

# Digital Whisper

גליון 59, מרץ 2015

מערכת המגזין:

אפיק קסטיאל, ניר אדר

מייסדים:

אפיק קסטיאל

מוביל הפרויקט:

שילה ספרה מלר, אפיק קסטיאל

עורכים:

יובל סיני, גל תא שמע ו-d4d.

כתבים:

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 /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](mailto:editor@digitalwhisper.co.il)

---

## דבר העורכים

---

ברוכים הבאים לגיליון ה-59 של DigitalWhisper!

אח, זה פשוט לא נגמר. נראה שכל חודש-חודשיים חברת אנטי-וירוס אחרת מפרסמת אודות "הוירוס החזק ביותר שהתגלה עד כה", לפני מספר חודשים שמענו מפי חברת Symantec אודות Regin - "כלי הריגל המורכב ביותר שזוהה אי-פעם" והחודש נראה כי תורה של חברה Kaspersky לצאת עם פרסומים בסיגנון דומה, הפעם אודות ה-"Equation Group". מעבר להמלצה לקרוא את הדו"חות הטכניים והמעניינים (מאוד) שפורסמו ב-SecureList אני באמת לא מבין את הרעש מסביב העניין. אם הרעש היה בעקבות הטכנולוגיה - הייתי מצטרף לדיונים, אבל ברוב המקרים הרעש הוא מסביב ל"איום" שאותו וירוס יוצר.

תמיד בשבוע-שבועיים לאחר פרסומים כאלה, כל אתרי החדשות קוראים לעדכן את כל המערכות, השרתים, האנטי-וירוסים וכו', אבל איכשהו ההיגיון שלי אומר לי שזאת בדיחה, שאם במערכת של שרת שאני מנהל התגלתה איזו חולשה שמאפשרת לגורם זר להריץ עליה קוד, כל רשתות הבוטנטים יגיעו אלי הרבה יותר מהר מאשר Regin או כל וירוס אחרי בסיגנון. לפי איך שאני רואה את זה, בתור אזרח מהשורה, כל עוד אני לא מתכוון לייצר פצצת אטום בשנה-שנתיים הקרובות, נראה שלא הם האיום היומי שלי, אלא וירוסים כמו Zeus, Carberp או שפעת.

דוגמא טובה לחוסר ההבנה של האיום האמיתי היא הרעש והפרסום שנעשה בעקבות כל הפרסומים האלה. כמעט באותו הזמן ש-Kaspersky הוציאו את White Papers שלהם אודות ה-"Equation Group", הם פרסמו White Paper אודות קמפיין APT בשם "Carbanak": קמפיין שבגיננו נשדדו מספר לא קטן של בנקים, נגנבו מעל לחצי מיליארד דולר ונרשמו הפסדים של יותר ממיליארד דולר. יש לא מעט פרסומים אודות הקמפיין הנ"ל בתקשורת הישראלית (הרי אין שום סיכוי שאתרי החדשות בארץ יפספו כזה סיפור חם אודות אירוע סייבר), אך ההתייחסות לכך היא יותר כאל "עוד סיפור מעניין" ואין הבנה כי זה האיום היומי שלנו וניתנים פחות צעדים קונקרטיים כיצד ניתן להמנע מכך. זה נכון שכשמדובר במתווה העבודה של Carbanak למשתמש הקצה אין יותר מדי מה לעשות, אך הרעיון הוא אותו רעיון.

לפי איך שאני מבין את המפה (ותרגישו חופשי לא להסכים איתי), בתור אזרח פשוט האיום היומי שלי הוא לא שה-NSA יעקבו אחרי אלא וירוסים "פשוטים" יותר, עם מניעים "אזרחיים" יותר, כגון וירוסים מסוג Ransomwhare שינסו לנעול לי את המחשב עד שלא אשלם להם, או כל ה-Bot-Net-ים שהמטרה שלהם היא לגנוב כרטיסי אשראי או כאלה שמבצעים מתקפות MITB ומנסים לרוקן לנו את חשבונות הבנק מבלי שנרגיש.



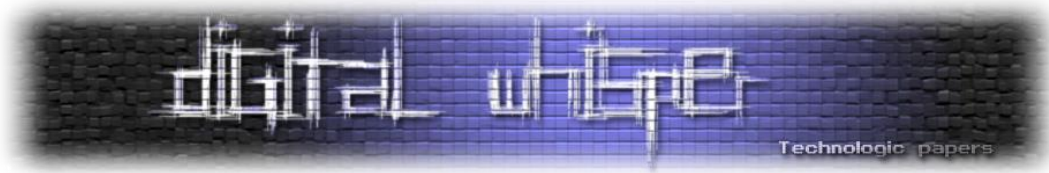
וירוסים בסיגנון של Regin או Stuxnet מאוד מעניינים ברמת הטכנולוגיה, אך ממש לא ברמת האיום על הרשת הפרטית שלי בבית או על הרשת הארגונית שלי (שוב, כל עוד אני לא מעשיר אורניום בחצר האחורית). נראה שלא הרבה אנשים מבינים את זה, ואולי זאת הסיבה שעדיין ניתן לראות היום מחשבים נגועים ב-Conficker.

ולפני שנגיע לחלק הבאמת מעניין של הגיליון, נרצה להגיד תודה רבה לחבר'ה שבזכותם הגיליון ה-59 פורסם החודש! חבר'ה שהשקיעו מזמנם הפרטי לטובת הקהילה. **תודה רבה ליובל סיני, תודה רבה לגל תא שמע ותודה רבה ל-d4d!**

וכמובן, תודה רבה לעורכת שלנו: שילה ספרה מלר, שאין לי מושג איך היא עדיין מצליחה להחזיק מעמד איתנו ☺

## קריאה מהנה!

נר אדר ואפיק קסטיאל.

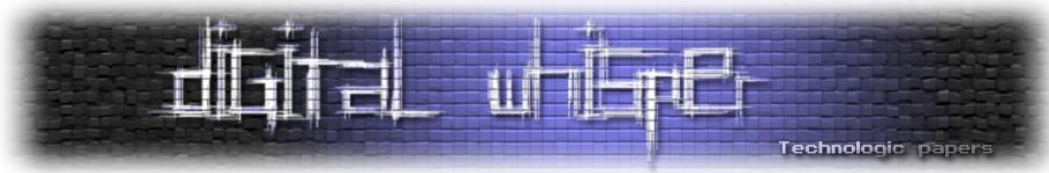


---

## תוכן עניינים

---

2	דבר העורכים
4	תוכן עניינים
5	מבוא לשימוש ביכולות Machine Learning בפתרונות אבטחת מידע וסייבר
43	The Husky Code
54	חלק ג' - Hacking Games For Fun And (Mostly) Profit
69	דברי סיכום



---

# מבוא לשימוש ביכולות Machine Learning בפתרונות אבטחת מידע וסייבר

מאת יובל סיני

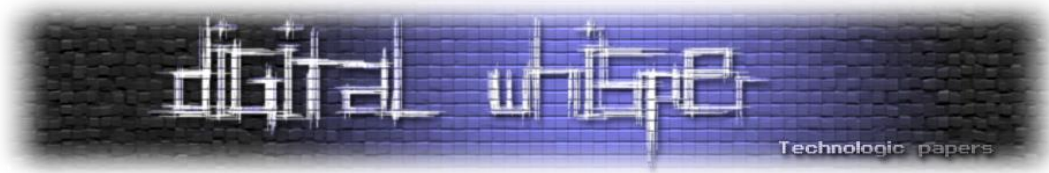
---

## מבוא

מזה תקופה ארוכה ארגונים נאלצים להתמודד עם כמויות גדלות והולכות של מידע, דבר הכולל מידע המאוחסן בפורמטים שונים ומגוונים, אשר מקורו בערוצים רבים. לצד סוגיות עסקיות ואתגרים עסקיים טהורים, איתור מידע רלוונטי מהווה נתבך חשוב בנושא התמודדות עם סוגיות אבטחת מידע וסייבר (בהתאם לבאז הקיים), כדוגמת איתור חריגות בהתנהגות משתמשים ולאז מחשבים, איתור APT ([Advanced Persistent Threat](#)) ו-AVP ([Advanced Volatile Threat](#)).

פתרונות אבטחת המידע המסורתיים, כדוגמת SIEM ([Security Information and Event Management](#)) מתקשים לספק לארגונים תמונה הוליסטית, וזאת כאשר תקורות הניהול של מוצרים אלו גדלות מעת לעת. אף פתרונות ה-Sandbox המסורתיים מתקשים להתמודד עם תקיפות APT ו-AVT, וזאת עקב התחכום הרב של התוקפים, מגבלות הקיימות בכמות ואיכות הבדיקות המתבצעות ב-Sandbox, וכי הבדיקה במסגרת ה-Sandbox מוגבלת בזמן במרבית המקרים, דבר הנובע בין השאר מאילוצים עסקיים שונים.

אחד הפתרונות המובילים כיום לשם התמודדות עם הסוגיות אשר צוינו קודם לעיל הינו שילוב יכולות AI (Artificial Intelligence) - בינה מלאכותית - במוצרי אבטחת מידע וסייבר. בהתאם לכך, חלקו הראשון של המאמר יתמקד במספר תתי תחומי מחקר (בספרות המחקר ישנו שימוש במושגים שכיחים נוספים, כדוגמת: Architectures, Frameworks, Models, Domains) שכיחים למושג AI: Machine Learning (ML), Deep Machine Learning (DML) ושימוש ב-Biological Computation (ביולוגיה חישובית \ ביואינפורמטיקה). חלקו השני של המאמר יתמקד במספר מימושים עכשוויים של יכולות ML במוצרי אבטחת מידע וסייבר שכיחים, כדוגמת מערכת לזיהוי Fraud.



לקורא שאינו בקיא בתחום ה-AI אני מצ"ב הגדרה (מאוד) מופשטת ל-AI:

"A.I. is the study of how to make computers do things at which, at the moment, people are better.", Elaine Rich

או בעברית: תחום מחקר זה עוסק בדרכים אשר יאפשרו למחשבים לבצע פעולות (דברים), אשר כיום בני אדם מבצעים באופן טוב יותר.

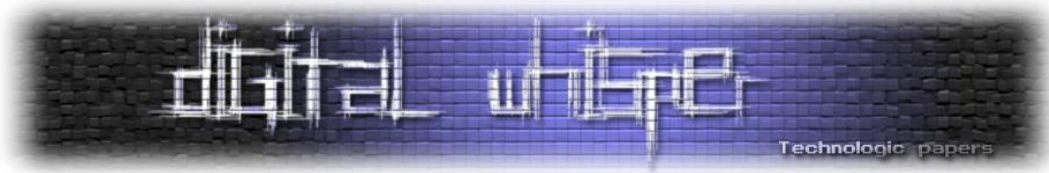
כמו כן, הגדרה ראשונית (אשר תורחב בהמשך המאמר), שכיחה ומופשטת (יחסית) ל-ML הינה:

"Machine Learning is Ability of a machine to improve its own performance through the use of a software that employs artificial intelligence techniques to mimic the ways by which humans seem to learn, such as repetition and experience. Machine Learning relates with the study, design and development of the algorithms that give computers the capability to learn without being explicitly programmed. It is the methodologies that are used to allow computers do intelligent task as human do; prediction, detection, classification, recognition, etc", Prachi A

ובעברית, למידת מכונה (ML) הינה יכולת רכיב מכונה לשפר את הביצועים של עצמה, וזאת באמצעות שימוש בתוכנה הכוללת יכולות בינה מלאכותית אשר מחקות את הדרך שבה בני האדם לומדים, כדוגמת ניסוי וחזרה. כמו כן, למידת מכונה מתייחסת למחקר, תכנון ופיתוח של אלגוריתמים המעניקים למחשב יכולת ללמוד, וזאת מבלי שהמחשב תוכנת באופן מפורש (מראש). תחום זה מציג מספר מתודולוגיות המאפשרות למחשב לבצע משימות אינטליגנטיות בדומה לאדם, כדוגמת: חיזוי, זיהוי, סיווג והכרה.

אקדים את המאוחר ואציין כי אין מאמר זה מתיימר להציג את תחום ה-AI ותתי התחומים השונים (כדוגמת תת התחום ML) לעומקם, וכי על מנת לפשט את המאמר בוצעו מספר הכללות, ובכלל זה יתכן כי יופיעו מספר אי דיוקים מסוימים בין המופיע בספרות המחקרית לבין הכתוב במאמר. מן הראוי אף לציין כי המידע המוצג במאמר זה הינו על קצה המזלג, וכי תחום ה-AI, ותתי תחומים השונים מהווים פרויקט חיים עבור ארגונים וגורמי מחקר. כמו כן, עקב מגבלות שונות, כדוגמת קניין רוחני (Intellectual Property), הגבלה באיכות ובכמות המידע המוגש לציבור ע"י יצרני פתרונות, המידע אשר מוצג במאמר זה מתבסס על מקורות מידע ציבוריים אשר מטבעם מוגבלים בתוכנם ואיכותם.

יש לציין כי תחום ה-AI ותתי התחומים השונים מתבססים על שימוש בכלים מתמטיים וסטטיסטיים מתקדמים (כדוגמת [The R Project for Statistical Computing](http://The R Project for Statistical Computing)), אשר מטבע הדברים אינו יכול להציגם לעומק.



## ההיסטוריה של ה-AI Artificial Intelligence בקליפת האגוז

מקובל לטעון כי בסיס ה-AI נובע משילוב משנתם של הפילוסופים היוונים סוקרטס ואריסטו (400 שנה לפני הספירה):

"Socrates: "I want to know what is characteristic of piety which makes all actions pious...that I may have it to turn to, and to use as a standard whereby to judge your actions and those of other men" (algorithm)

Aristotle: Try to formulate laws of rational part of the mind. Believed in another part, intuitive reason"

והתפתחות כלי חישוב קדמונים, כדוגמת [Abacus](#) (חשבוניה).

על בסיס עקרונות הפילוסופיה היוונית נבנו בשלהי תקופת הרנסאנס יסודות פילוסופיה נוספים, כדוגמת [Materialism](#), [Dualism](#), ושנים לאחר מכן בוצעה התפתחות נוספת, כדוגמת לידתן של [תורות ההחלטה](#) (Decision Theories), עקרונות הפסיכולוגיה (וביחוד תחום [הפסיכולוגיה הקוגניטיבית](#)) והמדע המודרניים (ובכלל זה נירולוגיה וביולוגיה), אשר בסופו של יום הביאו בשנת 1956 ללידתו של המושג Artificial Intelligence (AI) (בעברית, בינה מלאכותית) ב-Dartmouth Conference, וזאת בשילוב פיתוח תוכנת המחשב [Logic Theorist](#) - (ע"י החוקרים Alan Newell ו-Herbert Simon) אשר היוותה מימוש בסיסי וראשוני ל-AI, אשר מטרתו הייתה להוכיח משפטים.

אבן נוספת חשובה בהתפתחות ה-AI הינה פעילותו של מר. אלן מת'יוסון טיורינג, אשר "[הנחיל לעולם את מכונת טיורינג - מודל מופשט לאופן פעולתו של המחשב, ואת מבחן טיורינג - מבחן הבדק האם למכונה כלשהי יש בינה מלאכותית שלא תאפשר להבחין בינה לבין אדם.](#)"

כשלב אבולוציוני נוסף, נוצרו שפות תכנות ראשוניות לטובת AI, כדוגמת [LISP](#), וזאת במקביל להתפתחותם של תחומים חדשים, כדוגמת [NLP](#) (Natural Language Processing). למשנת חלק מהחוקרים בתחום, ה-NLP היווה שלב מכונן בהיסטוריה של ה-AI, וזאת משני טעמים עיקריים: ראשית כל, ה-NLP הציג הזדמנות עסקית ממשית אשר אפשרה גיוס תמיכה ציבורית וכלכלית - וזאת במקביל להוכחת יכולות פרקטיות (ביחוד בתחום העסקי). שנית, ה-NLP אפשר לראשונה למחשב להבין משמעות משפה טבעית (Natural Language Input). ולמותר לציין כי אחד המימושים היותר שכיחים וידועים בימינו ל-NLP הינה מערכת [Siri](#) (אשר שייכת למשפחת פתרונות ה-ASR Automatic Speech Recognition), אשר כוללת אף יכולות (ML) הנכללת במרבית מכשירי המובייל של חברת [Apple](#).

בפתרונות אבטחת מידע וסייבר Machine Learning **מבוא לשימוש** ביכולות

[www.DigitalWhisper.co.il](http://www.DigitalWhisper.co.il)





כמו כן, מן הראוי להזכיר את [VC Theory](#) אשר מהווה בסיס לתיאוריות ומודלים רבים בתחום ה-ML, אשר מטרתן להסביר את תהליך הלמידה מנקודת המבט הסטטיסטית. דוגמא לתיאוריה אשר נולדה בעקבות כך הינה [PAC learning](#) (Probably Approximately Correct Learning), שפותחה לשם מתן אפשרות לבחירת פונקציה אופטימלית לעיבוד קלט, כך שיחס השגיאה להכללת מידע בקלט ([Generalization Error](#)) יהיה בעל ערך נמוך ככל הניתן.

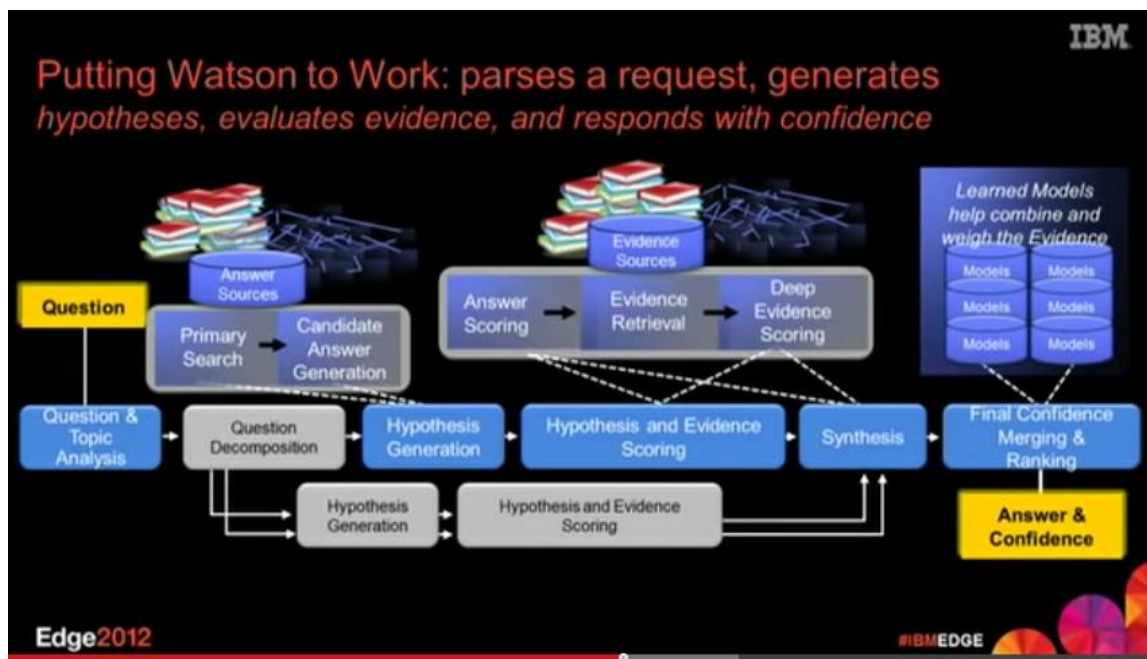
ובשולי הדברים, הועלו טענות כי [DARPA](#) (Defense Advanced Research Projects Agency), אשר מהווה מזה שנים רבות את אחד הארגונים המובילים את תחום ה-AI (וה-ML) בעולם, ניסתה לפתח בשלהי שנות ה-80 של ה-20 "מחשב חכם", וזאת בדומה למחשב העל [SkyNet](#) (מהסרט [The Terminator](#), ובעברית: שליחות קטלנית). בהתאם לכך ניתן ללמוד כי ה-AI ותתי התחומים השונים זכו להכרה כתחומי מחקרי אשר יש ביכולתם לסייע למדינות בהשגת עליונות צבאית על מדינות אחרות.

