינוי לא מבוקר ע"י גורם עוין.

להלן דוגמא למימושים מקובלים בתחום:

- אתחול בטוח (Secure Boot) של מערכת ההפעלה / קושחה (Firmware), ושימוש בפונקציות תוכנה אשר תדווח על כל בעיה בשלמות המידע (Attestation Functions), וזאת על מנת להשבית רכיב חומרה ו\או קוד תוכנה המעלה חשד, וזאת עד להחלפתו. דוגמא שכיחה למימוש זה הינה שימוש במודולי קוד מאומתים (Authenticated Code Modules), אשר בהינתן קיום חומרה מתאימה (כדוגמת מעבד תומך טכנולוגיית Intel® Platform Protection), מאפשרים שמירה על סודיות והגנה על שלמות ואימות מסר בכל תוואי זרימת המידע (Chain of Trust).
- שימוש בטכנולוגית הרצה בטוחה (Trusted Execution Technology) אשר מספקת ולידציה חומרתית למהימנות מערכת ההפעלה ו\או אפשרות לזיהוי המשתמש על בסיס מפתח פרטי (Private Key). דוגמא למימוש יכולת זו הינו רכיב TPM (Trusted Platform Module), הכולל אף יכולות קריפטוגרפיות מתקדמות.
- שימוש במעבד (CPU) המכיל תמיכה בקיומן של שכבות בידול מובנות (secure Vs. non-secure), כאשר הסביבה "הבטוחה" (secure) מאפשרת אחסון מידע רגיש והפעלת תהליכים רגישים באזור

[10] מקור: Transportation Cyber Security, Edward Fok, Federal Highway Administration – Resource Center Operations Technical Service Team, נדלה ב-14.2.2015

נפרד ואוטונומי, הנגיש למספר מצומצם של אפליקציות "בטוחות" ומוכרות. טכנולוגיית ARM TrustZone מהווה דוגמא למימוש מסוג זה.

- שימוש במודל אמון שורשי (Root of Trust), המספק רמת שלמות (Integrity) ואותנטיות (Authenticity), אשר לאחר הקמתו הוא לא ניתן לשינוי, וכפועל יוצא מכך ניתן לסמוך על רמת מהימנותו הגבוהה. דוגמא למימוש יכולת זו הינה טכנולוגיית Intel EPID (Enhanced Privacy ID) אשר מאפשרת לרכיב חומרתי להציג פרטים על עצמו ואת התאמתו לפעילות משותפת של קבוצת רכיבים, כאשר מצד אחד הרכיב לא מסגיר מידע רגיש על עצמו (שומר על אנונימיות), אך מצד שני הרכיב מספק רמת וודאות נאותה שני הצדדים הינם "מהימנים".
- האצת הצפנה (Encryption Offloading) ודחיסת נתונים (Compression Offloading) ברמת חומרה (ASIC), וזאת לשם קיצור זמן הקבלה והשליחה על המידע, וכן על מנת לאפשר החלפת מפתחות הצפנה בתדירות גבוהה יותר.
- Proof Caring Data - חיוב כל פעולה בהצגת הוכחה כי החישוב הינו נכון. נציין כי מימוש זה הינו חלק מתפיסת מודל Multilevel Security, אשר קצרה היריעה מלדון בה במאמר זה. שורה של מקורות מידע רלוונטיים זמינים לעיון בפרהסיה, וזאת כדוגמת "הספר הכתום" (Orange Book), אשר מהווה דוגמא בסיסית וראשונית לניסיון אימוץ גישה זו. דוגמא אחרת הינה תקן משרד ההגנה האמריקאי TCSEC (Trusted Computer System Evaluation Criteria).
- הטמעת רכיבי "Firewall" ייעודי אשר מטרתם לחסום שדרים עוינים/ניצול של פונקציות אסורות. מימוש לדוגמא של רכיב מסוג זה הינו CAN Bus Firewall, אשר מגדיר "רשימה לבנה" (Whitelist) של פרמטרים, וזאת כדוגמת פרטי הרכיב השולח את השדר, כיוון תנועת השדר, סוג השדר ומבנה השדר.