אחד הפרויקטים המובילים כיום בתחום ה-ML הינו פרויקט PPAML (Programming for Advancing Machine Learning) המנוהל ע"י DARPA, המשתמש ב-[Probabilistic Programming](#) (תכנות הסתברותי), אשר אחת המטרות העיקריות הינה להקטין את חוסר הוודאות ביחס להשערה מסוימת) וזאת על מנת להנגיש את האפשרות לכתיבת אפליקציות מסוג Machine Learning (ML) (המושג יזכה להרחבה בהמשך המאמר), לשפר את יעילות הפתרון ותהליך הפיתוח, יצירתם של יישומים חדשים אשר אינם קיימים כיום. כמו כן, פרויקט ה-PPAML שם דגש על נושא קיצור אורך הקוד של מודולי ה-ML, קיצור זמן הפיתוח, בניית מודולים עשירים ואפקטיביים יותר, ומענה לבעיית ה-[Overfitting](#), אשר מהווה את עקב האכילס של מרבית פתרונות ה-AI כיום. פרויקט מוביל נוסף הינו פרויקט [MICrONS](#) (Machine Intelligence from Cortical Networks) המנוהל ע"י IARPA (Intelligence Advanced Research Projects Activity), ומטרתו לבצע Reverse Engineering לפעילות מוחית, וזאת לשם איתור ובניית אלגוריתמים אשר יוכלו לדמות את תהליך החשיבה של אדם בעת פתירת בעיות. ומן הראוי אף להתייחס לפרויקט [ATLAS](#) המנוהל ע"י DARPA, ומטרתו לבנות רובוט אשר באפשרותו להסתגל לתנאי שטח מגוונים ולא מוכרים, בדומה לאדם. ברי כי מדובר בפרויקט אשר מטרתו ליצור חייל ביוני-אינטליגנטי, אשר מסוגל לפעול באופן אוטונומי.

פרויקט דגל נוסף בתחום ה-ML הינה מערכת המחשב "Watson" (ווטסון) של חברת [IBM](#), אשר משמשת כבר כיום לביצוע תהליכי אבחון מחלה וניהול הקשר עם החולה, וזאת בהתאם לעקרונות Evidence Based Medicine (רפואה מבוססת ראיות מדעיות). מערכת המחשב "Watson" (ווטסון) משתמשת ביכולות שלה לשם ניתוח מידע רב ממקורות מגוונים (Big Data), ובאמצעות קורלציה למידע אשר התקבל מפרופיל החולה, תכני שיחה עם החולה (אשר מקורם מממשק אדם-מכונה) וההיסטוריה הרפואית שלו, המערכת



מציעה לרופא המטפל אופציות אופטימליות לטיפול בחולי סרטן (לדוגמא). להלן תרשים המציג ברמת על את התהליכים העיקריים בפעילות מערכת המחשב "Watson" (ווטסון):



להרחבה בנושא ההיסטוריה של IA: [History of artificial intelligence](#)

## תחומי מחקר בתחום ה-Machine Learning (ML)

### מבוא

מספר תחומי המחקר בתחום ה-AI עולה כיום על כ-50, ומטבע הדברים לא ניתן להציגם במאמר מבוא מסוג זה. עם זאת, כמות תתי תחומי המחקר נמצאת בגדילה תמידית, דבר הממחיש את העניין הגובר של ארגונים בכלים מתקדמים אשר יוכלו לספק להם מענה לאתגרים שונים, אשר עד לתקופה האחרונה נחשבו כבלתי אפשריים.

ישנן שתי גישות שכיחות העומדות בבסיס יצירת ML, ובהתאם לכל גישה נגזר תחום המחקר הרלוונטי:

א. **Subsymbolic AI (חיקוי מבנה המוח האנושי)** - " באמצעות ייצוגים וירטואליים (כלומר, לא פיזיים) של רשתות נוירונים במחשב, ניתן לחקות את הפעולה של מספר תאי עצב במוח; אולם המספר האדיר של נוירונים במוח והמוגבלות בארכיטקטורה של המחשבים כיום מקשים מאוד על התפתחות גישה זאת."

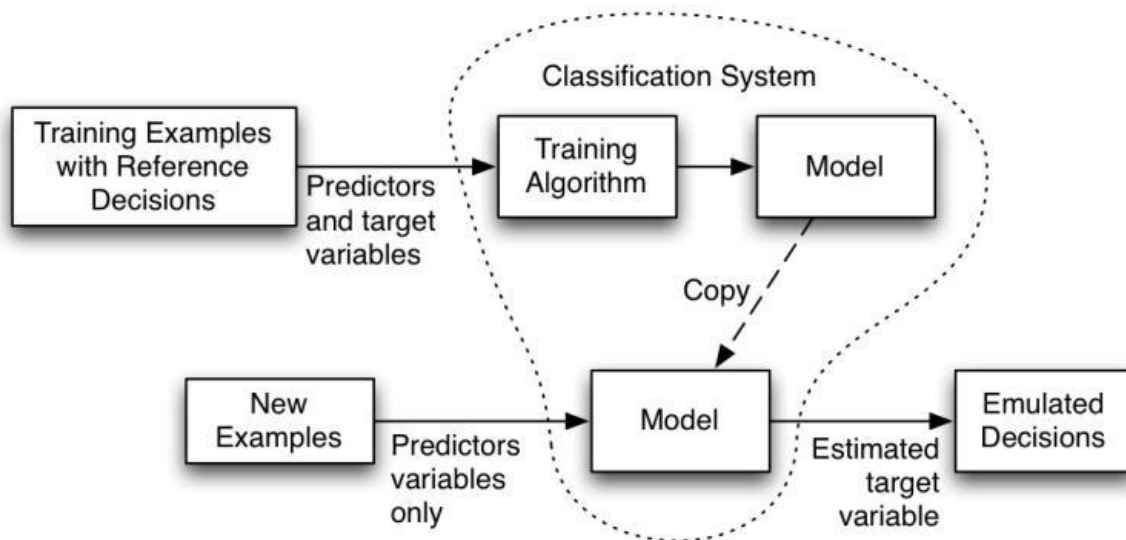
בפתרונות אבטחת מידע וסייבר Machine Learning **מבוא לשימוש** ביכולות

[www.DigitalWhisper.co.il](http://www.DigitalWhisper.co.il)

ב. **Symbolic AI (כתיבת תוכנות המחקות את ההתנהגות של המוח)** - "בניגוד לגישה הקודמת, בגישה זאת אין כל ניסיון לבנות מערכת הדומה למבנה של המוח. כאן המתכנתים משתמשים בדרך שנוחה להם, כדי להשיג את המטרה. כאמור, גם גישה זאת רחוקה מלהשיג את התוצאות המבוקשות. בכל מקרה, מקובל על המדענים שעל מנת להגיע למערכות מורכבות באמת, בעלות ידע מקיף על העולם סביבן, צריך לתת להן כלים ללמוד בעצמן, ולא לנסות לספק להן את המידע מראש. כך המכונה תלמד דברים הנחוצים לה, שהמתכנתים לא חשבו עליהם."

ניתן לחלק את שיטות הלמידה ב-ML לשלוש משפחות עיקריות (בפועל ישנן יותר משלוש משפחות), כאשר הבחירה בכל משפחה מושפעת בעיקר מסוג ה-Data (מידע) הזמין ברשותנו:

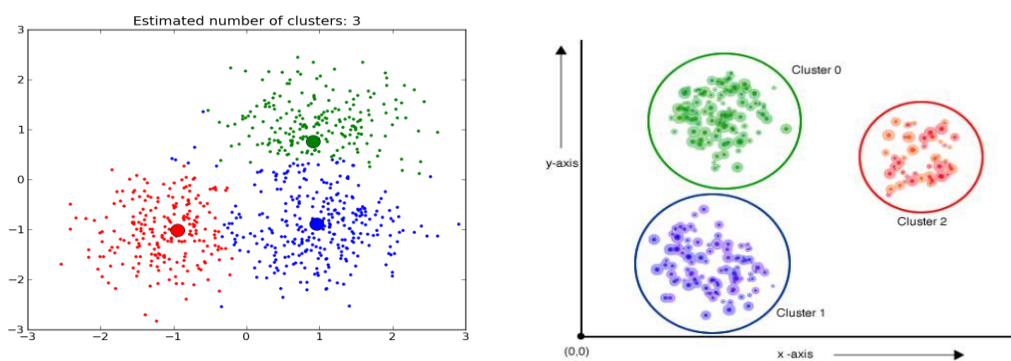
א. **Supervised Learning (למידה מפקחת)** - ישנו "מורה" שיודע את התשובה הרצויה לכל דוגמא (כולא). כך לדוגמא, ניתן לכייל את המערכת שתדע לזהות צבעים (Label המייצג קטגוריה/מאפיין/תכונה) של פירות. בהתאם לצבע של הפרי המערכת תוכל להסיק מהו הפרי המוצג בפניה. מקובל לממש את שיטת לימוד זו ע"י שימוש באלגוריתמים מסוג Classification (סיווג). אלגוריתמים ידועים המתבססים על שיטת לימוד זו: Bayesian Statistics, Kernel Estimators, Artificial Neural Network.



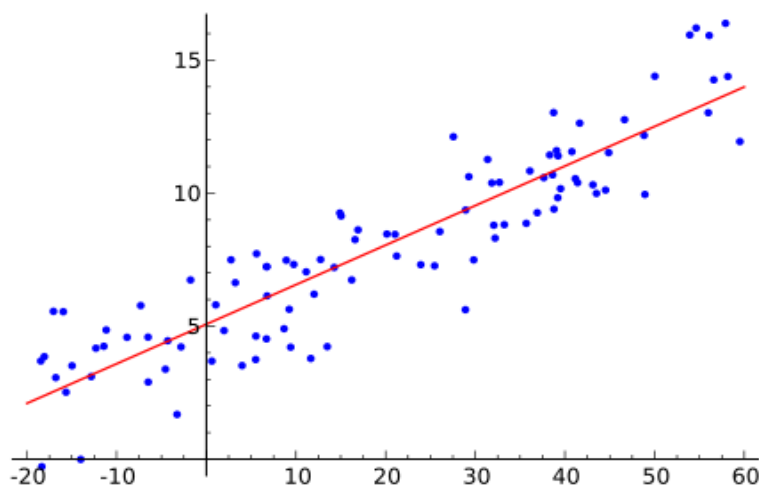
ב. **Unsupervised Learning (למידה לא מפקחת)** - אין "מורה" אשר יודע את התשובה הרצויה, ויש להשתמש בשלל "תרגילים מתמטיים"<sup>1</sup> על מנת לגלות את הקשר האפשרי (או העדר הקשר) הקיים בין הדוגמאות המוצגות בפני במערכת. רוצה לומר, ב-Unsupervised Learning המטרה היא לאפיין

<sup>1</sup> Fuzzy Logic (לוגיקה מעורפלת) - "שם כללי לתורות לוגיות המנסות להחיל את עקרונות החשיבה הרציונלית על תחומים שבהם נראה כי שני חוקי היסוד של הלוגיקה הקלאסית אינם מתאימים. בעיקר מדובר על תחומים שבהם יש צורך להתבסס על הערכות סובייקטיביות או רב-משמעיות"

את החוקיות הסטטיסטית של עולם הקלטים, וזאת לאור העובדה כי אין לנו Label (מייצג קטגוריה/מאפיין/תכונה) מוגדר מראש ל-Data. מקובל לממש את שיטת לימוד זו ע"י שימוש באלגוריתמים מסוג Clustering (אשכול) \ איתור "קבוצות טבעיות" (אלגוריתמים ידועים המתבססים על שיטת לימוד זו: k-means, mixture models, hierarchical clustering, Singular Value Decomposition). חשוב לציין כי אין לנו מקבלים תשובות של כן/לא, אלא לנו מקבלים השערות סטטיסטיות בעלות יכולת חיזוי (דיוק) כזו או אחרת. כך לדוגמה, אם נפזר תוצאות של דגימות על  $X, Y$  בגרף נתון, נוכל לזהות משפחות של Vectors (נקודות) בעלות שייכות, ובכך לייחד אותן ביחס לאחרות.

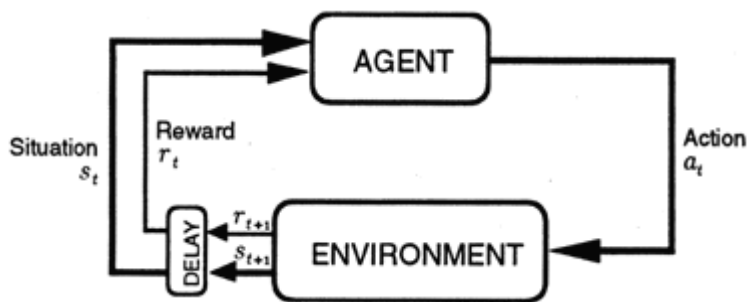


כמו כן, סוג נוסף של אלגוריתמים השכיח לראות במקרים מסוג אלו הינם Regression Algorithms (אלגוריתמי רגרסיה), המאפשרים יכולת זיהוי של קשר אפשרי (מתאם  $r$ ) בין משתנים, וזאת ע"י שימוש בגרף לינארי אשר יאפשר לנו לבנות קו אשר יגרום למינימום סטיות.



ג. **Semi-Supervised Learning (למידה מפוקחת למחצה)** - שיטת לימוד זו מהווה הכלאה בין שיטת הלימוד Supervised Learning, לשיטת לימוד Unsupervised Learning. יחוד השיטה הינו שימוש בדוגמאות Data אשר מרביתן ללא Label ייחודי. הסיבה לשימוש בשיטת לימוד זו הינה יכולת להגעה לדרגת דיוק גבוהה בתרחישים מסוימים, מאשר בשיטות החלופיות.

ד. **Reinforcement Learning (למידת חיזוק)** - בשיטת למידה זו ישנו "מורה", אך הוא אינו מספק תשובות של כן/לא, אלא הוא מספק משוב חיובי או שלילי (שיטת המקל והגזר \ "ניסוי וטעיה"). אחד המודלים השכיחים למימוש שיטת למידה זו הינו [תהליך החלטה מרקובי<sup>2</sup> \(Markov Decision Process\)](#), במקרה זה ניתן להפעיל בהתאם אלגוריתמים כגון: Q-Learning. כמו כן, ניתן לראות בשיטת למידה זו מצבים של התקדמות ונסוגה לסירוגין, דבר הנובע מקבלת משוב, וניסיון של האלגוריתם לאתר את הדרך הנכונה ביותר.



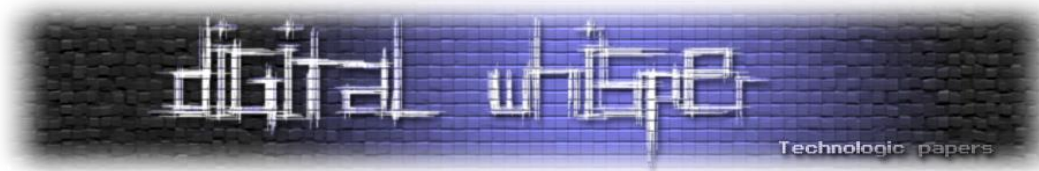
### Machine Learning (ML)

בנוסף, מקובל לסווג את האלגוריתמים בתחום ה-ML בהתאם לשתי גישות לימוד שכיחות:

**Lazy Learning (לימוד עצל)** - בשיטה לימוד זו המידע אינו זוכה לעיבוד מקדים (כדוגמת "אימון מקדים"), ורק לאחר הצגת בשאלה האלגוריתם מבצע תהליך חישוב, המציג בסוף התהליך תשובה (בהתאם למודל שנבנה). אחד מהיתרונות הבולטים של שיטה זו הינו זמן הלימוד הקצר, אך החסרון העיקרי של שיטת לימוד זו הינו זמן ארוך למתן תשובה (כדוגמת "חיזוי"). יתרון נוסף של שיטת לימוד זו הינו הצגת רמת דיוק גבוהה יותר ביחס לשיטה החלופית - Learning (לימוד נמרץ\חרוץ).

**Eager Learning (לימוד נמרץ\חרוץ)** - בשיטה זו מתבצע עיבוד מקדים למידע (כדוגמת "אימון מקדים"), ובהתאם למודל שנבנה (מראש), מתבצע תהליך חישוב, המציג בסוף התהליך תשובה (כדוגמת "חיזוי"). כאמור לעיל, יתרון שיטת לימוד זו הינו זמן קצר למתן תשובה.

<sup>2</sup> משפחת אלגוריתמים AI ידועה המתבססת על תהליך החלטה מרקובי הינה [DBM \(Deep Boltzmann Machines\)](#). עקב קוצר היריעה אין מאמר זה סוקר את משפחת אלגוריתמים זו.



המונחים ML ו-DML מהווים כיום שם נרדף אחד לשני, למרות שבמקור מדובר בדורות שונים של התפתחות. במקור ה-ML נועד להקנות למחשב יכולת ניתוח, הדרכה עצמית, התבוננות וניסיון. ברי, כי מדובר בחזון צנוע יחסית, אשר במהלך השנים התבגר, ובכך העלה את רף הציפיות.

תהליך ה-ML בד"כ<sup>3</sup> מורכב מארבע שלבים עיקריים:

Collection (איסוף) -> Extraction (חילוץ) -> Learning (למידה)<sup>4</sup> -> Classification (סיווג)

להלן מצ"ב תיאור של תהליך ML לדוגמא (ברמה המופשטת), אשר מטרתו לאתר קיומה של נזקה בקבצים:

1. Collection (איסוף) - בשלב זה מתבצע איסוף מידע (כדוגמת: קבצים, פקטות מידע) אשר מטרתו:
  - 1.1 ליצור מדגם מייצג והולם של פריטי המידע ברמה הכמותית (כמות מספרית), אשר יוכל להוות בסיס לביצוע פעולות סטטיסטיות.
  - 1.2 ליצור מדגם מייצג והולם של פריטי המידע ברמה האיכותית הכללית (כדוגמת: סוג קובץ), אשר יוכל להוות בסיס לביצוע פעולות סטטיסטיות.
  - 1.3 למנוע מצב שבו תהיה נטייה לאיסוף מידע בעל שכיחות גבוהה בלבד (כדוגמת קבצים בעלי מאפיין מסוים).

בהתאם לממצאים, הקבצים מסווגים ברמה הכללית לשלוש קטגוריות, ובהתאם מועברים ל-Queue (תור) ייעודי:

- Known and Verified Valid (הקובץ התגלה כתקין)
  - Known and Verified Malicious (הקובץ התגלה כמכיל נזקה)
  - Unknown (אין ידיעה האם הקובץ תקין או לא)
2. Extraction (חילוץ) - בשלב זה מיוצא מהקובץ מידע מסוג Uniquely Atomic Characteristics (מידע מאפיין אטומי \ תכונה אטומית, כדוגמת PE<sup>5</sup> file size). התוצאה של שלב האיסוף הינה יצירת File Genome (גנום מייצג לקובץ), אשר בדומה ל-DNA (Deoxyribonucleic Acid) של גוף האדם אשר מאפשר זיהוי ייחודי, ובכך הוא מאפשר שימוש במודלים מתמטיים אשר מאפשרים חיזוי של מאפיינים של קבצים דומים.

---

<sup>3</sup> שנים מודלים רבים בתחום אשר כוללים אף שימוש ב-Two-Stage Machine Learning (ML) (למידת מכונה דו שלבית).  
<sup>4</sup> לעיתים שלב זה כולל אף שלב Training (תרגול) אשר מטרתו לספק לאלגוריתם יכולת מראש לאתר מידע (כדוגמת מאפיינים/תכונות).  
<sup>5</sup> קובץ PE (בד"כ קובץ מסוג קובץ הרצה \ Executable) מורכב מפתיח (Header) PE, המכיל רשימה של ערכי רישום בספרית הנתונים \ מפה של ערכי רישום הספריות, ומספר החלקים המוגדרים לאחר פתיח (Header) ה-PE.

בפתרונות אבטחת מידע וסייבר Machine Learning מבוא לשימוש ביכולות

[www.DigitalWhisper.co.il](http://www.DigitalWhisper.co.il)



3. Learning (למידה) - בשלב זה המידע אשר נאסף בשלב הקודם עובר תהליך נרמול, ולאחריו המידע עובר המרה לערכים מתמטיים, אשר יאפשרו ביצוע חישובים סטטיסטיים עליהם, ובכלל זה בניית מודל סטטיסטי מתאים (אלגוריתם)<sup>6</sup>. ראוי לציין כי בשלב זה מתבצעים מספר שלבי "ניקוי" (כדוגמת - [Matrix Vectorization](#) [טרנספורמציה לינארית אשר ממירה את המטריצה לוקטור עמודה](#)), אשר מטרתם להקטין את ההסתברות לטעויות, ובכך לשפר את יכולת הדיוק. למותר לציין כי שלב זה מהווה שלב מהותי, ומרבית היצרנים אינם חושפים את האלגוריתמים בהם הם משתמשים בשלב זה.

4. Classification (סיווג) - בשלב זה האלגוריתם לסיווג מידע זמין למערכת, ובהתאם לכך מתבצע סיווג פרטני לקבצים מסוג Unknown (אין ידיעה האם הקובץ תקין או לא), וככלל התוצאה הינה כי הקבצים הנ"ל מסווגים (בד"כ) לשתי קבוצות:

- Known and Verified Valid (הקובץ התגלה כתקין)
- Known and Verified Malicious (הקובץ התגלה כמכיל נזקה)

כמו כן, המערכת מספקת למנהל המערכת חיווי מסוג "Confidence Score" (רמת הוודאות כי הסיווג אכן נכון), ובכך היא מאפשרת למנהל המערכת להפעיל את שיקול הדעת האישי על מנת לאשר או לדחות את ההמלצה. בד"כ מנהל המערכת יכול להגדיר מראש את ה-"Confidence Score" (רמת הוודאות כי הסיווג אכן נכון) אשר יאפשר להגדיר את הקובץ כ-Known and Verified Valid (הקובץ התגלה כתקין).

ראוי לציין כי ארבעת השלבים הנ"ל מתבצעים בזמן קצר יחסית, התלוי בפרמטרים הבאים בעיקר: כושר העיבוד של המערכת, כמות הקבצים, מספר המאפיינים (תכונות), וטיב האלגוריתם אשר נבנה בשלב ה-Learning (למידה). עם זאת, משך התהליך הקצר משמעותית מאשר ביצוע תהליך דומה ע"י אדם.

### Deep Machine Learning (DML)

אתר DeepLearning.net משתמש בהגדרה הבאה ל-DML:

"Deep Learning is a new area of Machine Learning research, which has been introduced with the objective of moving Machine Learning closer to one of its original goals: Artificial Intelligence."

ובעברית, DML מהווה תחום חדש בחקר למידת מכונה, והוא מציג בפנינו אפשרות להתקרב ליעוד האמיתי של למידת המכונה - שהינו השגת יכולת בינה מלאכותית.

תחת ה-DML נכללות מערכות מבוססת Cognitive Computing (מחשוב קוגניטיבי). מערכות אלו מאופיינות בכך שהן מנסות לרכוש מידע מה-Data המוזן, וזאת ע"י ביצוע Data Mining (כריית נתונים).

<sup>6</sup> פרקטית נבנים ולא יישנו שימוש במספר מודלים (אלגוריתמים) מוגדרים מראש, ובאמצעות אלגוריתם בחירה נבחר המודל (האלגוריתם) אשר לו את הסבירות הגבוהה להציג מענה אידיאלי לדרישות ה-Classification (סיווג).

בפתרונות אבטחת מידע וסייבר Machine Learning **מבוא לשימוש** ביכולות

[www.DigitalWhisper.co.il](http://www.DigitalWhisper.co.il)



תהליך ה-Data Mining כולל בחובו תהליך מיצוי ואיתור דפוסים, כך שעיבוד המידע מאפשר לאתר בעיות חדשות, וכן איתור מודולי פתרונות אפשריים.

### הבדלים מהותיים בין Machine Learning (ML) קלאסי ל-Deep Machine Learning (DML)

חברת [Skymind](#) מצאה לנכון לציין מספר הבדלים מהותיים בין ML ל-DML: בעוד למידת מכונה מסוגלת להגיע לרמות גבוהות של דיוק, יכולת זו מושגת בד"כ לאחר מאמץ ארוך, וישנו צורך בעיבוד ולימוד כמות גדולה של נתונים. למידה עמוקה יכולה להניב תוצאות משמעותיות יותר מדויקות מאשר למידת מכונה. עם זאת, עד לאחרונה הייתה בעיה במשך הזמן שלקח להכשרת רשתות למידה עמוקות. כיום כבר קיימת יכולת לבזר את תהליך הלמידה על מספר יחידות עיבוד (ביחוד מסוג [GPU](#)), דבר שהפך את הבעיה לזניחה. לפיכך ההבדל המהותי שנשאר כיום הינו ברמה טכנית: רשתות למידה עמוקה נבדלות מהרשתות העצביות (למידה עמוקה) בכמות השכבות החבויות; כלומר, מספר שכבות הצומת שדרכם נתונים מועברים בתהליך רב-שלבים של זיהוי תבניות. אם מדובר ביותר משלוש שכבות (הכוללים קלט ופלט), ניתן לסווג את הלמידה כלמידה עמוקה. אם מדובר במספר שכבות נמוך יותר, אזי מדובר על למידת מכונה. ברי כי מספר השכבות משפיע על המורכבות של התכונות שיכולות להיות מזהות.

### הקשר בין Machine Learning (ML) ל-Big Data

אחת השאלות השכיחות כיום הינה האם קיים קשר בין ML ל-Big Data, או לחילופין, האם לטובת ML ישנה חובה לשלב Big Data.

על מנת להבהיר את התשובה לשאלה זו, מן הראוי להיזכר במהות ה-Big Data, ולשם הנוחות אני מצ"ב את הגדרתה של חברת [גרטנר](#) ל-Big Data:

"Big data is high-volume, high-velocity and high-variety information assets that demand cost-effective, innovative forms of information processing for enhanced insight and decision making."

ובעברית, Big Data עוסק בנפחי מידע גדולים, קצבי עיבוד גבוהים של מידע, מגוון רחב של סוגי מידע - המחייבים שימוש בגישות יעילות וחדשניות לטובת עיבוד מידע, ובכך לקבל תהליכי הסקה וקבלת החלטות משופרים.

בהתאם לכך, ניתן להסיק כי ההחלטה האם לשלב Big Data כחלק מפתרון ה-ML תלויה במספר פרמטרים עיקריים:

א. סוג ה-Data הזמין לנו.

ב. כמות ה-Data הזמינה לנו.

---

בפתרונות אבטחת מידע וסייבר Machine Learning מבוא לשימוש ביכולות

[www.DigitalWhisper.co.il](http://www.DigitalWhisper.co.il)





ג. שיטת הלמידה האידיאלית לעבודה עם ה-Data הזמין לנו.

ד. האם ברצוננו לבחון Trends לאורך זמן ארוך.

ה. האם ניתן לבצע Training (אימון) מקדים לנתוני דמה, ובכך לייצר label (מייצג קטגוריה/מאפיין/תכונה).

לפיכך התשובה לשאלה זו מחייבת בחינה בהתאם לתרחיש הנידון, ובד"כ לא ניתן לתת תשובה חד ערכית לכל מקרה מראש. סייג לכך הינן מערכות CSI (Cyber Intelligence System), אשר מהותן מתבססת על שימוש ב-Big Data.

מן הראוי לציין כי מקובל לשייך את מערכות אבטחת המידע הכוללות שימוש ב-Big Data למשפחת פתרונות BDSA (Big Data Security Analytics).

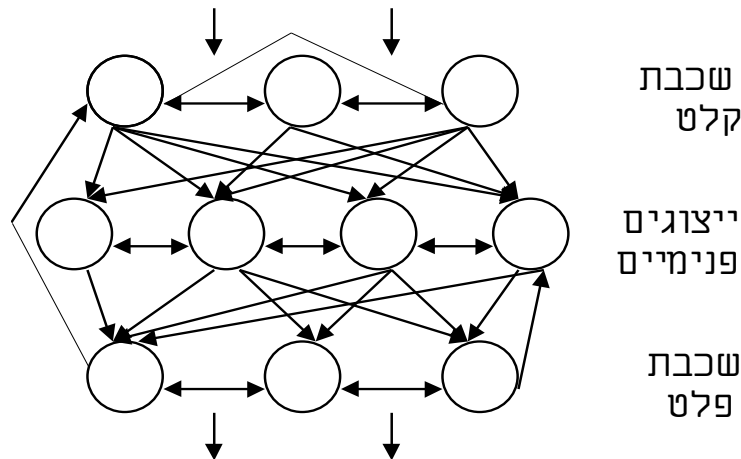
### Biological Computation (ביולוגיה חישובית \ ביואינפורמטיקה)

תחום מחקר נוסף אשר מסייע כיום לקידום המחקר של ה-ML הינו Biological Computation (ביולוגיה חישובית \ ביואינפורמטיקה). פרופ' אונגר רון מהפקולטה למדעי החיים, אוניברסיטת בר אילן מספק לנו את ההגדרה הבאה למודל זה:

"ביולוגיה חישובית עוסקת באינטראקציה בנושאים שונים בין מערכות ביולוגיות ותהליכים חישוביים. אנו עוסקים בעיקר בחקר המבנה של מולקולות ביולוגיות. המבנה של חלבונים למשל יכול לספק מידע חשוב להבנת הטיפקוד שלהם. השיטות הניסיוניות לקביעת מבנה של חלבונים דורשות זמן רב, כך שיש חשיבות רבה לפיתוח שיטות חישוביות שיאפשרו לנבא את מבנה החלבון. למרות שיש הוכחות רבות לכך שהאינפורמציה לתהליך הקיפול נמצאת כולה ברצף חומצות האמינו, הרי התהליך שבו רוכשת השרשרת הלינארית את המבנה שלה עדיין אינו ברור. מתוך התבוננות במבנים החלבוניים הידועים ומתוך השערות לגבי תהליך הקיפול אנו מנסים לפתח שיטות חישוביות שיאפשרו לנו לבצע במחשב "קיפול" של חלבונים למבנה הנכון. שיטת חישוב המתבססת על אלגוריתמים גנטיים שפיתחנו, נראית כבעלת פוטנציאל רחב לפתרון בעיות שונות בתחום זה."

לפיכך, ניתן לזהות שימוש הולך וגובר במודלים אורגניים-ביולוגיים בתחום ה-ML, ואף שכיח לראות כי מומחי אבטחת מידע וסייבר הינם אנשים מרקע שאינו קשור ישירות לעולם המחשוב.

לשם ההמחשה אציג באמצעות Wikipedia על קצה המזלג את מערכת "רשתות הניורונים" (חיקוי לפעילות מוחית, הכולל שילוב של ידע ותכנון דינמי של פעילות), אשר מהווה אבן בסיס לאלגוריתמי ML ו-DML רבים (כדוגמת אלגוריתמים המתבססים על ה-Reinforcement Learning):



"ישנם מודלים רבים של רשתות עצביות. המשותף לכולם הוא קיומן של יחידות עיבוד בדידות (המקבילה במודל לניורונים הביולוגיים) הקשורות ביניהן בקשרים (בדומה לקשרים הקיימים בין הניורונים הביולוגיים). הפרטים - מספר הניורונים, מספר הקשרים, מבנה הרשת (סידור בשכבות, מספר השכבות) - משתנים ממודל למודל. מקובל להשתמש בכלים מתחום האלגברה הליניארית כגון מטריצות ווקטורים על-מנת לייצג את הניורונים והקשרים ביניהם. הפעולה שמבצע כל נירון על הקלט שלו בדרך כלל מיוצגת על ידי פונקציה.

רשת ניורונים מאופיינת על ידי: אופן החיבור בין הניורונים ברשת; השיטה הקובעת את משקלי החיבורים בין הניורונים; פונקציית האקטיבציה. הפסקאות הבאות מסבירות כל מאפיין.

רשתות ניורונים מורכבות ממספר רב של יחידות עיבוד פשוטות הנקראות ניורונים, אשר מחוברות באופן היררכי ומובנות בשכבות. השכבה הראשונה נועדה לקלוט מידע לרשת, השכבה האמצעית ידועה כשכבה החבויה, ולבסוף השכבה האחרונה אשר נועדה להחזיר את המידע המעובד כפלט. הצמתים בכל שכבה מחוברים באופן מלא לצמתים בשכבות הסמוכות באמצעות חיבור ישיר בין הניורונים, כאשר לכל קשר קיים משקל מסוים. המשקל בכל קשר קובע עד כמה רלוונטי המידע שעובר דרכו, והאם על הרשת להשתמש בו על מנת לפתור את הבעיה. כל צומת בשכבת הקלט (השכבה הראשונה) מייצג תכונה שונה מהמבנה, ושכבת הפלטים מייצגת את הפתרון של הבעיה. בשכבה האמצעית והחיצונית קיימים "ערכי סף" הניתנים לכיול במערכת ממוחשבת, וקובעים את חשיבות הקשרים השונים.

תהליך הלמידה מתבצע על ידי "תגמול" ו"ענישה" של קשרים שונים ועל ידי חשיפת רשת הניורונים לדוגמאות רבות. "תגמול" ו"ענישה" של הקשרים מתבצע על ידי שינוי המשקל של אותו הקשר, כך שכל

קשר ש"מתוגמל" משקלו יגדל וכל קשר ש"נענש" משקלו ירד, כמובן תהליך זה משפיע במידה רבה על תהליך העיבוד ברשת הניורונים ולקבלת תוצאה שונה. כל נירון בשכבה הראשונה מתעדכן בנתון רלוונטי הנקרא קלט, ולאחר תהליך עיבוד בשכבות הביניים כל נירון בשכבה האחרונה מיצר פלט הנקרא פתרון. כל אחד מן הניורונים יכול להשפיע במידה מסוימת על המידע או על תהליך העיבוד שיתבצע בתא אחר בשל הקשרים הקיימים ברשת. פעולת החשיבה נעשית על ידי הזנת נתוני קלט לניורונים שבשכבה העליונה, והעברת הנתונים בין הניורונים ובמורד השכבות, עד שנוצר מערך של נתוני פלט בניורונים שבשכבה התחתונה. נתוני הפלט מהווים מערך של פתרונות הרשת לנתוני הקלט. מדוגמה לדוגמה מתעדכנים "ערכי הסף" של הניורונים שבמערכת, כך שהפלט יהיה אופטימאלי. הערכים המתמטיים של ערכי הסף יכולו, כך שכל קשר שצריך היה לאשרו יאושר, וכל קשר שצריך היה לדחותו יידחה. מעתה ואילך, אפשר לצפות שהרשת תקבל החלטות לאשר או לדחות באופן שמבטא בצורה מלאה את הניסיון הנצבר בתהליך הלמידה. יכולתה של רשת הניורונים מושתתת על יכולתה לספק דיוק מקסימלי לכל פונקציה קיימת כאשר דיוק זה יכול להתבצע על ידי הגברת מורכבות הרשת והגדלת מספר השכבות החבויות."

[מקור: [רשת עצבית מלאכותית](http://he.wikipedia.org/wiki/רשת_עצבית_מלאכותית)]

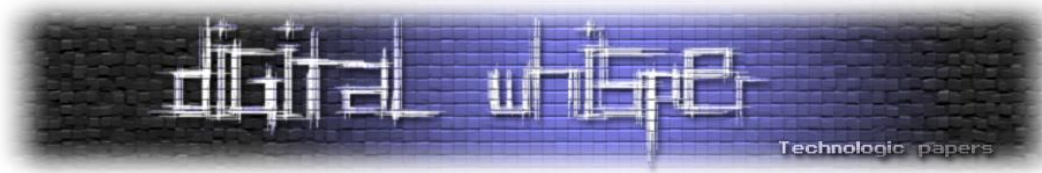
ועל מנת לחדד את הגדרת "רשתות ניורונים", אשתמש בהגדרתו של ד"ר אורן שריקי, הפקולטה למדעי המחשב, אוניברסיטת בן גוריון: "מערכות המורכבות ממספר רב של מעבדים פשוטים, הקשורים הדדית בקשירות גבוהה ופועלים במקביל. המידע נרכש דרך תהליך למידה ומאוחסן בקשרים."

אחת ממשפחות האלגוריתמים הראשונות המבוססות על עקרונות "רשתות הניורונים" הינה ANN (Artificial Neuron Network). רוצה לומר, ANN מהווה משפחה של אלגוריתמים המבוססים על מודל חישובי מבוסס על המבנה ופונקציות של רשתות עצביות ביולוגיות. מידע הזורם ברשת משפיע על המבנה של ANN, כאשר יש לזכור כי מדובר ברשת עצבית משתנה \ לומדת, וזאת בהתאם לקלטים והפלטאים. משפחת האלגוריתמים אשר "החליפה" את ANN הינה Deep Learning (למידה עמוקה) אשר מאפשרת התמודדות טובה יותר עם כמות נתונים רבה יותר, תוך שיפור הביצועים וזמני החישוב.

לסיום חלק זה, מצ"ב רשימה (חלקית אך מייצגת יחסית) של משפחות האלגוריתמים בתחום ה-ML:

Regression, Instance-based Methods, Regularization Methods, Decision Tree Learning, Bayesian, Kernel Methods, Clustering Methods, Association Rule Learning, Artificial Neural Networks, Deep Learning, Dimensionality Reduction, Ensemble Methods

לשם הרחבת היריעה בנושא משפחות האלגוריתמים הנ"ל ניתן לעיין ב: [A Tour of Machine Learning Algorithms](#).



## שימוש ביכולות Artificial Intelligence (AI) (עם התמקדות ב-ML) בפתרונות אבטחת

### מידע וסייבר

השימוש ב-AI מאפשר מענה למספר דילמות ובעיות בתחום אבטחת המידע<sup>7</sup>, כדוגמת:

1. ביזור תהליך קבלת החלטות והסקה.
  2. מתן מענה לבעיית חישוב רב-משתתפים בטוח, ובכלל זה מניעת אפשרות של צד לחשוף מידע של צד אחר באמצעות שימוש ב"סוד" מזויף (דבר המאפשר להתגבר על חולשות אינהרנטיות באלגוריתמים כדוגמת [Diffie-Hellman key exchange](#)).
  3. ביצוע כריית נתונים (Data Mining) תוך שמירה על פרטיות (Privacy).
  4. ביצוע מתקפות מסוג [Adaptive side-channel](#) (התקפה קריפטוגרפית המנצלת מידע שמושג מאופן היישום הפיזי או השימוש של מערכת ההצפנה, ולא באמצעות Brute Force כנגד האלגוריתם עליה היא מבוססת או [קריפטואנליזה](#) תאורטית של אלגוריתם ההצפנה).
  5. תכנון, בחינה ומעקף של מנגנוני הגנה כדוגמת CAPTCHAs.
  6. ביצוע Scoring לאובייקט (כדוגמת מחשב, משתמש) בנושא רמת ביטחון (Trust Level) ומוניטין (Reputation).
  7. איתור פגיעויות אבטחה (Vulnerabilities) באמצעות ביצוע בדיקות מסוג Intelligent Probing (רכישה סלקטיבית של נתונים על בסיס פרמטרים של מידע ואותות אשר נאספו מהרשת וממשקים, תוך מתן מענה למקרים של חוסר במידע), אשר מטרתן לגרום לצד המותקף לחשוף נקודות כשל אשר התוקף יוכל להסתייע בהן לשם ביצוע התקיפה<sup>8</sup>. דוגמה למימוש תקיפה מסוג זו הינה תקיפת [Fuzzing](#) המשתמשת ב-Malformed/Semi-malformed Data Injection לשם חשיפת נקודות כשל בצד המתוקף.
  8. ניהול מדיניות אבטחה מונחית תוכן (Content-driven Security Policy Management) הכוללת לעיתים קרובות ניהול הרשאות גישה ברמה פרטנית (Access List) המוגדרת מראש. כך לדוגמה, המערכת תוכל לזהות לפני מתן גישה לתוכן של מידע האם עובד פלוני צריך לקבל או לא לקבל הרשאה לגשת למידע, וזאת ללא צורך בהגדרה ידנית מראש.
  9. טכניקות ושיטות ליצירת ערכות הדרכה ובדיקה ללא מגע אדם.
  10. זיהוי התנהגות חריגה ברשת (Anomalous Behavior Detection), וזאת לטובת מתן מענה כנגד איומים שכיחים, כדוגמת מניעת הונאה (Fraud Prevention), פעילות APT ברשת הארגונית.
- בנוסף, השימוש ב-AI אמור להנחיל לקוד תוכנה (בשלב כזה או אחר) יכולת תיקון והשתפרות עצמית, ובכך יינתן מענה לפגיעויות אבטחת מידע וסייבר הנובעות מכשל אנושי ולא כשל קוד (כדוגמת ביצוע

<sup>7</sup> Source: <http://www.wikicfp.com/cfp/servlet/event.showcfp?eventid=38637&copyownerid=1398>

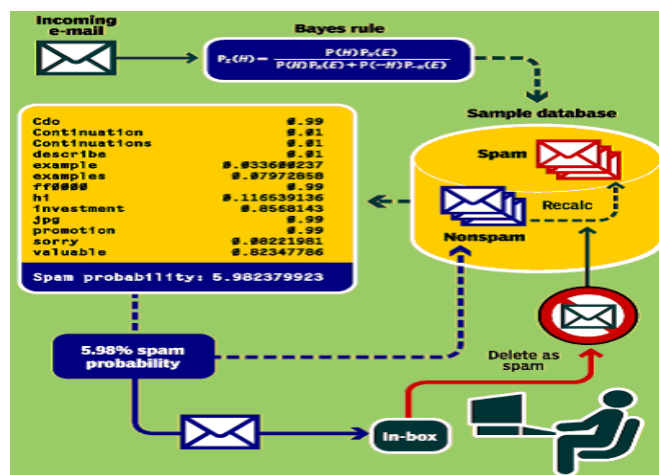
<sup>8</sup> תקיפה ידועה המשתמשת בטכניקה זו הינה Probe Response Attack, אשר מטרתה לאתר את מיקום הסנסורים אשר הארגון הטמיע לשם איתור זיהוי תקיפות ותוקפים.