הטמעת פתרונות הגנה ברמה תוכנתית

- מעבר לאפליקציות וירטואליות, וזאת כדוגמת שימוש בטכנולוגית Docker אשר מצמצמת מרחב הנגישות של אפליקציה למערכת ההפעלה ככלל, ומערכת הקבצים בפרט.
- אימוץ גישת קוד פתוח (Open Source), וזאת על מנת לנצל את יכולות הקהילה לאיתור בעיות אבטחה. להלן דוגמא לפלטפורמות קוד פתוח בתחום; Ostro, Brillo אשר תומכות במעבדים מבית Intel, ARM ואף יצרני מעבדים נוספים. כמו כן, יצרניות נוספות, כדוגמת Microsoft מאמצות אף הן את גישת הקוד פתוח, וזאת כדוגמת אימוץ תקן OPC (Open Platform Communications) המאפשר שיתוף מידע ב"אופן בטוח" בין רכיבים תעשייתיים שונים.
- חיזוק ממשקי ההזדהות (Authentication) ובקרת גישה (Access Control), דבר אשר עשוי לכלול אינטגרציה עם יכולות אבטחה בחומרה, וזאת כדוגמת TPM או מפתח רכב המכיל "מפתח פרטי" (Private Key) של בעל הרכב.

- בניית לוגיקה מתקדמת לקבלת החלטות, על מנת לצמצם את הסיכון לקבלת החלטות שגויות כתוצאה מקבלת שדרים כוזבים ממספר חיישנים מצומצם. בכלל זה עשויה להיבחן אפשרות להקמת רשות תחבורה אשר בסמכותה להשבית אוטומטית פעילות או לטול את השליטה על אמצעי תחבורה אשר דרגת הסיכון שלו תעלה על המותר במרחב.
- החלת הצפנה מקצה לקצה (End to End Encryption) ובדיקת מהימנות כבחירת מחדל של שדרי הממשקים הפנימיים והחיצוניים (לרבות Provider to Provider Communication). בין המימושים השכיחים ניתן למצוא שימוש באלגוריתם HMAC (Keyed-Hashing for Message Authentication) וניצול יכולות האבטחה המובנות בפרוטוקול TLS (Transport Layer Security). דוגמא אחרת הינה אימוץ Oauth 2.0 וזאת במטרה לספק מסגרת אמון משותפת (Common Trust Framework) בין המשתמש הסופי (End User) לספקי השירות (Service Providers) השונים.

הגנה על תשתית הענן (Cloud)

תשתית הענן (Cloud) מכילה כיום מידע רב על המשתמשים, מידע קריטי בנושא ניווט, תהליכי קבלת החלטות ובטיחות, ולפיכך נדרש לוודא כי תשתית זו מוגנת בפני האיומים השונים. לאור הספרות הענפה אשר נכתבה בעברית בנושא, לרבות סדרה של מאמרים אשר פורסמה בעבר ב-Digital Whisper, מוצע כי גורם אשר מעוניין להרחיב את הידיעה בנושאים אלו יעיין בפרסומים הנ"ל.

אבטחת שרשרת האספקה (Supply Chain Security)

שרשרת האספקה מוכרת מזה מספר שנים כנקודת תורפה של מרבית הארגונים בעולם, ולפיכך יצרני הפתרונות השונים בתחום יאלצו (אם באופן רצוני, ואם באופן לא רצוני וזאת עקב הטלת רגולציה) לאמץ מתודולוגיות אשר יצמצמו את הסיכונים מווקטור תקיפה זה. דבר זה עשוי לכלול החלת בדיקות קבלה פרטניות של רכיבים קריטיים, וכן החלת הרגולציה העתידית אף על ספקי משנה.

הכללת מערכות המחשוב בתעבורה כחלק אינטגרלי ב-BCP (Business Continuity Planning)

לאור העובדה כי אין מערכת אשר חסינה באופן מוחלט לאיומים טבעיים ולא טבעיים, נדרש כי הגופים אשר מספקים שירותי תחבורה יכינו עצמם להתמודדות עם מקרים של פגיעה במערכות המחשוב בתעבורה, לרבות מעבר לעבודה באתר חירום מרוחק וביצוע תרגולים יזומים.

סיכום

מערכות המחשוב בתחבורה מהוות מרכיב מהותי בהתקדמות החברה האנושית. עם זאת, מערכות אלו חשופות מטבע הדברים לאיומים שונים, וזאת כדוגמת פגיעות לאיומי אבטחת מידע. למרות ההתקדמות הטכנולוגית בשנים האחרונות, חלק ניכר מפתרונות מערכות המחשוב בתחבורה ו"האינטרנט של הדברים" עדיין חשופים לפגיעויות מסורתיות, דבר אשר מאפשר לגורם עוין לנצל חולשות אלו לרעה.

במאמר זה הוצגו חלק מהאיומים הקיימים על מערכות המחשוב בתחבורה, תוך הצגה של מספר פתרונות לדוגמא. עם זאת, ללא תפיסת אבטחה הוליסטית, וזאת החל משלב התכנון, וכלה בניהול מחזור החיים (Life Cycle Management) ושלב הגריטה, סביר להניח שאיומים אלו יהיו רלוונטיים יותר מתמיד אף בשנים הבאות.

> "In five to ten years' time when everything is connected, someone could stop all the large trucks, for example, and cause serious disruption as a diversion while they do something else."
>
> Daniel Miessler

תודות

ברצוני להודות למר אורן אלימלך, מומחה וחוקר אבטחת מידע וסייבר, ראש אגף סייבר במשרד התחבורה, מייסד חברת חברת CST360º ועמית מחקר במכון למדיניות נגד טרור במרכז הבינתחומי בהרצליה בהתמחות על איומי סייבר-טרור בתחבורה, בהעשרת המאמר בתכנים רלוונטים. כמו כן, ברצוני להודות למר מיקי (מיכאל) שאודר, מומחה אבטחת מידע וסייבר, על המשוב שסיפק בעת כתיבת המאמר. בנוסף, ברצוני להודות לגב' הדס שני מליק, מומחית הגנה בסייבר על המשוב שסיפקה בעת כתיבת המאמר. תודה למר ארז מטולה, מייסד חברת AppSec Labs, על הסכמתו להכללת המצגת אשר העביר בכנס OWASP ישראל 2016 במאמר.

על המחבר

יובל סיני הינו מומחה אבטחת מידע, סייבר, מובייל ואינטרנט, חבר קבוצת SWGDE של משרד המשפטים האמריקאי. כמו כן, יובל סיני קיבל הכרה מחברת Microsoft העולמית כ-MVP בתחום Enterprise Security ו-Cloud and Datacenter Management.

מילות מפתח

CAN Bus, Car Hacking, Chain of Trust, IoE, Internet of Everything, IoT, Internet of Things, M2H, Machine-to-Human, M2M, Machine-to-Machin, Root of Trust, Smarts Cities, Smart Transport, TCS, Transportation Cyber Security, V2V, Vehicle-to-Vehicle, V2I, Vehicle-to-Infrastructure, V2P, Vehicle-to-Pedestrian

ביבליוגרפיה

ביבליוגרפיה באנגלית

מאמרים:

- Assessing the Cyber Threat to the Train Industry, Oren Elimelech and Nir Tordjman, International Institute for Counter Terrorism (ICT), April 2016
- Hackers take Remote Control of Tesla's Brakes and Door locks from 12 Miles Away
- New Details on Google's Brillo and Weave
- Exploring the Hacker Tools of Mr Robot
- Car hacking: you ain't seen nothing yet!
- Microsoft introduces new open-source cross-platform OPC UA support for the industrial Internet of Things
- Ransomware's next target: Your car and your home
- Automotive Cybersecurity Best Practices, AUTO-ISAC, July 2016
- Automotive Security Best Practices, Recommendations for security and privacy in the era of the next-generation car, McAfee
- Developments in Car Hacking, Roderick Currie. SANS, December 5th, 2015
- OWASP Top 10 Privacy Risks Project
- iBeacon vs NFC vs GPS: Which Indoor Location Technology will your Business Benefit from
- 14 Year Old Hacks Car with Homespun Kit with Circuits Bought From Radio Shack, Feb 2015
- Cyber security and Critical National Infrastructure, Dr Richard Piggin, Atkins
- Automotive Electronic Control Systems Safety and Security
- Adventures in Automotive Networks and Control Units, Dr. Charlie Miller, Chris Valasek, 2014
- Hackers Remotely Kill a Jeep on the Highway - With Me in It
- The Future is Now: Car Hacking, Dimitar Kostadinov
- Progressive Snapshot Exposes Drivers to Car Hacking
- ITS Israel Magazine- November 2014
- Connected Vehicle Assessment, Cybersecurity and Dependable Transportation, System Assurance, Operations and Reactive Defense for Next Generation Vehicles, Intelligent Highway Infrastructure, and Road User Services (Version 2), Steven H. Bayless, Sean Murphy, Anthony Shaw, 2014