בדיקות קלט ולא פלט לא מספק המאפשר (XSS). רוצה לומר, הקוד יכיל מעין מערכת חיסון ( Immune System) אשר תהיה באפשרותה להתמודד עם תקיפות ופגיעויות (כולל Zero Day Attack), ולתקן את הטעון תיקון בזמן אמת (Real Time) או לכל הפחות בזמן הקרוב לאמת (Near Real Time). כמו כן, שימוש ב-AI אמור להקטין את האפשרות לטעויות מסוג False Positive ו-False Negative, ובכך לשפר את הדיוק ומהימנות תהליכי אימות ותהליכי קבלת ההחלטות. ומעבר לכך, באמצעות יכולות AI ניתן יהיה לבצע תהליכים ארגונים-אבטחתיים, כדוגמת: Risk Assessment (הערכת סיכונים) ו-Risk Management (ניהול סיכונים) באופן ממוכן, ובכך לצמצם את התקורות הנדרשות לשם ביצוע התהליכים הנ"ל, וזאת במקביל להעלאת רף הדיוק וקיצור זמן הביצוע.

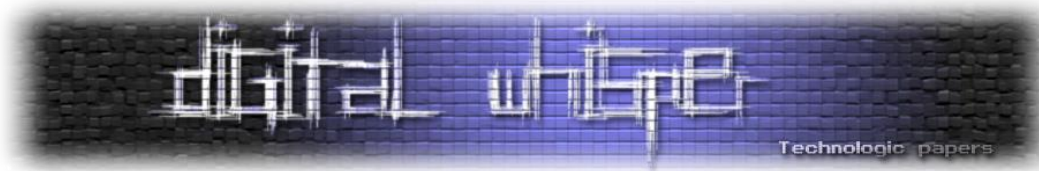
להלן סקירה ברמת-על של מספר פתרונות<sup>9</sup> ו-Use Cases (תרחישי שימוש) שכיחים בתחום אבטחת מידע וסייבר הכוללים שימוש ביכולות ML:

### א. פתרון Anti-spam Detection מבוסס Bayesian Filtering

אחד התחומים הראשונים בתחום אבטחת המידע אשר ניצל את יכולות ה-ML הינו תחום ה-Antispam. לפיכך שכיח לראות את פתרון זה מוצג במרבית שיעורי המבוא ל-ML באקדמיה. אחד האלגוריתמים הידועים הינו Bayesian Filtering אשר מאפשר לסווג את הדוא"ל לשתי ישויות עיקריות: Spam & Legitimate Emails, ובכך לאתר מעל 98% מהודעות ה-Spam המגיעות לארגון מהעולם. היתרון העיקרי של אלגוריתם Bayesian הינו היכולת שלו לזהות את הקשר בין התכנים הנכללים בדוא"ל (כדוגמת: מילים, מחרוזות תווים, HTML Tags, ) וזאת ע"י שימוש בתאוריה של היקש הסתברותי. בהתאם לתיאוריה, ניתן לבנות מסד נתונים אשר מסוגל להשוות בין תכני דוא"ל (וההקשר שלהם), העדפות משתמש, ובכך לאפשר איתור Spam, ובכלל זה השגת יכולת לחיזוי האם הודעת דוא"ל עתידית תסווג בסבירות גבוהה כהודעת Spam.



<sup>9</sup> למען הסר ספק, הפתרונות המוצגים במאמר זה הינם להמחשה בלבד, וכי אין באמור במאמר זה בכדי להוות ייעוץ ולא המלצה לפתרון כזה או אחר.

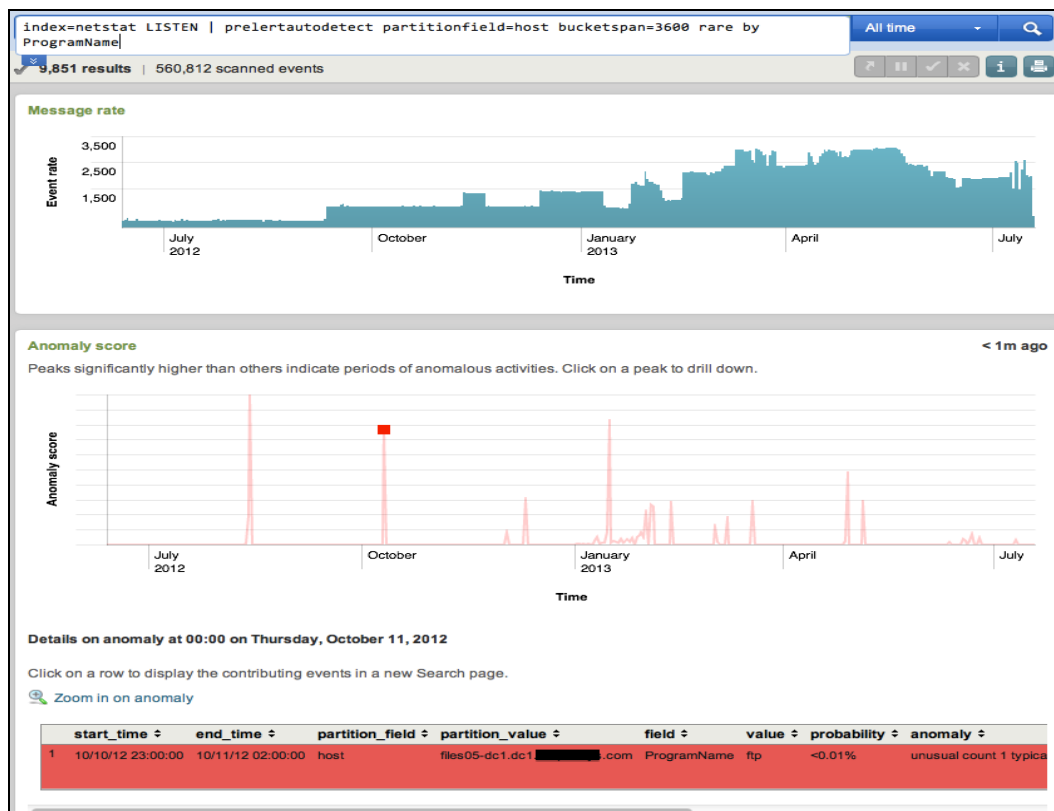


## ב. פתרון Behavioral Analytics & Anomaly Detection מבית חברת [Prelert](#):

חברת Prelert מציעה מספר פתרונות מעניינים, הכוללים שימוש ביכולות AI לטובת מתן מענה לתרחישי אבטחת מידע וסייבר שכיחים. אחד היתרונות הבולטים של החברה הינו ביצוע אינטגרציה אינטגרטיבית עם פתרונות [SIEM \(Security Information and event Management\)](#), כדוגמת [Splunk](#).

הפתרונות של חברת Prelert מתבססים על אלגוריתמי ML אוטומטיים (לטובת עיבוד Semi-Structured Data) מסוג: Clustering, Time Series Decomposition, Bayesian Distribution, Modeling, and Correlation analysis העובדים ב-Online, אשר יש ביכולתם לקבוע את דפוסי "ההתנהגות הנורמלית" (שם נרדף: פרופיל התנהגותי) על בסיס מספר רב של מקורות מידע, וכל זאת ללא צורך בהתערבות יד אדם (Zero Touch). חריגה מ"ההתנהגות הנורמלית" מאפשרת גילוי מוקדם של דפוסים אשר יכולים להעיד על ניסיון חדירה ו/או דליפה של מידע ארגוני (ובכך להפעיל תהליך Incident Response בארגון). כמו כן, ה-ML מאפשר Prediction (יכולת חיזוי) ובכך לצמצם את ה-False Positive Error Rate וה-Positive False Error Rate.

ב-Use Case הבא ניתן לראות כי המערכת מבצעת בחינה של טבלת ה-Connections של מערכת ההפעלה (Netstat) וזאת לשם איתור חריגות בהתנהגות התחנה לאורך תקופה:



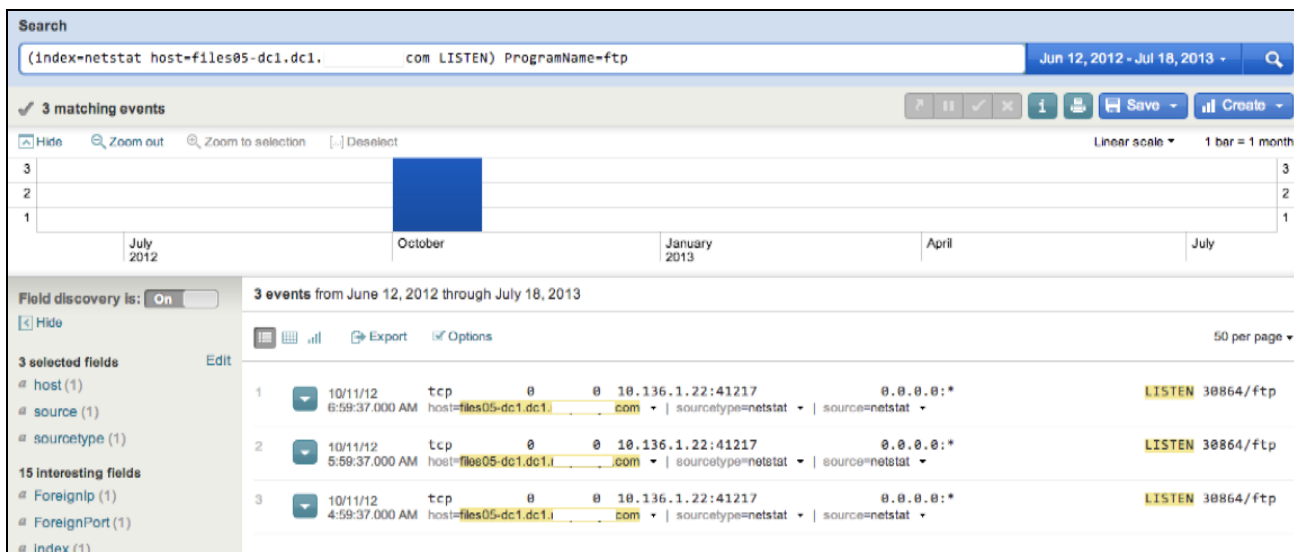
בפתרונות אבטחת מידע וסייבר Machine Learning מבוא לשימוש ביכולות

[www.DigitalWhisper.co.il](http://www.DigitalWhisper.co.il)





בהתאם לגרף ניתן לזהות כי בשנת 2012 (לדוגמא) אותר מקרה חריג, וע"י ביצוע ניתוח עומק:



אותרה התקשרות חריגה בפורט TCP 41217 עם שרת FTP "לא מוכר" (10.136.1.22). במאמר מוסגר, מן הראוי לציין כי על ידי שימוש בכלי Digital Forensic ברמת מערכת ההפעלה ניתן לבצע בחינה פרטנית לאפליקציה אשר יוזמת את ההתקשרות החריגה. כמו כן, אלגוריתם ה-ML יכול לבצע ניתוח של אנורמליות נוספת ברמת מערכת ההפעלה, כדוגמת Process, Running Applications.

לסיכום, Use Case זה מציג כי בניגוד לגישה המסורתית בה ישנו צורך לנתח לוגים ידנית, השימוש ב-ML מאפשר לחסוך משאבים ואף להגיע לרמת דיוק גבוהה יותר. למידע נוסף:

[Automated Anomaly Detection: A Look Under the Hood](#)

### ג. איתור חשבונות Skype המשמשים לטובת הונאה

חברת [Microsoft](#), בשיתוף אוניברסיטת [Tartu](#) פיתחה מנגנון אשר מטרתו לאתר חשבונות Skype המשמשים לטובת הונאה. ראוי לציין כי במקרה הנדון הונאה הוגדרה כמעשה של שימוש לרעה בפרטי תשלום (כדוגמת כרטיס אשראי), ביצוע Abuse לחשבונות של משתמשי Skype, ובכלל זה הפצת Spam באמצעות הפלטפורמה של Skype. בהתאם לפרסום משותף של הגורמים הנ"ל, לפתרון שפותח ומומש ישנה יכולת מוכחת לזיהוי כ-68% ממקרי ההתחזות וההונאה תוך כ-4 חודשים, וערך ה-False\Positive עומד על 5 אחוז. ברי כי אחת ממטרת המחקר העיקריות הייתה פיתוח יכולת ה-Automated Pattern Classification (סיווג דפוס באופן אוטומטי), ובכך לאפשר בידול בין חשבון "תקיין" (Normal) לחשבון "לא תקיין" (Fraudulent).

בפתרונות אבטחת מידע וסייבר Machine Learning מבוא לשימוש ביכולות

[www.DigitalWhisper.co.il](http://www.DigitalWhisper.co.il)



אקדים את המאוחר ואציין כי אני ממליץ בחום לקרוא את המאמר המלא העוסק בסוגיה זו, וזאת מכיוון שהמסמך כולל תיאור פרטני (ברמה היחסית) של מימוש אשר הוטמע בסביבת Production:

[Early Security Classification of Skype Users via Machine Learning](#)

להלן מצ"ב סקירה ברמת-על של התהליך: ה-Data הנאסף על משתמש ה-Skype נגזר ממספר מקורות מידע, והתרשים המצ"ב מציג סקירה לדוגמא של פריטי מידע, תוך ציון מקור המידע:

**Table 1: Sets of features (with activity logs in *italic*)**

<b>Profile set</b>	gender age country OS platform ...
<b>Skype product usage</b>	<i>connected days</i> <i>audio call days</i> <i>video call days</i> <i>chat days</i>
<b>Local social activity</b>	<i>additions by a user</i> <i>deletions by a user</i> <i>additions of a user</i> <i>deletions of a user</i> accept rate (%) degree
<b>Global social activity</b>	full contact graph

בהתאם לטבלה הנ"ל, ניתן ללמוד כי פריטי המידע חולקו לארבעה משפחות:

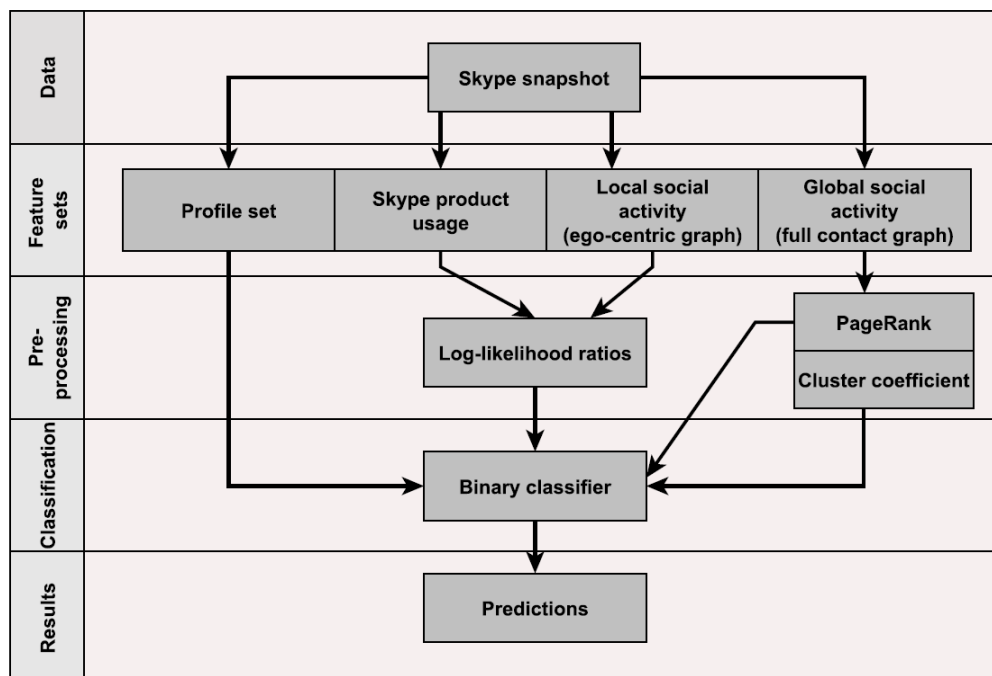
1. Static User Profile - מידע סטטי אשר מקורו בפרופיל המשתמש (גיל, וכו').
  2. Dynamic Product Usage - מידע פעילות דינמי (Activity Logs) של חשבון ה-Skype (הבדלי זמן בין שיחה אחת לאחרת, וכו').
  3. Local Social Behavior - אופי פעילות (מקומי) של בעל חשבון ה-Skype ברשת חברתית (הוספת/מחיקת "חברים", וכו'). שם נרדף הינו Egocentric Network (רשת אגוצנטרית).
  4. Global Social Features - נתונים גלובליים של פעילות בעל חשבון ה-Skype ברשת חברתית (כמות Likes לפרסומים בעל החשבון \ PageRank, וכו'). מידע זה נגזר מ-Full Social Graph (גרף חברתי מלא) אשר מהווה בסיס למערך הידע ב-Social Networks שכיחות.
- סוגיית ה-Graph Theory (תורת הגרפים) אינה נסקרת במאמר זה וזאת עקב קוצר היריעה והמורכבות הגבוהה (יחסית) של תיאוריה זו. פרק הביבליוגרפיה במאמר זה כולל הפנייה למקורות מידע להרחבת הדעת בנושא.

במהלך בניית הפתרון פותחו Labels המעידים על חשד להונאה, וכן הושם דגש לנושא אנונימיות של ה-Data ככל הניתן. הנחת המוצא הייתה כי הצלבת מקורות מידע (סינרגיה) תסייע לאיתור קיומה של הונאה, תוך הצעת דרגת False\Positive נמוכה. בנוסף, החוקרים יצאו מהנחת מוצא כי הצלבת מקורות המידע (סינרגיה) תאפשר להתגבר על חוסרים בפריטי מידע, תוך שמירה על רמת דיוק אופטימלית.

בהתאם לכך, החוקרים נדרשו לפתח מודל Classification (סיווג) מורכב, אשר יש ביכולתו להציג יכולת Classification אופטימלית לכל משפחה של פריטי מידע, וכן לייצר Classification (סיווג) ברמת על, אשר יש באפשרותו להציג משקלים יחסים ודינמיים (כאשר המשקל היחסי תלוי בטיב המידע אשר על בסיסו בוצע ה-Classification ברמת משפחת פריטי המידע) של Classification (סיווג) של כל משפחה של פריטי מידע.

התרשים הבא מציג את ה-Workflow (תהליך זרימה של התהליך) של תהליך ה-Classification, הכולל בחובו שימוש במגוון של אלגוריתמים, כדוגמת: א. Random Forest (אלגוריתם ממשפחת Classifiers Decision Tree) אשר מסייע בהפחתת השונות ע"י הקטנת המתאם בין ה-Trees, ב. PageRank & Local Clustering Co-Efficient לניתוח מידע על בקשות (או דחיית בקשות) בעת עבודה עם Contact List, ג. HMMs (Hidden Markov Models) אשר מאפשרים זיהוי של חוזק הקשר בין חשבוניות Skype, ד. Atandard Baum-Welch אשר מסייע בקביעת ערכי הסתברות מקסימליים ל-HMMs.

Figure 1: Entire workflow for the classification process.



<sup>10</sup> האלגוריתם שימש אף למשימות נוספות, כדוגמת זיהוי ביצוע פעולות Spam ב-Social Networks, ועל ידי כך הוא מאפשר מתן משב למערכת בעת קביעת ערך ה-Reputation לחשבון.

התוצאה הסופית של תהליך מורכב זה הינה סידרה של גרפים המציגים את הממצאים (תוך השוואה ביחס ל-Baseline ראשוני), ובכלל זה את ההשפעה היחסית של משפחת פריטי המידע על התוצאה הסופית:

Figure 2: Distribution of fraudulent users by their lifetime (of undetected activity) before using our approach and after eliminating those fraudsters caught by our approach



בהתאם לאמור לעיל, ניתן לראות כי הפתרון מציע יכולות ML אשר יש באפשרותם לזהות פעילות תקינה \ לא תקינה של חשבונות Skype, תוך צמצום למינימום של הצורך בהתערבות יד אדם בתהליך.

#### ד. Facial Recognition (זיהוי פנים) \ זיהוי ביומטרי מבוסס Behavioral Scientists (מדעי ההתנהגות)

זיהוי משתמשים ולקוחות אשר ניגשים למערכות המחשוב ברמת וודאות גבוהה היווה מאז ומתמיד אתגר לארגונים. כמו כן, כלי OSINT (Open source intelligence) מתקדמים אוספים מידע רב, דבר אשר מחייב ביצוע הצלבה ואימות בין מקורות רבים. לפיכך, פתרונות אימות ותחקור ביומטריים כוללים כיום שימוש נרחב ביכולות AI (ביחוד מתת תחום ML) לשם התמודדות עם כמויות המידע הגדלות והולכות, וזאת במקביל לשיפור רמת הדיוק (קורלציה) ואיתור הכוונה (Intention). כמו כן, ה-Internet of Everything (הדור הבא של IoT - Internet of Things, האינטרנט של הדברים) מציב אתגרים חדשים לארגונים ולחברה האנושית, אשר מחייבים בתורם פיתוח מודלים אוטומטיים וחדשים אשר ישפרו הן את חווית הלקוח והן את רמת אבטחת מידע ומוכנות הלקוח להתמודדות עם איומי סייבר.