  IoT (Internet of Things) & IoE (Internet of Everything)

- Lightweight Cryptography for the Internet of Things, Masanobu Katagi, Shiho Moriai, Sony Corporation
- A New Approach to IoT Security - 5 Key Requirements to Securing IoT Communications, PubNud
- Securing the Internet of Things: A Proposed Framework
- How to secure the virtual world and IoE?, 2015
- Internet of Things, IoT Governance, Privacy and Security Issues EUROPEAN RESEARCH CLUSTER ON THE INTERNET OF THINGS, IERC, January 2015
- Internet of Things, Privacy & Security in a Connected World, FTC Staff Report, JANUARY 2015
- Internet of Things - OWASP Top 10
- Appliances vulnerable to cyber-attack, 2014
- Internet of Things - Privacy and Security issues, 2014

קטעי וידאו:

- Cyber Security in Transportation: Hype or Armageddon
- ICS Security in Rail Transit Control and Communication
- Hackers Remotely Kill a Jeep on the Highway-With Me in It

ספרות:

- Protecting Transportation: Implementing Security Policies and Programs, R William Johnstone, Butterworth-Heinemann, 2015
- Understanding Homeland Security, Gus Martin, SAGE Publications, Inc, 2014
- The Cybersecurity Partnership Between the Private Sector and Our Government: Protecting Our National and Economic Security, Science, and Transportation and the Committee on Homeland Security and Governmental Affaris United States Senate Committee on Commerce Create Space Independent Publishing Platform, 2014
- Introduction to Transportation Security, Frances L. Edwards, Daniel C. Goodrich, CRC Press, 2012
- Transportation Security (Butterworth-Heinemann Homeland Security), Clifford Bragdon, Butterworth-Heinemann, 2008

מצגות:

- Hacking The IoT (Internet of Things) PenTesting RF Operated Devices, Erez Metula, AppSec Labs, OWASP Israel Meeting 2016

- Transportation Cyber Security, Edward Fok, Federal Highway Administration – Resource Center Operations Technical Service Team
- http://www.slideshare.net/orenelimelech/cyber-security-intransportation2015
- Brillo/Weave Part 1: High Level Introduction, Bruce Beare, Open IoTSummit, April 2016

ביבליוגרפיה בעברית

מאמרים:

- קווים מנחים לאסטרטגיה לאומית במרחב הסייבר, גבי סיבוני ועופר אסף, המכון למחקרי ביטחון לאומי, מזכר 149, אוקטובר 2015
- המלצות להפחתת חדירות סייבר לארגונים, גרסה 1.0, המרכז הלאומי להתמודדות עם איומי סייבר, יוני 2015
- פיצול מידע בשירותי ענן, מריוס אהרונוביץ, גליון 43, Digital Whisper, יולי 2013
- תקני אבטחת מידע במחשוב ענן, שחר גייגר מאור, גליון 41, Digital Whisper, מאי 2013
- הענן והמידע שלך, עו"ד יהונתן קלינגר, גליון 19, Digital Whisper, אפריל 2011
- אבטחת מידע בעולם העננים, עידו קנר ואפיק קסטיאל, גליון 27, Digital Whisper, דצמבר 2011
- פני הטרור העתידיים - סייבר-טרור, קרין תמר שפרמן, פרלמנט | גיליון 59 | טרור ודמוקרטיה אחרי ה-11 בספטמבר, המכון הישראלי לדמוקרטיה
- סטודנטיות בטכניון מימשו מתקפת סייבר על "Waze"
- האם אנו שולטים באמצעי התחבורה הממוחשבים שלנו?, מגזין הרכב גלגלים, 2014
- מערכות תבוניות שיתופיות Cooperative-ITS הדור הבא של ה- IS ?, זאב שדמי, יחידת המדען הראשי, משרד התחבורה והבטיחות בדרכים, 2014