נכון לזמן כתיבת מאמר זה, אחת היכולות המתקדמות בתחום זיהוי ביומטרי כוללת את היכולת לזהות חוסר עקביות בין הצגת המלל (הבעת מילים) של האדם, לבין הבעת הפנים שלו בזמן הדיבור (מושג שכיח בתחום הינו "Multimodal" - מגוון הרחב של פעולות פיזיות המפורשות על ידי מחשב). במקור, חוקרים מ-[Ohio State University](http://Ohio State University) גילו כי בני האדם מסוגלים להביע כ-20 הבעות פנים (כיום המספר עלה לכ-46), אשר מביעות את המצב הרגשי (כדוגמת שמחה) של האדם בזמן נתון. המחקר המקורי כלל פיתוח תוכנה לזיהוי פנים אשר השיגה יכולת של כ-96.9 אחוזים בזיהוי שישה רגשות בסיסיים, ו-76.9 אחוזים במקרה של רגשות המורכבים. רכיב תוכנה זה פותח על בסיס מחקרו של הפסיכולוג [Paul Elkman](#), אשר עסק בנושא איתור ומיפוי של תהליכי הבעת רגשות. המחקר הנ"ל היווה בסיס לפיתוח מערכות (Facial Action Coding) FAC, אשר יש ביכולתן אפשרות למפות ביטויים רגשיים בהתאם לפעילות הפיסית של פני הנבדק (כדוגמת פעילות שרירי הפנים ותנועות איברי הגוף, אשר ניתן להמירם לביטוי ריגשי מסוים).

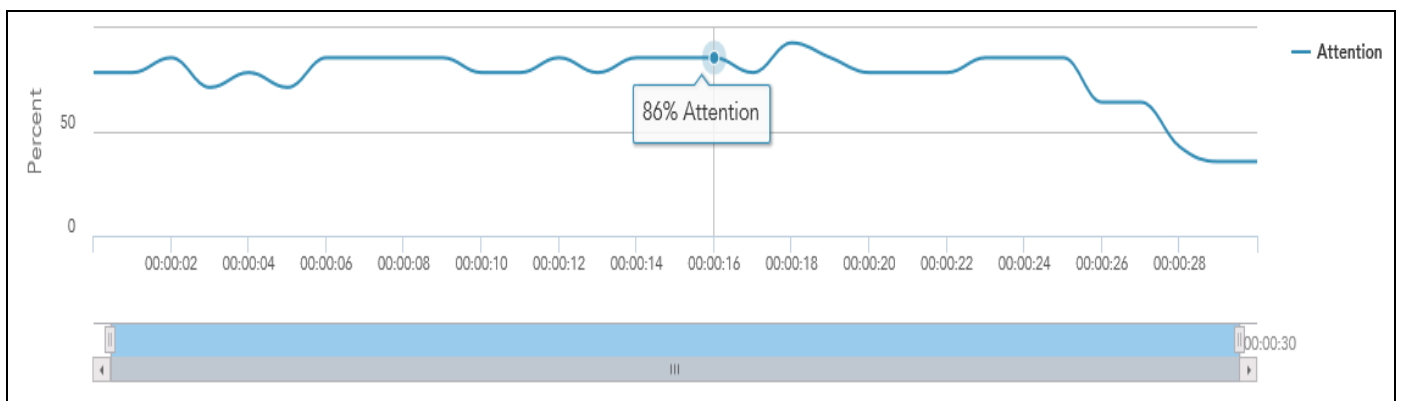
ובמאמר מוסגר, שימוש מעניין ליכולות מסוג אלו הינו איתור הכוונה ו/או רמת המוכנות של הצד שכנגד. כך לדוגמא, ניתוח נאום מצולם של פוליטיקאי יכול לאפשר לצד שכנגד לבדוק האם הפוליטיקאי אכן מתכוון לעמוד בהצהרותיו או שמא מדובר במס שפתיים בלבד. שימוש בשיטות ביומטריות מסוג אלו מהווה כלי אסטרטגי ([OSINT](#)) אשר יש ביכולתו להשפיע שדה המערכה.

חברת [Emotient](#) פיתחה רכיב תוכנה אשר מאפשר זיהוי וניתוח על בסיס הבעות פנים, ובכך הפתרון מהווה דור מתקדם של פתרונות ה-FAC. בהתאם לניתוח המתבצע על ידי שימוש באלגוריתמי ML שונים, ניתן לזהות ביטויים עיקריים ב-Real Time (זמן אמת) של רגש, כדוגמת רגשות חיוביים ושלייליים, שילוב של רגשות שונים, ומהם ניתן להסיק תבונות שונות. כמו כן, בהתאם לתורות נוירולוגיות-פסיכולוגיות שונות ניתן להסיק כי רגש מבוסס פעילות מוחית ועמוד שדרה באזורים מסוימים, ובהתאם לכך להשתמש בפתרון כמעין "גלאי שקר". כך לדוגמא, אמירה של ביטוי שקרי מפעילה אזורים שונים במוח ובעמוד השדרה, ובכך אופי התגובה כלפי חוץ יהיה שונה מאשר בעת אמירת ביטוי אמת. לטענת החברה היא הצליחה להשיג רמת דיוק של כ-85% בתנאי מעבדה, וכי ע"י ביצוע קורלציה בסיוע מנגנונים נוספים (כדוגמת רכיב המבוסס על זיהוי וניתוח קול) ניתן יהיה לשפר את רמת הדיוק באופן ניכר. מן הראוי לציין כי סוגיה זו מעלה לדיון את נושא - [Technology and the Right to Lie](#) (טכנולוגיה והזכות לשקר) אשר חורג ממסגרת מאמר זה, אך הוא מהווה אתגר מוסרי-חברתי לאנושות בפני עצמה.

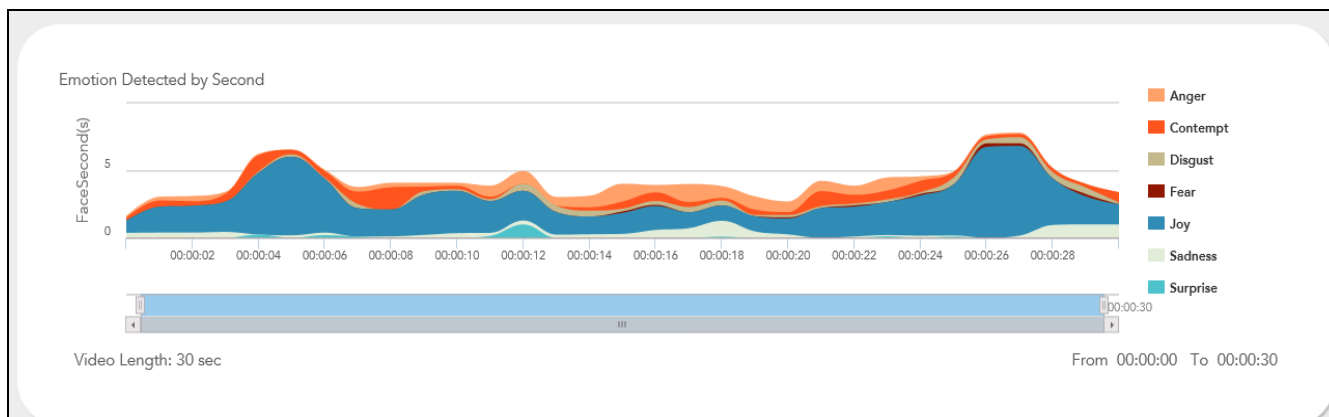
להלן מצ"ב Workflow (תהליך זרימה של התהליך) של תהליך הפיתוח של הפתרון האנליטי של חברת [Emotient](#), ובהתאם לפרסומי החברה יש באפשרותה לאתר מעל 22 תנועות ייחודיות של שרירי פנים:



כמו כן, מצ"ב תרשים לדוגמה אשר נוצר כחלק מדו"ח הפתרון האנליטי של חברת [Emotient](#), אשר מציג את רמת ה-Attention (תשומת הלב) של הנבדק ביחס לזמן:



התרשים הבא מציג את רמת ה-Emotional Engagement (מעורבות רגשית) של הנבדק ביחס לזמן:



לפיכך ניתן לראות כי הפתרון הנ"ל מאפשר זיהוי ביומטרי ברמת ודאות גבוהה יחסית, וזאת בנוסף למתן יכולת לקבלת תבונות נוספות על התנהגות הפרט, כדוגמת האם הוא דובר "אמת" או "שקר".

#### ה. **Transparent User Authentication (זיהוי משתמש באופן "שקוף") בעת גישה ל-Web**

בעשרות השנים האחרונות השימוש ב-Web (אינטרנט) גבר באופן ניכר, אך במקביל שיטות זיהוי המשתמשים הלקוחות לא השתנו באופן ניכר. כך לדוגמא, השימוש בסיסמאות שכיח אף באתרים פיננסיים רבים, ושימוש ביכולות זיהוי מתקדמות (כדוגמת אימות ביומטרי מסורתי), אינו ישים עקב מורכבות גבוהה, עלות גבוהה יחסית ובעיית תאימות הנובעת ממערכות Legacy (ישנות) ומהעדר תקנים משותפים ומקובלים בין היצרנים.

חברת [BioCatch](#) מציעה פתרון Transparent User Authentication (זיהוי משתמש באופן "שקוף") בעת גישה ל-Web, המבוסס על תהליך Invisible Challenge-Response (אתגר-תגובה "שקוף"), הכולל מימוש יכולות ML בצד שרת. מן הראוי לציין כי במקור שיטת זיהוי זו מבוססת על מחקרו של פרופ' [Nathan Clarke](#) מ-[CSCAN](#) (The Centre for Security, Communications and Network Research) אשר עסק ב-"Transparent User Authentication: Biometrics, RFID and Behavioural Profiling" (אימות משתמש באופן שקוף: ביומטרי, [RFID](#), פרופיל התנהגותי).

הפתרון מורכב משני מנגנונים עיקריים:

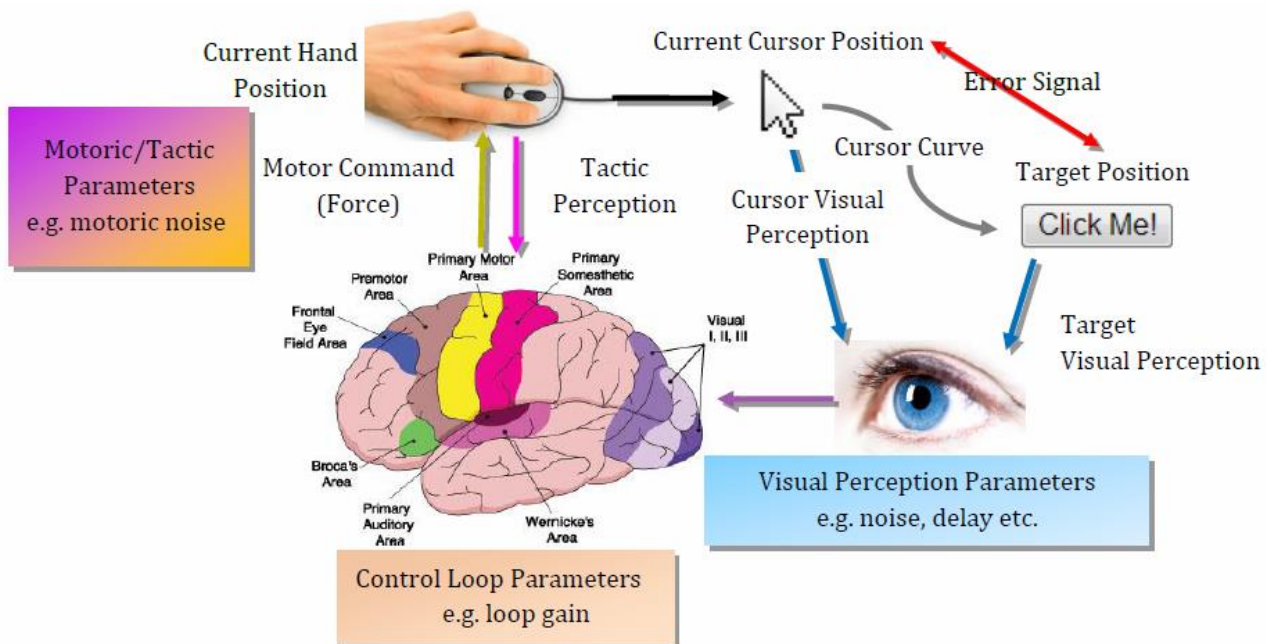
1. צד הלקוח - Challenge Inject (הזרקת "אתגר" רנדומלי) לצד הלקוח (דפדפן \ מובייל) על בסיס רנדומלי (לשם הקטנת הסבירות להצלחת מתקפת Replay, וכן לשם בחינת "חיות" צד הלקוח), וזאת במטרה לבחון את תגובת המשתמש הלא מודעת (כדוגמת רפלקסים ופעילות שרירית לא רצונית) במספר ממשקים: הפעלת עכבר, הפעלת מקלדת, הפעלת מסך מגע. העיקרון המנחה במקרה זה



הינו כי כל אדם מגיב בצורה שונה ל-Challenge Inject (הזרקת "אתגר"), וזאת עקב שוני בין הפרטים בתחומים הבאים:

1. תכונות קוגניטיביות, כדוגמת: תיאום עין-יד (קורדינציה), דפוסי התנהגות אפליקטיביים, העדפות שימוש, דפוסי אינטראקציה עם הממשק, ותגובות ייחודיות לתהליך ה-Invisible Challenge-Response (אתגר-תגובה "שקוף").
2. גורמים פסיולוגיים, כדוגמת: שימוש ביד ימין/שמאל, עוצמת הלחץ של האגודל, רמת הרעידות ביד, גודל היד, אופי השימוש בשרירים.
3. גורמים טכניים מסורתיים, כדוגמת: סוג המכשיר/אפליקציה (Device ID) עמו המשתמש מנסה לגשת למשאב, מיקום גיאוגרפי, ספק רשת האינטרנט.
2. צד שרת - שימוש בטכנולוגיית ML אשר בונה פרופיל ייחודי למשתמש, וזאת על סמך כ-400 פרמטרי פעילות אשר נאספו מצד הלקוח. בהתאם לפרופיל הייחודי אשר נבנה ישנה אפשרות לזהות הם האם המשתמש הוא אכן "אמיתי", האם ישנה פעילות רכיב עוין במכשיר המשתמש (כדוגמת סוס טרויאני המנסה לממש תקיפת [MiTM - Man in The Middle](#)) וכן האם מתבצע ניסיון Fraud (הונאה), כדוגמת ביצוע העברת כסף מחשבון לחשבון ע"י גורם עוין.

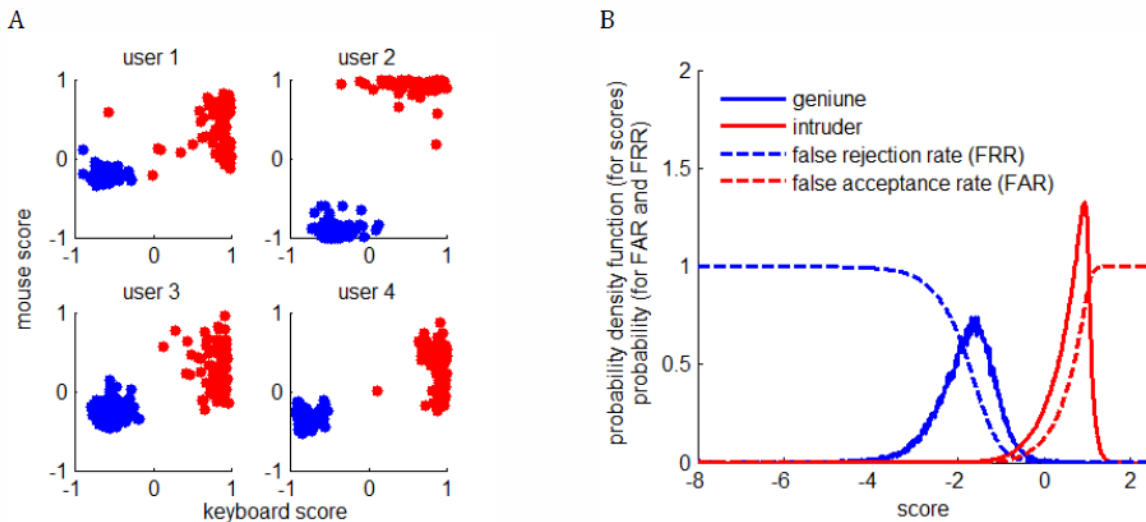
השרטוט המצ"ב מכיל תיאור סכמתי (לדוגמא) של תהליך Transparent User Authentication (זיהוי משתמש באופן "שקוף"):





השרטוט המצ"ב מציג את תוצאות ניתוח תהליך ה-ML בצד שרת, אשר מציג מספר פרמטרים מהותיים:

1. ניתוח פעילות המשתמשים השונים בעת הפעלת ממשק המקלדת והעכבר.
2. זיהוי האם מדובר במשתמש לגיטימי או לא.
3. ניתוח הסבירות לקיומה של טעות מסוג False\Positive.
4. ניתוח הסבירות לקיומה של טעות מסוג Positive\False.



לפיכך ניתן לראות כי שימוש מושכל ביכולות ML יחד עם טכנולוגיית Invisible Challenge-Response (אתגר-תגובה "שקוף") יכול לסייע לארגונים לזהות באופן טוב יותר משתמשים ללקוחות, וזאת במקביל לשיפור יכולת הארגון לזיהוי פעילות חריגות בצד המשתמש לקוח.

#### 1. איתור Maleware (נוזקה) באמצעות שימוש במנוע סטטיסטי CrowdSource

חברת [Invincea Labs](#) בשיתוף עם ארגון [DARPA](#) פיתחו את מערכת [Cynomix](#), אשר מהווה כלי עזר לאנליסטים לייצוג מידע של קובץ בינארי באופן חזותי ("Capability Profile" - פרופיל יכולות), וזאת באמצעות שימוש ב-CrowdSource (מנוע סטטיסטי המבוסס בעיקרו על יכולות עיבוד שפה טבעית), דבר המאפשר לאנליסטים יכולת זיהוי מגמות לאורך זמן. במאמר מוסגר אציין כי המערכת מאפשרת ביצוע Reverse Engineering (הנדסה לאחור), אשר הינו: ["תהליך של גילוי עקרונות טכנולוגיים והנדסיים של מוצר דרך ניתוח המבנה שלו ואופן פעולתו. לרוב, תהליך זה כולל פירוק המוצר למרכיביו, וניתוח פרטני של דרך פעולתם. לרוב, תהליך ההנדוס לאחר מבצע מתוך כוונה להרכיב מוצר חדש הפועל בצורה דומה, מבלי להעתיק למעשה את המקור."](#)

תיאור מופשט של התהליך אשר המערכת מבצעת הינו; ניתוח טקסטואלי סטטיסטי הולם של תוכן פרסומים מהאתר <http://stackexchange.com>, וזאת לשם ביצוע קורלציה עם שדות טקסט בקבצים בינאריים, דבר המאפשר להגיע למסקנות בנושא אופי הקובץ הנבדק. בנוסף, המערכת מאפשרת בניית "Capability Profile" (פרופיל יכולות), אשר יכולים להעיד כי הקובץ הבינארי אכן Maleware (נוזקה). כמו כן, ה-"Capability Profile" (פרופיל יכולות) יכולים לשמש כבסיס להשוואה בין קבצים שונים, ובכך לאפשר לזהות קווים מקבילים בין Malwares (נוזקות), ובכלל זה מוטציה של Maleware (נוזקה).

התרשים הבא מציג באופן סכמתי את תהליך האיתור וההצלבה (תהליך התחקור) בין תוכן הפרסומים באתר <http://stackexchange.com> לשדות טקסט שחולצו מהקובץ הבינארי:

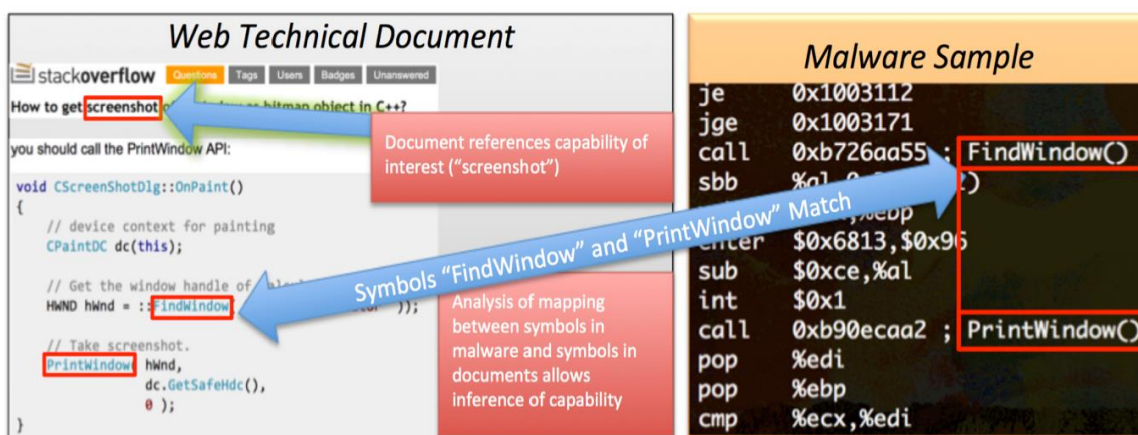


Figure 1. An illustration of the intuition that we can correlate terms found in malware binaries and terms found in StackExchange documents to identify high-level software capabilities within malware

התרשים הבא מציג את תהליך ה-Workflow (תהליך זרימה של התהליך) אשר מייצר בסוף התהליך את ה-"Capability Profiles" (פרופיל יכולות):

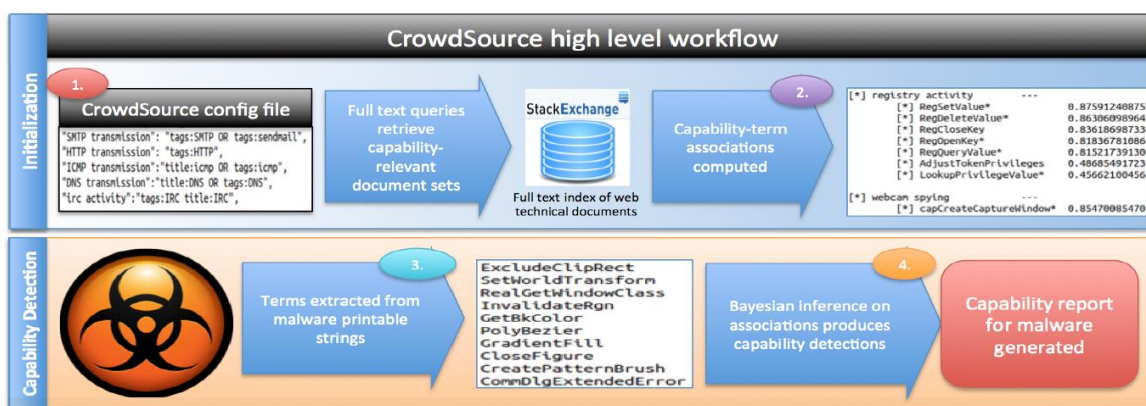
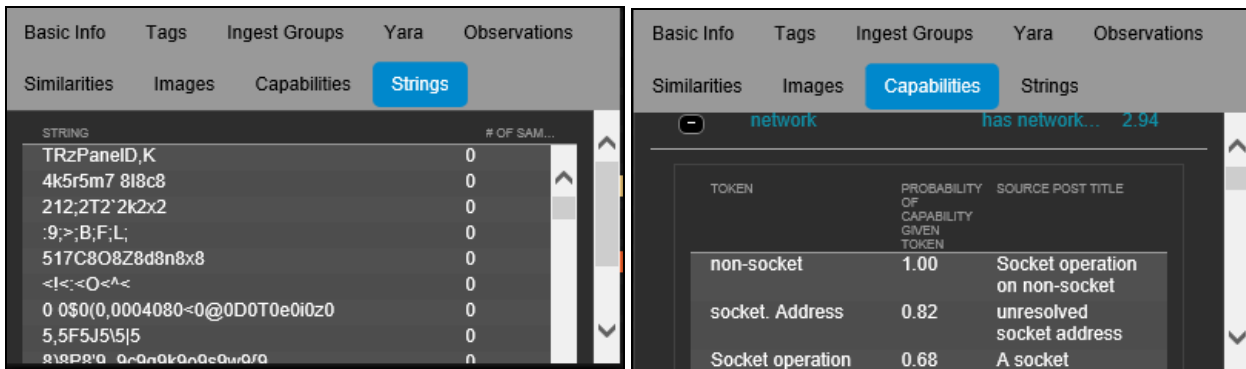


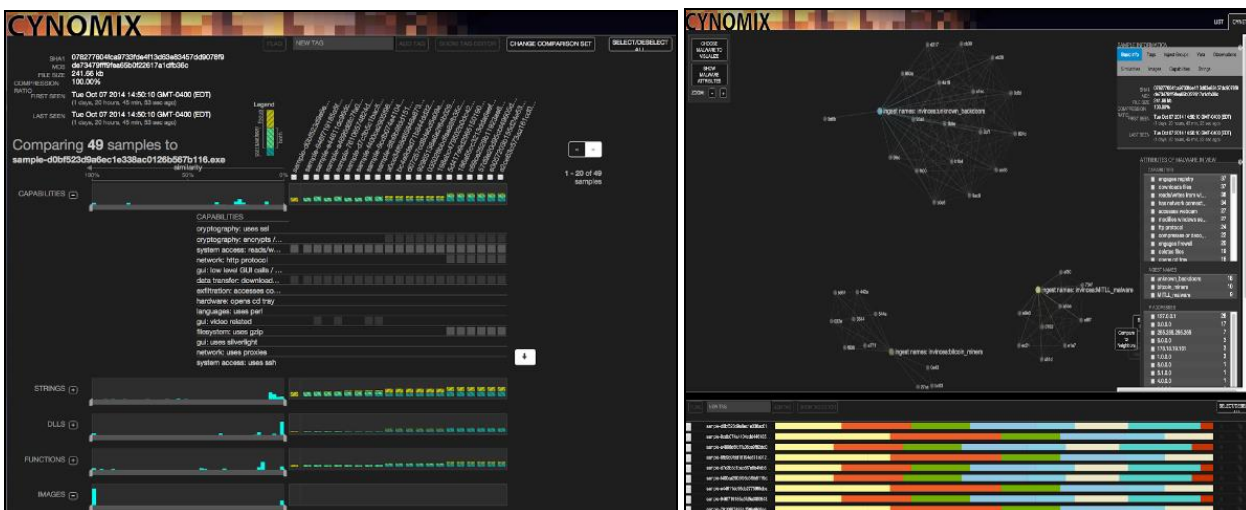
Figure 2. An overview of the CrowdSource workflow.

<sup>11</sup> ניתן לראות כי התהליך כולל שימוש באלגוריתם ממשפחת ה-Bayesian.

צילום המסך בצד ימין מציג את שדות הטקסט אשר חולצו מהקובץ הבינארי. צילום המסך בצד שמאל מציג את ה-"Capability Profile" (פרופיל יכולות) אשר נבנה בסוף התהליך:



צילום המסך בצד ימין מצג השוואה בין קבצים בהתאם ל-"Capability Profile" (פרופיל יכולות). צילום המסך בצד שמאל של המסך מציג את חוזק הקשרים בין הקבצים (דבר הנעשה באמצעות אלגוריתמי ML, כדוגמת אלגוריתם ממשפחת ה-Clustering):



לסיכום, מערכת [Cynomix](http://Cynomix.com), הכוללת את המנוע הסטטיסטי CrowdSource מציעה גישה חדשנית לבניית "Capability Profile" (פרופיל יכולות), אשר יכולים לשמש כבסיס להשוואה בין קבצים שונים, ובכך לאפשר לזהות קווים מקבילים בין Malewares (נוזקות), ובכלל זה מוטציה של Maleware (נוזקה).

המאמר סקר על קצה המזלג את אבני הדרך החשובות בהיסטוריה של ה-AI. כמו כן, המאמר סקר מספר שיטות הלימוד השכיחות, כאשר יש לזכור כי הבחירה בשיטת הלימוד משפיעה באופן ניכר איכות התוצר ביחס לבעיה הניצבת בפנינו. בנוסף, המאמר סקר את תחומי המחקר הדומיננטיים בתקופתנו. מעבר לכך, המאמר כלל סקירה כללית של שימושים אפשריים ב-AI לטובת מתן מענה לסוגיות אבטחת מידע וסייבר שונים, תוך התמקדות ברמת-על במספר פתרונות שכיחים מתת התחום ML - Antispam, Fraud, Transparent User Authentication, Anomaly Detection, Maleware Detection, Face Recognition, ו- Face Recognition. אני מניח כי ברור לכל בשלב זה כי תחום ה-AI נמצא עדיין בחיתוליו, וכי תחום זה מורכב ממודלים ביולוגיים-מתמטיים מורכבים. בניגוד לגישה הרווחת בציבור, לטענת חוקרים רבים קפיצות משמעותיות ביכולות ה-AI עשויות לקחת מאות שנים, אך כבר בשלב זה ניתן לראות מספר מימושים מעניינים של AI בתחום אבטחת מידע והסייבר, אשר יש באפשרותם להקל את על הצד המגן (Protector), והן על הצד התוקף (Predator). כמו כן, לאחרונה התפרסמו דיווחים על התפתחותם של תחומי מחקר (ומודלים) חדשים, כדוגמת [Computational Creativity](#) (יצירתיות חישובית) ו-[Theory of Everything](#) (התיאוריה של הכל), אשר יש באפשרותם להשפיע על תחום ה-AI באופן ניכר.

"All animals exist by instinct, but humankind progress by intelligence - knowledge (the well-justified true belief leading to hope & action). Knowledge of humankind is increasing exponentially day by day (save the behavioural ethics), future is bright only for the knowledgeable kind, AI & machines ought to replace the ignorant!"

Bashir Nedeem



## תודות

ברצוני להודות לפרופ' ליאור רוקח, הפקולטה להנדסת תוכנה, אוניברסיטת בן גוריון על אישורו להשתמש בתכני הלימוד, וכן על התנדבותו למתן משוב למאמר זה. בנוסף ברצוני להודות למר אוהד עשור על מתן המשוב לטיטת המאמר הראשונית. ברצוני אף להודות למר. אבי תורג'מן ומר. אורי ריבנר, יזמי ומנהלי חברת [BioCatch](#) על הסכמתם כי אוכל להשתמש בתיעוד הטכני של החברה כחלק מהמאמר. כמו כן, ברצוני להודות לפרופ' אונגר רון, הפקולטה למדעי החיים, אוניברסיטת בר אילן, ולפרופ' משנה איילת לם, הפקולטה לביולוגיה, הטכניון על ההכוונה לחלק ממקורות המידע אשר אפשרו את העמקת היריעה בנושאים השונים אשר נידונו במאמר.

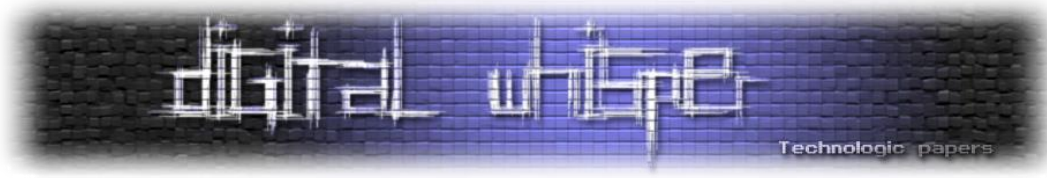
## על המחבר

[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.

## מילות מפתח

*Adaptive Anomaly Detection, Anomaly Detection, Anomalous User Behavior, Artificial Intelligence, AI, A.I., Cyber, Deep Machine Learning, DML, Deep Learning, DL, Data Analysis, Data Scientist, Fraud Detection, Information Security, Machine Learning, ML, Network Behavior Anomaly Detection, NBAD, User Behavior Analysis, UBA*





## ביבליוגרפיה

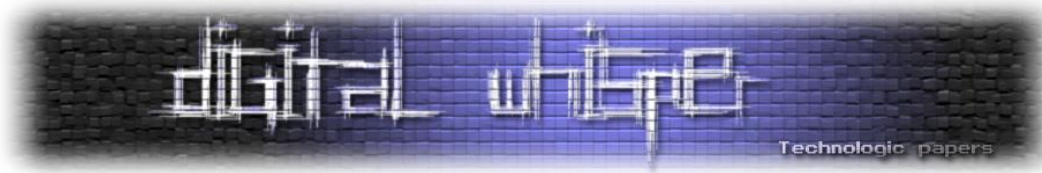
### מושגים שכדאי להרחיב את הדעת לגביהם:

- ASR (Automatic Speech Recognition)
- DNN (Deep Neural Net)
- DSSM (Deep Structured Semantic Model)
- DBN (Deep Belief Network)
- DBM (Deep Boltzmann Machine)
- Decision Tree Learning
- K-nesarest
- Logistic regression
- Naïve-Bayes

### ביבליוגרפיה באנגלית

#### Professional Books:

- Data-Driven Security: Analysis, Visualization and Dashboards, Jay Jacobs, Bob Rudis, Wiley, February 24 2014
- [DEEP LEARNING: Methods and Applications, Li Deng and Dong Yu, Microsoft, 2014](#)
- Machine Learning for Hackers, Drew Conway, John Myles White, O'Reilly, 2012
- Biological Computation (Chapman & Hall/CRC Mathematical and Computational Biology), Chapman and Hall/CRC; 1 edition (May 25, 2011)
- Machine Learning: Concepts, Methodologies, Tools and Applications (3 Volumes), Information Resources Management Association (USA), July 2011
- Transparent User Authentication: Biometrics, RFID and Behavioural Profiling, Nathan Clarke, Springer, 2011
- Artificial Intelligence: A Modern Approach (3rd Edition), Stuart Russell, Peter Norvig, Pearson India, 2009
- Pattern Recognition and Machine Learning, Christopher M. Bishop, Springer, 2007
- [Graph Theory With Applications, John Adrian Bondy, Elsevier Science Ltd/North-Holland, 1976](#)



- Computational Analysis of Terrorist Groups, Lashkar-e-Taiba, V.S. Subrahmanian, Aaron Mannes, Amy Sliva, Jana Shakarian, John Dickerson, Springer, 2013

#### Articles:

- [The History of Automatic Speech Recognition Evaluations at NIST](#)
- [The History of Artificial Intelligence, History of Computing CSEP 590A, University of Washington December, 2006](#)
- [Learning Deep Structured Semantic Models \(DSSM\) for Web Search using Clickthrough Data](#)
- [Encoding and Decoding](#)
- [Deep Boltzmann Machines, Ruslan Salakhutdinov and Geoffrey Hinton, 12th International Conference on Artificial Intelligence and Statistics \(2009\)](#)
- [An Efficient Learning Procedure for Deep Boltzmann Machines, Ruslan Salakhutdinov and Geoffrey Hinton, Neural Computation August 2012, Vol. 24, No. 8: 1967 - 2006](#)
- [ARTIFICIAL NEURAL NETWORKS IN PROTEIN SECONDARY STRUCTURE PREDICTION: A CRITICAL REVIEW OF PRESENT AND FUTURE APPLICATIONS, BIOMEDIN 231: Computational Molecular Biology Professor and Instructor: Doug Brutlag and Dan Davison](#)
- [Deep Machine Learning—A New Frontier in Artificial Intelligence Research, Itamar Arel, Derek C. Rose, and Thomas P. Karnowski, The University of Tennessee, USA, NOVEMBER 2010 | IEEE COMPUTATIONAL INTELLIGENCE MAGAZINE 13](#)
- [An Introduction to Machine Learning Theory and Its Applications: A Visual Tutorial with Examples, Nick McCrea](#)
- [A Deep Learning Tutorial: From Perceptrons to Deep Networks, Nick McCrea](#)
- [Deep Learning Tutorials](#)
- [Data Mining: Classification VS Clustering \(cluster analysis\)](#)
- [Fundamentals of Machine Learning](#)
- [cognitive computing, Margaret Rouse, 2014](#)
- [Cloud Based — Distributed Data Mining](#)
- [Artificial Neural Network \(ANN\)](#)
- [What is Data Science, Mike Loukides, O'Reilly Radar](#)
- [Big Data, Gartner Glossary](#)
- [Tour of Machine Learning Algorithms, Jason Brownlee](#)
- [List of machine learning concepts](#)

---

בפתרונות אבטחת מידע וסייבר Machine Learning **מבוא לשימוש** ביכולות

[www.DigitalWhisper.co.il](http://www.DigitalWhisper.co.il)

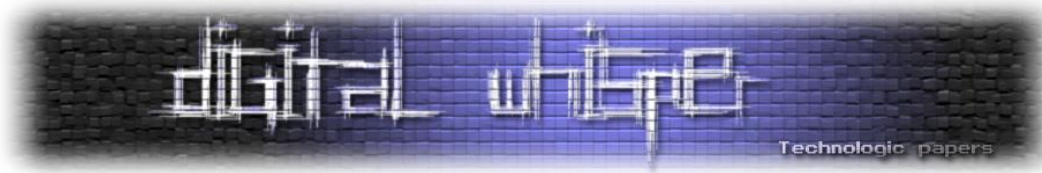




- [Machine Learning and Cognitive Systems: The Next Evolution of Enterprise Intelligence \(Part I\)](#)
- [Machine Learning and Cognitive Systems, Part 2: Big Data Analytics](#)
- [Neural Implementation of Reinforcement Learning, Kenji Doya](#)
- [Predicting the future of artificial intelligence has always been a fool's game](#)
- [Computational Creativity: The Final Frontier?, Simon Colton, Geraint A.Wiggins, Computational Creativity, Centre for Digital Music, School of Electronic Engineering and Computer Science, Queen Mary, University of London, UK](#)
- [Intelligent probing: A cost-effective approach to fault diagnosis in computer networks, M. Brodie, I. Rish, S. Ma, IBM SYSTEMS JOURNAL, VOL 41, NO 3, 2002](#)
- [Psychology of Intelligence Analysis, Richards J. Heuer, Jr.CIA \(Central Intelligence Agency\)](#)
- [Bill Gates is the latest brilliant person to warn artificial intelligence could kill us all](#)
- [New Human-Machine Interfaces: Beyond Verbal Communication, Kerry Doyle](#)
- [IBM updates Watson with five new features, now better than ever](#)
- [Modeling and Predicting Behavioral Dynamics on the Web, Kira Radinskyz, , Krysta Svorey, Susan Dumais, Jaime Teevany, Alex Bocharovy, Eric Horvitz, CS Department, Technion—Israel Institute of Technology & Microsoft Research, Redmond](#)
- [Towards Detecting Anomalous User Behavior in Online Social Networks, Bimal Viswanath, M. Ahmad Bashir, Mark Crovella, Saikat Guha, Krishna P. Gummadi, Balachander Krishnamurthy, Alan Mislove, MPI-SWS, Boston University, MSR India, AT&T Labs-Research, Northeastern University](#)

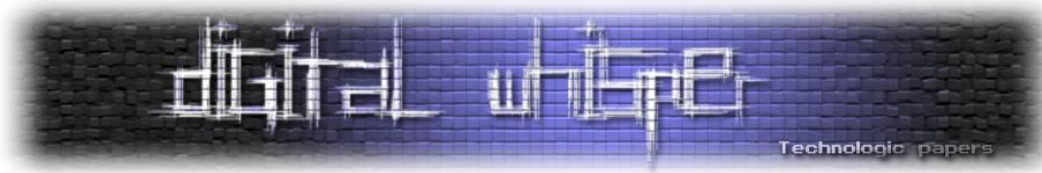
#### **Learning Methodologies:**

- [Reinforcement Learning](#)
- [Supervised learning](#)
- [Unsupervised learning](#)
- [Semi-supervised learning](#)
- [Machine Learning Algorithms Comparison:](#)
- [A Tour of Machine Learning Algorithms](#)
- [Comparison of Machine Learning Algorithms](#)
- [Machine Learning Algorithms Comparison Table](#)
- [Learning Knowledge and Skills Development](#)



## Security:

- [Cybersecurity and Acritical Intelligence, From fixing the plumbing to Smart Water, IEEE SECURITY & PRIVACY, SEPTEMBER / OCTOBOR 2008](#)
- [Cold remedies and Oracle Security](#)
- [Machine Learning Final Project Spam Email Filtering, Shahar Yifrah, Guy Lev, Tel Aviv University, March 2013](#)
- [Machine Learning Techniques in Spam Filtering, Konstantin Tretyakov, Institute of Computer Science, University of Tartu, May 2004](#)
- [Naive Bayes spam filtering](#)
- [You've Got Spam!](#)
- [Defending Networks with Incomplete Information: A Machine Learning Approach](#)
- [Applying Machine Learning to Network, Security Monitoring, Alex Pinto, Chief Data Scien2st | MLSec Proj](#)
- [Getting Smart about Threat Intelligence \(#TIQtest\), Alexandre Pinto, CISSP-ISSAP, Chief Data Scientist, MLSec Project](#)
- [Botnets Behavioral Patterns in the Network, Garcia Sebastian, CTU University, Czech Republic. UNICEN University, Argentina, October 23, 2014](#)
- [Early Security Classification of Skype Users via Machine Learning](#)
- [Data Analytics for Security Intelligence - Cloud Security Alliance \(CSA\), 2013](#)
- [MAEC Language Overview, Version 4.1](#)
- [Characterizing Malware with MAEC and STIX](#)
- [Cylance Whitepaper Math vs Malware](#)
- [CrowdSource: Automated Inference of High Level Malware Functionality from Low-Level Symbols Using a Crowd Trained Machine Learning Model, Joshua Saxe, Rafael Turner, Kristina Blokhin, Invincea Labs & DARPA, 2014](#)
- [Threat Intelligence Overtaking SIEMs and Firewalls as Primary Countermeasure](#)
- [Artificial Intelligence Applications in Database Security, Computer Security Journal, 1990 \(Miller Freeman Publishers\), Vol. 6, No. 1, \(co-authors: W. Tsai, T. Keefe, and D. Thomsen\)](#)
- [Applications of Artificial Intelligence Techniques to Combating Cyber Crimes: A Review, Selma Dilek, Hüseyin Çakır, Mustafa Aydın, International Journal of Artificial Intelligence & Applications \(IJAIA\), Vol. 6, No. 1, January 2015](#)