קטעי וידיאו:

- מתוך: אירוע חדשנות של EcoMotion, אוניברסיטת ת"א ואריאל, פברואר 2014, בית התפוצות
- ארז קריינר, Transportation CyberSecurity
- זיו לוי, Transportation CyberSecurity
- ישראל פלדמן, Transportation CyberSecurity
- ישראל רון, Transportation CyberSecurity
- ערן טרומר, Transportation CyberSecurity
- רם לוי, Transportation CyberSecurity
- אורן אלימלך https://www.ict.org.il/Article/1650/Assessing-the-Cyber-Threat-to-the-Train-Industry

הזלגת זיכרון ב-Nexus 5x דרך USB

מאת רועי חי

הקדמה

תקיפה פיזית של מכשירים סלולריים הינה סיכון שהיצרניות מתייחסות אליו בכובד ראש. לתקיפה יכולות להיות מטרות שונות - מגניבה מוצלחת ועד פורנזיקה. דוגמאות לא חסר, ואחת הבולטות מביניהן היא ללא ספק מקרה סן ברנרדינו, שבו ה-FBI ביקש מ-Apple "לפתוח" עבורו את מכשיר ה-iPhone של המפגע.

הכלל הוא שבהינתן נגישות פיזית של תוקף למכשיר נעול, לא אמורה להיות לו שום אפשרות להזליג מידע רגיש של הקורבן. כמובן שזו בעיה קשה, ולכן יש מספר מנגנונים על מנת לממשה, הן ב-iOS והן ב-Android. מאמר זה הוא על חולשה באנדרואיד, ולכן אתמקד בו.

מנגנוני ההגנה באנדרואיד סביב תקיפות פיזיות הם מגוונים, וכוללים את היכולות הבאות:

1. Full Disk Encryption - data/ (שמכיל מידע אישי של המשתמש) מוצפן עם מפתח שתלוי בסוד (סיסמא \ קוד \ תבנית) שהמשתמש מספק בזמן עליית המערכת, Salt ומידע שקיים ב-TEE ואינו נגיש באופן אפליקטיבי.
2. נשים לב שתיאורטית, אם זו היתה ההגנה היחידה בלבד, התוקף היה יכול לגשת פיזית ובאופן זמני למכשיר, להחליף את מערכת ההפעלה במערכת זדונית, ולהחזיר את המכשיר לקורבן. בשלב זה התוקף פשוט היה מחכה למידע שיהיה זמין (לא מוצפן). כדי להתמודד עם בעיה זו, מכשירי אנדרואיד נעולים לא מאפשרים לצרוב מערכת הפעלה חדשה, מבלי לעשות להם קודם unlock, דבר הגורר Factory Reset ומחיקת מידע המשתמש.
3. כמו כן, קיים מנגנון בשם Verified Boot, אשר מונע שימוש במערכת הפעלה זדונית, במידה והתוקף הצליח איכשהו כן לשנות את מערכת ההפעלה. מכשירים נעולים יסרבו לרוץ, מכשירים בלתי נעולים יתריאו מיד למשתמש.
4. Factory Reset Protection (FRP) - נועד לא לאפשר לגנבים לעשות Factory Reset למכשיר, ובכך למנוע את כדאיות הגניבה.

אבל מה עם **הזיכרון**? מה מונע מהתוקף להעתיק את זיכרון המכשיר, שבאופן כמעט ודאי מכיל מידע רגיש?

השאלה הזו נשאלה ע"י מספר חוקרים בעבר, ולכן לא מפתיע שניתן למצוא מספר מחקרים בנושא. (למשל: FROST).