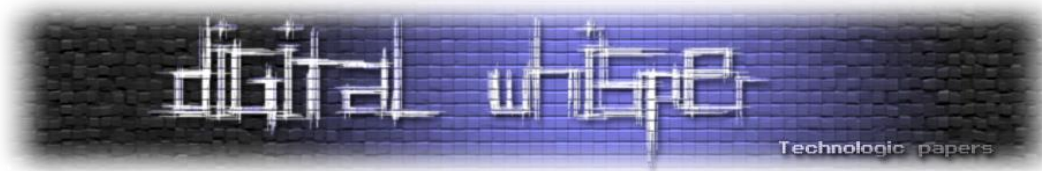
- [Multi-view Face Detection Using Deep Convolutional Neural Networks, Sachin Sudhakar Farfade, Mohammad Saberian, Li-Jia Li, Feb 2015](#)
- [Artificial Intelligence in Cyber Defense, Enn Tyug, Tallinn, Estonia, Cooperative Cyber Defense Center of Excellence \(CCD COE\) & Estonian Academy of Sciences, 2011](#)
- [Cyber Expert. Artificial Intelligence in the realms of IT security, Oleg Zaitsev, 2010](#)
- [Multi-view Face Detection Using Deep Convolutional Neural Networks](#)

#### Online Lectures:

- [\(ML 1.2\) What is supervised learning?](#)
- [\(ML 1.3\) What is unsupervised learning?](#)
- [Artificial Intelligence: Machine Learning Introduction](#)
- [Neural networks \[7.7\] : Deep learning - deep belief network](#)
- [Can Artificial Intelligence Change Cyber Security?](#)
- <http://passbaconference.com/>
- [Fuzzy Logic - Computerphile](#)
- [IBM Watson: How it Works](#)
- [IBM Watson-Introduction and Future Applications](#)
- [Applying Machine Learning for Security Incident Response - Invincea Threat Data Server](#)
- [Cynomix Automatic Analysis, Clustering, and Indexing of Malware](#)
- [Science Documentary - Theory of Everything](#)
- [Jonathan Byrne - Computational Creativity & Evolutionary Design](#)
- [Obama speaks at cyber security summit, Feb 2015](#)

#### Power Point Presentations:

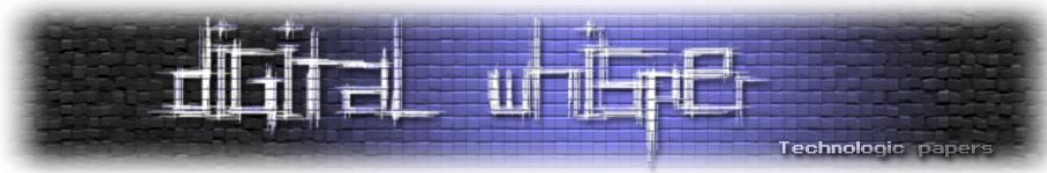
- [When Cyber Security Meets Machine Learning, Lior Rokach, \(5 SlideShares\) , Faculty Member at Ben-Gurion University of the Negev](#)
- [Applying Machine Learning to Network Security Monitoring - BayThreat 2013](#)
- [Artificial intelligence in cyber defense, Ujjwal Tripathi](#)
- [CS104 Information and Information Systems Social Networks and Graph Theory, Morgan Harvey](#)



- [Mapping Internet Sensors with Probe Response Attacks, Protecting Internet Sensor Anonymity, Jason Franklin, Department of Computer Science University of Wisconsin, Madison](#)
- [Mehari: Information risk analysis and management methodology](#)
- [Artificial Intelligence - Our Attempt to Build Models of Ourselves, Elaine Rich](#)
- [Combining Model-Based Testing and Machine Learning, Roland GROZ, LIG, Université de Grenoble, France, TAROT Summer School 2009](#)

#### Samples of Solutions that Using AI/ML:

- <http://www.c-b4.com/>
- <http://megvii.com/>
- <http://www.skymind.io>
- <http://mahout.apache.org/>
- <http://www.aorato.com/> (In 2014 the Company was acquired by Microsoft)
- <http://www.fireblade.com/> (Original Name - SiteBlackBox)
- [Cognitive Assistant that Learns and Organizes](#)
- [J.A.E.S.A : Next Generation Artificial Intelligence](#)
- [Cyber Intelligence System, IAI, ELS-8910](#)
- [PreAlert: Anomaly Detection](#)
- <http://allenai.org/>
- <https://www.palantir.com/>
- [Facebook's DeepFace facial recognition technology has human-like accuracy](#)
- [Computers Are Getting Better Than Humans at Facial Recognition](#)
- [Facebook will soon be able to ID you in any photo](#)
- <http://www.emotient.com/>
- <http://www.invincea.com/tag/cynomix/>
- [BioCatch](#)
- [Cyber Spear](#)
- [EMC NetWitness](#)
- [Memex \(Domain-Specific Search\)](#)
- [AIEngine \(Artificial Intelligent Engine\)](#)



- <http://www.cylance.com/>
- <http://www.lockheedmartin.com/us/what-we-do/information-technology/cyber-security.html>

#### DARPA and IARPA:

- [DARPA's 1980s Vision for Skynet-Like AI](#)
- [DARPA launches PPAML artificial intelligence program to move machine learning forward](#)
- [DARPA Is Developing an Intelligent Machine That Can Think on Its Feet](#)
- [Request for Information \(RFI\) on Research and Development of a Cortical Processor](#)

#### ביבליוגרפיה בעברית

##### ספרות מקצועית:

- ביואינפורמטיקה: אנליזה של רצפים וגנומים - מדריך למידה, אוגור, סטיבן בקר, עדה ניר, האוניברסיטה הפתוחה

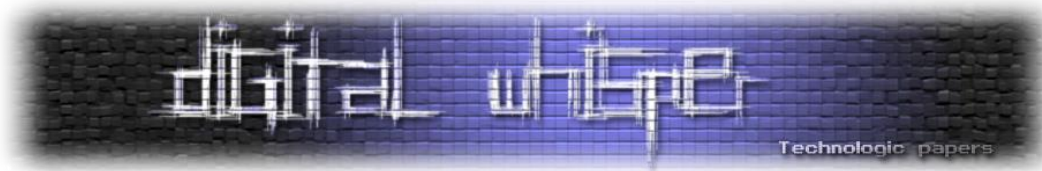
##### מאמרים:

- [בינה מלאכותית, ניר אדר, 2004](#)
- [גוגל פיתחה אינטליגנציה מלאכותית שמזהה תמונות](#)
- [בינה מלאכותית - מחשבים החושבים בכוחות עצמם / יאסר ס' אבו-מוצטפה](#)
- [למה Deep Learning זה הדבר](#)
- [ראסלאן סאלאחודינוב - מודלים יצירתיים ועמוקים](#)
- [חלוקת סוד](#)
- [מבוא למערכות לומדת, ניסוי מעבדות 2-3, מעיין הראל, אורלי אבנר, הטכניון](#)
- [למידה חישובית](#)
- [מדען אורח פרופ' ליאור רוקח](#)
- [למידה חישובית](#)
- [המחשב עדיין לא חושב, יהושפט \(שפי\) גבעון' 2015](#)
- [סקירה בנושא: "בינה מלאכותית"](#)
- [ווטסון לעזרת הרופא: יבמ מציגה מערכת ניתוח נתונים וייעוץ לטיפול בחולי סרטן](#)
- [מערכת ווטסון של יבמ תסייע למאיו קליניק](#)
- [הכירו את BioCatch, הסטארטאפ שיהרוג את הסיסמא \[Microsoft Azure\]](#)

---

בפתרונות אבטחת מידע וסייבר Machine Learning **מבוא לשימוש** ביכולות

[www.DigitalWhisper.co.il](http://www.DigitalWhisper.co.il)



- [מבוא ל-Web 3.0 Security, יובל סיני, Digital Whisper, 2013](#)
- [כישלון שיטות הגנת הסייבר הקלאסיות — מה הלאה? אמיר אורבור, גבי סיבוני, צבא ואסטרטגיה, כרך 5, גיליון 1, אפריל 2013](#)
- [בינה מלאכותית - מבוא והצגת בעיות כגרפים, ניר אדר \(UnderWarrior\), גיליון 1, אוקטובר 2009, Digital Whisper](#)

#### מצגות:

- [רשתות ניורונים, הרצאת טעימות, ד"ר אורן שריקי, הפקולטה למדעי המחשב, אוניברסיטת בן גוריון](#)

#### הרצאות Online:

- [קורס למידה חישובית \(למידת מכונה\) - פרופ' ליאור רוקח - הרצאה 1](#)
- [למידה חישובית - פרופ' ליאור רוקח - הרצאה שניה - מבוא ועקרונות נראות מקסימלית](#)
- [קורס למידה חישובית \(למידת מכונה\) - פרופ' ליאור רוקח - הרצאה 7 - רשתות ניורונים חלק ב](#)
- [קורס למידה חישובית \(למידת מכונה\) - פרופ' ליאור רוקח - הרצאה 11 Support Vector Machines -](#)
- [קורס למידה חישובית \(למידת מכונה\) - פרופ' ליאור רוקח - הרצאה 12 - זיהוי אנומליות](#)
- [מחשוב חכם יותר - מה תעשו עם ווטסון, מחשב העל של יבמ?](#)
- [ערב עיון: יחסי אדם-מכונה, לאן?](#)



---

## The Husky Code

מאת גל תא שמע

---

### הקדמה

בזמן האחרון התחלתי להתעסק מעט ב-javascript. ככל שעבר הזמן למדתי לאהוב את התכונות המוזרות של השפה, אמנם לרוב הן לא שימושיות אך לפעמים הן מאפשרות דברים נחמדים. התכונות האלו אפשרו לי ליצור את היצירה הזאת. הגיתי את הפרויקט לאחר שחבר הראה לי כל מיני סוגים של אובפוסקציות ב-javascript. המאמר של Ender (גיליון 56), נתן לי את ההשראה לכתוב על הפרויקט. את הפרויקט עשיתי ביחד עם אחי, נועם תא שמע. נהננו מכל רגע. (:

### הפרויקט

מטרת הפרויקט הייתה ליצור קוד javascript המורכב מקבוצה מוגבלת של אותיות. הקוד צריך להיות בנוי כך שנוכל לכתוב אותו בצורות, ולא יחייב מבנה פיזי ספציפי. על הפרויקט נועם ואני עבדנו בערך 24 שעות, את שאר השבועות אחר-כך השקענו בניסיון להסביר מה לעזאזל עשינו ולמה זה עובד ©. התוצר הסופי הוא ה"האסקי הסקי" וקומפיילר ההופך את כל תהליך היצירה של ההאסקי שבמאמר לאוטומטי.



## כתיבת javascript חלקי עם אוצר מילים מוגבל

בפרויקט שלנו החלטנו להשתמש רק באותיות A,S,C,I (ובסימנים), אך יש להבין שמדובר בהחלטה שרירותית, היינו יכולים לבחור כל קבוצה אחרת של אותיות והקוד עדיין אמור לעבוד. אם היה מדובר בשפה של בני אנוש, כנראה שלא היינו מצליחים לתקשר אחד עם השני, למזלנו מדובר בשפה של מחשבים ולפעמים הם קצת יותר טובים מאיתנו בתקשורת. למען האמת, כנראה שרוב השפות לא היו מאפשרות לעשות את מה שאנחנו מבקשים לעשות אך למזלנו javascript היא שפה מיוחדת בפרט הזה.

### אותיות ללא אותיות

חלק מהדוגמאות בחלק הבאה בהשראת [הפוסט](#) של Patricio Palladino. בדוגמה למטה אפשר לראות התנהגות מוזרה של השפה. בשביל להבין את השורות הבאות צריך להבין עקרון מפתח: כאשר מבצעים חיבור על האובייקט [] הוא קורא באופן implicit לפונקציה toString של האובייקט המשורשר. בפשטות, הוא ממיר את שני האובייקטים למחרוזות ומשרשר ביניהם. התכונה הזאת פועלת גם על כמה אובייקטים אחרים אך היתרון ב-[] הוא שהערך של עצמו הוא מחרוזת ריקה, לכן הוא "שקוף" כאשר משרשרים אותו:

```
// makes the letters a,b,c,d,e,f,I,i,j,l,N,n,O,o,r,s,t,u,y [, ]  
[console.log([]+{})] // [Object object]  
console.log([]+![]) // false  
console.log([]+!![]) // true
```

הדוגמאות הבאות קצת יותר מורכבות:

```
console.log(([]+[]) + []) // undefiend  
console.log(+{}+[]) // NaN  
console.log(![]/[ ]+[]) // Infinity
```

הסבר על 3 השורות האחרונות:

- []+[] - אנו מנסים לגשת למערך במקום לא מוגדר, לכן מוחזר לנו: undefined.
- +{} - הסוגריים המסולסלים יוצרים אובייקט, בגלל שאנחנו מנסים להמיר אותו למספר אנחנו מקבלים NaN (לא מספר).
- ![]/[ ] - מערך ריק הוא שווה ערך ל-0, לכן כאשר נחלק מספר ב-0 (שהוא לא 0 בעצמו) נקבל אינסוף.

נשתמש במה שלמדנו למעלה וניצור את המחרוזת "alert 1", קיבלנו את השורה הבאה:

```
("(1)"+(0)+(![]+[]) (1)+(![]+[]) (4)+(![]+[]) (2)+(![]+[]) (1)+(![]+[]) (1) ")
```



## מספרים ללא אותיות

כמובן שזה לא מספיק, אסור לנו השתמש במספרים, לכן יהיה עלינו לייצר אותם כמו שיצרנו את האותיות קודם. לייצר מספרים זאת משימה קצת יותר קלה. הפעם יש לנו שני עקרונות מנחים:

- הפעולה + ממירה אובייקט לערך המספרי שלה.
- האובייקט false שווה ערך ל-0 ו-true ל-1.

לדוגמה:

```
+ [] //0
+!! [] //1
!+ []+!! [] //2
!+ []+!! []+!! [] //3
!+ []+!! []+!! []+!! [] //4
!+ []+!! []+!! []+!! []+!! [] //5
!+ []+!! []+!! []+!! []+!! []+!! [] //6
!+ []+!! []+!! []+!! []+!! []+!! []+!! [] //7
!+ []+!! []+!! []+!! []+!! []+!! []+!! []+!! [] //8
!+ []+!! []+!! []+!! []+!! []+!! []+!! []+!! []+!! [] //9
```

[] הוא שווה ערך ל-true, לכן כשנמיר אותו למספר נקבל 0. []! שווה ערך ל-true לכן נקבל 1. גם הביטוי []+! שווה ערך ל-true. עכשיו אנחנו יכולים לעמוד בתנאי הראשון שהגדרנו. נכתוב מחדש את הדוגמה הקודמת, הפעם ללא הספרות ונקבל את השורה הבאה:

```
(! []+[]) [+!! []]+(! []+[]) [!+ []+!! []]+(! []+[]) [!+ []+!! []+!! []]+(! []+[]) [+!! []]+(! []+[]) [+!! []]+["(+!+[])+" ] // "alert(1)"
```

## עוד קצת אותיות

בעזרת שילוב של המספרים והאותיות מלמעלה ניתן להשיג אותיות נוספות. כדי לעשות זאת ננצל את העובדה שניתן לגשת לתכונות של אובייקט בשני דרכים:

1. גישה לאובייקט תכונה.

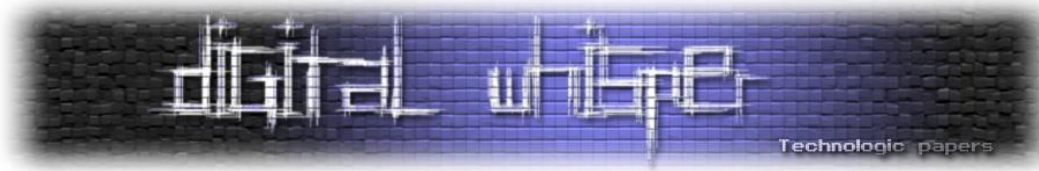
2. גישה לאובייקט ["תכונה"].

בפועל משמעות שני השורות זהה. יתרה מזאת, פונקציה יכולה להיות תכונה של אובייקט. העיקרון הזה מאפשר לנו לגשת לפונקציות מבלי לכתוב במפורש את שם הפונקציה שאליה ניגש. כדי להמחיש:

```
obj.funcName === obj["funcName"]
```

נוכל להשתמש בשורות הבאות כדי להשיג עוד אותיות. (כמובן שאת כל המחרוזות נבטא כמו שלמדנו קודם).

```
[] ["sort"] ["constructor"] //Function
[] ["constructor"] //Array
(! []) ["constructor"] //Boolean
(+! []) ["constructor"] //Number
{} ["constructor"] //Object
([!+[]]) ["constructor"] //String
```



## כתיבת javascript מלא

כדי שנוכל לכתוב בתחביר המלא של javascript, נצטרך למצוא דרך לבטא לפחות את כל אותיות ה-ASCII. אמנם ב-javascript כל אות מיוצגת ע"י תו utf-16 אבל ה-syntax של השפה עצמה מבוסס על אותיות ASCII ולכן מספיק להשיג רק אותם (למרות שאפשר גם יותר). בפסקאות הבאות נסקור שלוש דרכים שונות להשלים לנו את כל האותיות החסרות. כמובן שכל השיטות מסתמכות על דרך להריץ מחרוזת כ"פקודה".

- escaping בעזרת פונקציה.
- התנהגות מוזרה של מנוע ה-javascript V8 והפונקציה toString.
- escaping טבעי ב-javascript.

### escaping בעזרת פונקציה

באופן טבעי, כדאי לנו להפיש את מבוקשנו בפונקציות escaping כמו ו-unescape decodeURI. לשתיהן לא נוכל לקרוא עדיין, כיוון שאין לנו מספיק אותיות. מבין שתי הפונקציות אנחנו יותר קרובים להצליח לכתוב את הפונקציה unescape. להזכירכם, בינתיים יש לנו רק את האותיות הבאות. האותיות הכתומות הן האותיות האותיות הנוספות שהשגנו.

```
A, a, B, b, c, d, e, F, f, g, I, i, j, l, m, N, n, O, o, r, S, s, t, u, y.
```

כדי לקרוא לפונקציה חסרה לנו רק האותיות w ו-p. כדי להשיג את אותה, נוכל להשתמש בפונקציונלית של javascript בהמרה בין בסיסים. השורה הבאה תיתן לנו את המחרוזת "ק":

```
(25).toString(26) // p
(32).toString(33) // w
```

הפונקציה window.unescape עובדת בצורה הבאה:

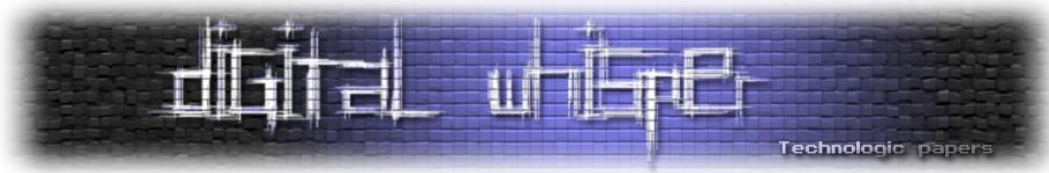
```
window.unescape("%" + HEXA_ASCII_VALUE)
```

[ניתן לקרוא עוד על הפונקציה כאן]

### התנהגות מוזרה של V8 (מנוע ה-javascript של כרום)

שיטה נוספת מנצלת באג מוזר במנוע ה-V8 של כרום. משום מה כאשר מחשבים מספר גדול ומנסים לייצג אותו בתור מחרוזת עם בסיס אי זוגי (כמו 33, או 35), המחשב פולט יותר אותיות מהדרוש. אני חייב לציין שאני לא מבין לגמרי מה הולך שם בדיוק, אתם מוזמנים לקרוא עוד פה. את ההתנהגות הזאת אנחנו יכולים לנצל לקבל את כל האותיות חוץ מ-z (האות ה-36)

```
(1.15369999999997645e-10).toString(35)
0.0000007erx1a0hn1fm2728fnw2y3orr9g8u5j1c1ootamq9nkgtvxgt00hsjgp03mm7v09
fw4u15n80xjjekf1aotqr5mq1bkdqg6suffum5rvutblwwu0f2uxpqr0u0460emj18g7pe01
e29gf7wfmqwj2jsgr31ewao653dld7vfd8q21jm7p6rktaogc6pt8piae0jfi5d6k9u8wmeb
8b0j1sxjto035qcwso7od3anm4b1otchaqs850ht95x1xnsblbi6fyd10yewenm9bd0ch3je
```



uc1av7752tj6nr1w8u3s2mxbv19a3cljw8a23x924fjgksoqlg8tqh6ct099e5nxgsiercop  
sk9mqrlr5qi048o43di4y7ehgbxt3904549cx7x51ve8xsibvgrygpcxa3u2df6t9qs0c5bkr  
p9993d7n4ggipslednjm72186sk2ixdx0cd04g1lyamawmdh3ov00vvnqgh3v1awuuagv76  
ycqfaqa2wurln1xnio3c9p1d35xoj6on5icfpec9vj84xwgvghvf9ix2k6vww3huicclkvmo  
0rjmv282ikjdce0ai80vs8v4dc4nlfr5xrxjtaodefwarcybk1p7ixj6pwnyfhmyq28f2ls  
mhswn7gwebldnivyf6adumf5yf43nd6b4jm9kah7kcu85dnifffs56y9dnp0ylax74r4ffsub  
x96ukq5y82r9lb3gxqxvbdv829nlrxhxjeac4ey8vhdlyitxiq4tbu9pmwp8xulbd64fcune  
ejn0yu79flrsft3tbjx4nqk3ggles0blifl2qikssff9osolni9ge65i7l05af4lrvhcmudb  
lw35gna70ld7ksmxonxir14rqv908p4joepjmrys6g2cpm8u87adwx6l8l1xmc90fppsyyvkq  
b6tq8bh0x4go7vsoowgo66bkgrwkwnduimk77tak9q3qxffu083n9634rt9fir0o7la9ifm  
c601kik08l3dva6tomrt4spn8u2tkwxhx5qxsx7c3he3mdi4kv8ppi3c04nayngpo0b468bn  
7e211nqkbg4nhnltcew4

בכלל, כל אות שחסרה לנו מהאותיות הקטנות (lower) אפשר גם להשיג בעזרת הפונקציה `.toString`.  
לדוגמה:

```
(12).toString(36) // "c"  
(35).toString(36) // "z"  
(17).toString(18) // "h"  
(22).toString(23) // "m"
```

### השיטה הקלה - implicit escaping

בשיטה הבאה נשתמש ב-"character escape sequences". ב-javascript יש כמה סוגים שונים, ההבדל המרכזי ביניהם הוא בסיס הספירה שבו מיוצגות האותיות. שלושת השיטות לאסקיפינג שכזה הן:  
אוקטלי, הקסהדסימלי, ו-unicode.