הפגיעות

לאחרונה הצוות שלי, IBM X-Force Application Security, גילה חולשה, אשר לא תועדה מעולם, בגירסאות ישנות של ה-bootloader של Nexus 5X. החולשה איפשרה לתוקף פיזי "לשאוב" את כל זיכרון המכשיר, אפילו אם הוא נעול, דרך USB. את התקיפה עצמה ניתן לבצע בשניות ספורות, ללא שימוש בכלים מיוחדים - לכן ניתן לבצעה גם עם נגישות זמנית בלבד למכשיר. כמו כן, בתנאים מסויימים היא ניתנת לניצול גם ע"י שימוש במטען זדוני. (וקטור זה דורש ש-ADB יהיה דלוק על המכשיר, ושהקורבן יאשר למטען להתחבר.)

התקיפה מתחילה בכך שהתוקף מרסט את המכשיר למצב מיוחד של ה-bootloader, שחושף ממשק, דרך USB, בשם fastboot. ממשק זה במכשירים בלתי-נעולים מאפשר בין היתר לצרוב מערכת הפעלה חדשה, אולם במכשירים נעולים הוא אמור להיות די אנמי, ולא לאפשר לבצע שום שינוי במערכת ההפעלה, ו/או לגנוב מידע רגיש.
להפתעתנו גילינו פקודה די מעניינת, הזמינה במכשירי Nexus 5X בלבד:

```
fastboot oem panic
```

בעזרת פקודה זו ניתן לכפות קריסה ב-bootloader:

הקריסה עצמה כמובן שהאירה את עינינו, אולם היא אינה בעייתית כשלעצמה. מה שכן בעייתי היא ההודעה הבאה:

> "If you want to get ram dump, Please do the following action,
> 1) Please connect USB
> 2) Get the ram dump image using QPST Configuration."

זו כבר פגיעות! בגרסאות הפגיעות של ה-bootloader, ברגע שהאחרון קורס, הוא חושף ממשק סריאלי מעל USB, אשר מאפשר לתוקף לשאוב את כל זכרון המכשיר, בעזרת כלים כגון QPST Configuration.

כדי להוכיח את חומרת הפגיעות, שיניתי את סיסמת המכשיר שברשותי ל - buggybootload3r, וחיפשתי אותה בזיכרון שהדלפתי:

```
> hexdump DDRCS0_0.BIN | grep -10 bootloa
2675d060: 6f 00 69 00 64 00 2e 00 - 73 00 65 00 72 00 76 00 o.i.d...s.e.r.v.
2675d070: 69 00 63 00 65 00 2e 00 - 67 00 61 00 74 00 65 00 i.c.e...g.a.t.e.
2675d080: 6b 00 65 00 65 00 70 00 - 65 00 72 00 2e 00 49 00 k.e.e.p.e.r...I.
2675d090: 47 00 61 00 74 00 65 00 - 4b 00 65 00 65 00 70 00 G.a.t.e.K.e.e.p.
2675d0a0: 65 00 72 00 53 00 65 00 - 72 00 76 00 69 00 63 00 e.r.S.e.r.v.i.c.
2675d0b0: 65 00 00 00 00 00 00 00 - 3a 00 00 00 00 0d c4 b6 e.............
2675d0c0: 6d 42 cd 0a b1 00 00 00 - 00 00 00 00 00 00 00 00 mB..............
2675d0d0: 00 00 00 00 00 92 86 33 - e3 79 92 8b b7 d4 77 f5 ......3..y....w.
2675d0e0: 94 7f d0 2b fb b8 6e cc - 98 3b 9a a7 0d 7c 60 f6 .....n.....
2675d0f0: d7 70 68 c2 14 01 00 00 - 0f 00 00 00 62 75 67 67 .ph.........bugg
2675d100: 79 62 6f 6f 74 6c 6f 61 - 64 33 72 00 62 75 67 67 ybootload3r.bugg
```

כפי שניתן לראות, הסיסמא מופיעה ב-dump! ברגע זה, התוקף הפיזי יכול לעלות את המכשיר, להתחזות לקורבן, ולגשת למידע פרטי ששמור ב-data/.

את הפגיעות דיווחנו כמובן ל-Google. הם אישרו על דבר קיומה, ושהיא תוקנה בגירסה MHC19J 6.0.1 אשר שוחררה במרץ 2016. פרטים נוספים ניתן למצוא בבלוג של IBM Security.

בזאת אנחנו סוגרים את הגליון ה-76 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 אוקטובר.

אפיק קסטיאל,

ניר אדר,

30.9.2016