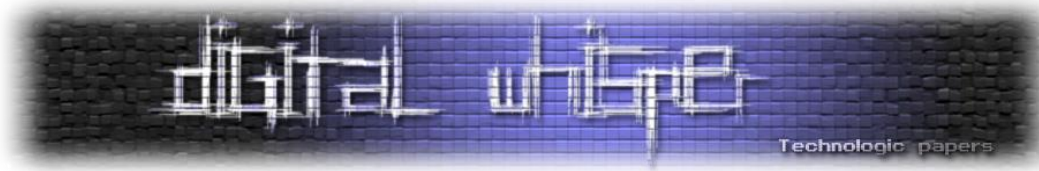
- octal: `"\110\105\114\114\117"`
- hexadecimal: `"\x48\x45\x4c\x4c\x4f"`
- unicode: `"\u0048\u0045\u004c\u004c\u004f"`

הקסהדסימלי לא כדאי לנו כיוון שנצטרך להשיג בשבילו את התו `x`. נשארנו עם unicode ואוקטלי, אני בחרתי להשתמש בשיטה האוקטלית כיוון שהיא משתמשת בפחות תווים מיותרים באופן משמעותי, הדבר נובע מהאפסים המיותרים (לרוב) לאחר ה-`u`.

הסבר קצר על השיטה האוקטלית: השם אוקטלי (octal) נובע מהעובדה שהייצוג הוא בבסיס 8, בפרט זה אומר שנשתמש בספרות 0-7 בלבד. היתרונות בשיטה הזאת הם הפשטות מצד אחד והיכולת לכתוב כל סימן מצד שני. להלן קוד שממיר מחרוזת לייצוג האוקטלי שלה:

```
function f(s) {  
  var msg = "";  
  for(index in s){  
    msg += '\\'+ s.charCodeAt(index).toString(8);  
  }  
  return msg;  
}
```

[מקום להעשרה על character escape sequences: <https://mathiasbynens.be/notes/javascript-escapes>]



## איך להריץ מחרוזת

הדרך הידועה ביותר להריץ מחרוזות כקוד ב-javascript היא בעזרת הפונקציה eval. למעשה הפונקציה עושה בדיוק את מה שרצינו אך היא משתמשת באותיות "אסורות", לכן לא נשתמש בה. אמנם נוכל ליצור את המחרוזת "eval" אך לא נוכל להריץ אותה כ"פקודה". יש מגוון רחב של פקודות כאלו ואחרות המאפשרות להריץ קוד כמו eval שלא נשתמש בהן מאותה הסיבה. בין היתר: "location=" או .setTimeout

אני אחדד את הצורך שלנו: אנחנו צריכים פונקציה שבהינתן מחרוזת, תריץ אותה, מבלי שהשם שלה יכיל אותיות. יש לציין שלא נוכל להשתמש באותיות שכבר קיבלנו למעלה לאור העובדה שאי אפשר להריץ שרשור של אותיות כ"פקודה". בדיוק בשביל זה בא לעזרתנו בחור נחמד בשם Yosuke HASEGAWA שפיתח שיטה להסתרת קוד javascript ללא שימוש בתווים אלפא-נומריים:

<http://utf-8.jp/public/jjencode.html>

הרעיון שלו מתבסס על העובדות הבאות:

- כל דבר ב-javascript הוא אובייקט.
  - כל אובייקט מממש פונקציה המחזירה את הבנאי שלו.
- תאורטית, אם נלך מספיק אחורה בשרשרת הבנאים נגיע לאב הקדמון שאחרי על בניית פונקציות. הוא יהיה מסוג פונקציה כיוון שכל בנאי הוא פונקציה והפונקציה הזאת תהיה בעלת יכולת לעצור פונקציות.

ניקח את ה-flow הבא לדוגמה:

◀ אובייקט מסוג מספר. הבנאי שלו:

◀ בנאי מספרים מסוג פונקציה. הבנאי שלו:

◀ **בנאי פונקציות** מסוג פונקציה.

כלומר לאחר שני קפיצות נוכל להגיע לפונקציה שתפקידה לבנות פונקציות, מעניין. בפועל המימוש של הפונקציה מקבל מחרוזת ומחזיר פונקציה שמכילה את המחרוזת. לדוגמה:

```
// returns a function builder
(function(){})["constructor"]

myFunc = 0["constructor"]["constructor"]("alert(42)")
// function anonymous() {
// alert(42)
// }
myFunc() // will show a popup (42)
```

איך זה עוזר לנו? אנחנו יכולים להריץ כל מחרוזת כקוד. בשילוב עם ה-escaping מבסיס אוקטלי נוכל להריץ כל קוד ללא תלות באותיות המשמשות את הקוד הנראה לעין. כדי שהכל יעבוד, נבנה את





המחרוזות constructor ו-return מראש, משם והלאה נוכל להשתמש בכל אות שעולה בדעתנו כמו בדוגמה הבאה:

```
(["constructor"] ["constructor"] ("return'"+"\\'"+"1"+"1"+"0"+"\\'"+"1"+"0"+"5"+"\\'"+"1"+"1"+"4"+"\\'"+"1"+"1"+"4"+"\\'"+"1"+"1"+"7"+"i")) ()
```

השתמשתי במספרים ובאותיות כדי לחסוך בתווים ולהקל על ההדגמה. לא רציתי לכתוב 34 תווים כל פעם שהספירה 7 נכתבת. נוכל להשתמש באותו רעיון כדי לקצר את הקוד גם בפועל. נשמור משתנים בהתחלה עבור כל הספרות והמילים הנחוצות להתחלה. מימוש אפשרי לזה יהיה:

```
A=+[], I='\'', C=[], S='', CIAA='\'';
// make 0-9 a-f characters
CIA=(!C+S)[A], CCA=A++, CSA=(!C+S)[A], CAI=A++, CII=({}+S)[A],
CCI=((C[+C])+S)[A], CSI=A++, CAC=A++, CIC=(!C+S)[A], CCC=A++,
CSC=({}+S)[A], CAS=A++, CIS=A++, CI=([+{}])[A], CCS=A++, CSS=A++,
CAAA=A++,
// make the constructor and reutrnr strings
CC=(!!C+S)[CAI]+CIC+(!C+S)[CCA]+(!C+S)[CSI]+(!C+S)[CAI]+((C[+C])+S)[CAI],
CS=CSC+(({}+S)+S)[CAI]+((C[+C])+S)[CAI]+(!C+S)[CAC]+(!C+S)[CCA]+(!C+S)[CAI]+(!C+S)[CSI]+CSC+(!C+S)[CCA]+(({}+S)+S)[CAI]+(!C+S)[CAI],
CA=CAI[CS][CS]; //make the function maker
```

עכשיו רק נשאר לנו לקרוא ל-CA ולהשתמש במשתנים שהגדרנו קודם בשביל המספרים.

Yosuke HASEGAWA עשה את אותו הדבר בדרך קצת שונה. הוא שמר את כל המשתנים בתוך dictionary. בפועל לא מדובר בהבדל גדול, מצד אחד הוא מוסיף לקוד הסופי שלנו הרבה נקודות, מצד שני הוא יצור הרבה פחות משתנים גלובליים ומקטין את הסיכוי להתנגשות עם משתנים אחרים.

כשנועם ואני מימשנו את ה"קומפיילר", השתמשנו בכל מיני שיטות בגרסאות השונות. בסוף בחרנו להשתמש באותה השיטה כמו של Yosuke HASEGAWA ביחד עם שמות מג'ונרטים לאיברים של ה-dictionary, ככה אנחנו לא מוגבלים לסט אותיות ספציפיות (אצלנו ASCII). אתם מוזמנים להסתכל על הקוד ואפילו לתרום [=

## סיכום ביניים

ראינו שיש הרבה דרכים לבטא ביטויים ב-javascript מבלי לכתוב אותם ב"אופן מפורש". כמובן שלא צריך את כל הפתרונות, אפילו לא את רובם. בפועל יצא שהמימוש שלנו דומה לזה של Yosuke HASEGAWA, כמובן שלא מדובר בהפתעה כיוון שלטעמי המימוש שלו מאד אלגנטי.

כמו שראיתם עד עכשיו, ניתן לחלק את הקוד לחלקים שונים בעלי תפקידים שונים. הנה סקירה של החלקים השונים ותפקידם:

- סגול וסגול מודגש:** מגדיר את המשתנים שימשו בהמשך. החלק המודגש הוא ה-dictionary.
- אדום:** יוצר את הפונקציה היוצרת, כמו שמוזכר ב"איך להריץ מחרוזת".
- ירוק:** יוצר פונקציה המכילה את המחרוזת הכתומה. כאשר נקרא לפונקציה שיצרנו הדבר יהיה שקול להרצה של הפקודה eval רק שהקונטקסט שלנו יהיה של פונקציה, מה שיאפשר לנו להחזיר ערך. הפונקציה נקראת מיד לאחר ההגדרה שלה והערך יוחזר לתוך הפונקציה הכחולה.
- כחול:** פונקציה, כאשר נקרא לה תריץ את הפלט של הפונקציה הירוקה.
- כתום:** התו המודגש הוא בעצם המחרוזת "return". אחריו יש את כל האותיות שירכיבו יחד את הביטוי האוקטלי של הקוד שרצינו. כמובן שאם יש אות שיצרנו בהתחלה לא נצטרך לעשות לה escaping ולכן פשוט נוסיף אותה כמו שהיא.

```
A=+[ ];I='\ ';C=[];S='';A={AA:[ ],IA:(!C+S)[A],CA:A++,SA:(!C+S)[A],AI:A++,
II:({}+S)[A],CI:((C+[C])+S)[A],SI:A++,AC:A++,IC:(!C+S)[A],CC:A++,SC:({}+
S)[A],AS:A++,IS:A++,I:([ ]+{})[A],CS:A++,SS:A++,AAA:A++};A.I+=A.I;A.AA=A.
I+A.I+A.I+A.I+A.I;A.IAA='\ '\ ';A.C=(!!C+S)[A.AI]+A.IC+(!!C+S)[A.CA]+(!!C+
S)[A.SI]+(!!C+S)[A.AI]+((C+[C])+S)[A.AI];A.S=A.SC+(({}+S)+S)[A.AI]+((C+[
C])+S)[A.AI]+(!C+S)[A.AC]+(!C+S)[A.CA]+(!C+S)[A.AI]+(!C+S)[A.SI]+A.SC
+(!!C+S)[A.CA]+(({}+S)+S)[A.AI]+(!C+S)[A.AI];A.A=A.AI[A.S][A.S];A.A(A.A
(A.C+I+A.SA+A.IAA+A.AI+A.AS+A.CC+A.IC+A.IAA+A.AI+A.IS+A.SI+A.IAA+A.AI+A.
IS+A.CC+A.IAA+A.AS+A.CA+A.IAA+A.CC+A.CS+A.IAA+A.AI+A.AI+A.CA+A.IC+A.IAA+
A.AI+A.AS+A.CC+A.IAA+A.AI+A.AS+A.CC+A.IAA+A.AI+A.AS+A.CS+A.IAA+A.CC+A.CA
+A.IAA+A.AI+A.CA+A.CC+A.IAA+A.AI+A.AS+A.AI+A.IAA+A.AI+A.CC+A.CS+A.IAA+A.
AI+A.AS+A.AI+A.IAA+A.AI+A.IS+A.CC+A.SA+A.IAA+A.AI+A.AS+A.CC+A.IAA+A.CC+A
.CA+A.IAA+A.AI+A.SI+A.CS+A.IAA+A.AI+A.AS+A.CA+A.IAA+A.AI+A.AS+A.AI+A.IAA
+A.AI+A.IS+A.AC+A.IAA+A.AI+A.IS+A.CA+A.IC+A.IAA+A.AI+A.IS+A.SI+A.IAA+A.C
C+A.AI+A.IAA+A.CC+A.CS+A.IAA+A.AS+A.AI+I)())()
```

בפועל הריצה תראה כך:

**בשלב הראשון**, נחבר את כל התווים הכתומים ונקבל את הפלט הבא:

```
"return'a\154e\162\164\50\47\110e\154\154\157\40\104\151\147\151\164a\154\40\127\150\151\163\160e\162\41\47\51'"
```

**בשלב השני**, ניצור פונקציה ונריץ אותה. נקבל את הפלט הבא:

```
"alert('Hello Digital Whisper!')"
```



בשלב האחרון, ניצור עוד פונקציה המכילה את הקוד הנ"ל ונריץ אותו:

### איך להכניס תמונה לקוד

נתחיל מהתמונה. תחילה נצטרך להמיר את התמונה לתבנית, התבנית תורכב משני סימנים (אצלי התו # ורווח). אם היה זה עולם מושלם היינו יכולים פשוט להחליף כל סולמית באות מהקוד המקורי, לצערנו הקוד בצורה הזאת עלול לא לעבוד. javascript לא מאד נוקשה בנהלים לירידת שורה למעט שני מקרים, ירידה באמצע שם משתנה וירידה באמצע מחרוזת. הפתרון הוא פשוט:

1. נחלק את הקוד שלנו לחלקים הקטנים ביותר.
2. נחלק את התבנית לקבוצות של חלקים זהים.
3. נחליף את הקבוצות מהתבנית בקבוצות מהקוד. אם ביטוי javascript ארוך מהמקום בתבנית, נשים במקום הקצר ביטוי חסר משמעות כמו הערה /\*\*/ ונקווה שהמקום הבא יהיה ארוך יותר.

לדוגמה התבנית:

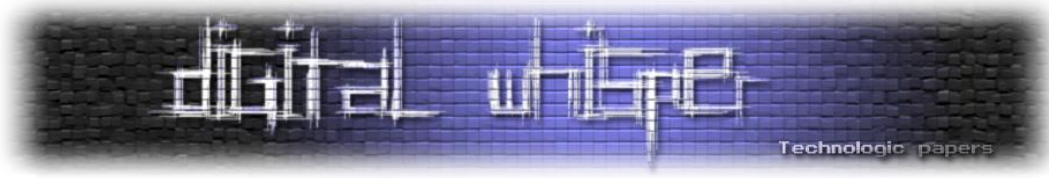
```
#####          #####          ###
#####          #####          ###
###            ###            ###
###            ###            ###
###            ###            #####
###            ###            #####
###            ###            ###
#####          ###            #####
#####          ###            #####
```

אם נחלק את הקוד שלנו לחלקים נקבל:

```
["A", "=", "+", "[", "]", ";", "I", "=", "''", ";", "C", "=", "[", "]", ";", "S", "=", "''", ";", "A", "=", "{", "AA", ":", "[", "]", " ", " ", "IA", ...]
```

עכשיו ביחד:

```
A=+[ ];I=''';C=[ ];S=''';A={AA:[ ] ,IA
:(!C+S) [A],CA: A++,SA:(!C+S) [A ] ,
AI: A++ ,II :({
}+S ) [A ] , CI:
((C [+C ])+S) [A],SI:A++ ,AC
: A++ ,IC:(!C+S) [A], CC:
A++ ,SC :({ }+S ) [A
],AS:A++,IS:A++ ,I: ([ +{ }) [A],CS:A++,
SS:A++,AAA:A++} ;A. I+= A.I;A.AA=A.I+A.
```



## קוד שמדפיס את עצמו (Quine)

נהוג לומר לפני ריצה: "תתחיל חזק ותגביר את הקצב לאורך הדרך". לאור שהעובדה שהקוד שלנו סוף כל סוף רץ, החלטנו להקשיב למשפט ולהוסיף לקוד שלנו טריק נוסף. היכולת להדפיס את עצמו, או כפי שנהוג לומר Quine. ויקיפדיה מגדירה quine כקוד לא ריק שלא מקבל קלט ומייצר עותק של עצמו כפלט היחיד. בפועל, לא רצינו להגביל את הקוד שלנו לזה, לכן החלטנו לאפשר לו לממש תכונות של quine. בעצם אפשרנו לתוכנה לגשת לקוד של עצמה הנמצא תחת המשתנה quine.

בקהילה נהוג לכתוב תכניות שכאלו בצורה שלא בונה על התכונות הספציפיות של השפה בה הם מומשו. אלא שלאור העובדה שרצינו שהקוד ימלא תפקיד מלבד מלייצר את עצמו, החלטנו להשתמש ביכולות השפה לטובתנו. לכן השתמשנו בעובדה שלפונקציות ב-javascript יש המרה יפה למחרוזת, לדוגמה:

```
function f(){
  console.log(1)
}

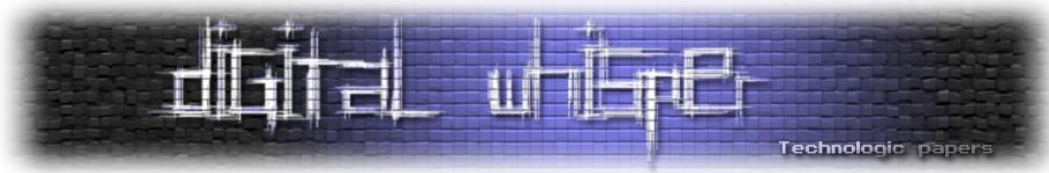
console.log(f+'')
// function f(){
//   console.log(1)
// }
```

אם נסתכל על המבנה של הקוד שלנו, נשים לב שלא כל הקוד עטוף בפונקציה לכן החלקים שבחוץ לא יופיעו כשנססה להדפיס את הפונקציה המרכזית. לכן נשרשר את ההגדרה של המילון בהתחלה ואת הקריאה לפונקציה בסוף כדי לקבל את "התמונה המלאה".

הקוד למטה ידע להדפיס את עצמו כל פעם. רק כדי להדגים שהשיטה עובדת ולא מדובר סתם בקריאה ל-alert על אותו הערך כל הזמן, עדכנתי בכל ריצה את המשתנה s. השיטה שהשתמשנו בה בפועל דומה מאד לדוגמה למטה. אפשר לחשוב על המשתנה s כעל ה-dictionary בתחילת הקוד ועל q כמשתנה המחזיק העתק של הקוד.

```
s = "something before"
f = function(){
  s += "q";
  alert(s);
  q = "s = \""+s+"\";\nqf="+f+";f();";
};f();
```

בפועל יש עוד כמה בעיות בהדפסה עצמית, הראשית מביניהם היא כזאת: אנחנו רוצים את הקוד של הפונקציה, אך הוא יכול את התוכן של הפונקציה לאחר חיבור המשתנים ולא ייצג את איך שהפונקציה נראית במציאות.



כדי להסביר את הנקודה אני אדגים: איך שהיינו רוצים שהמחרוזת תיראה:

```
console.log(f) // A.C+I+A.SA+A.IAA+A.AI ...
```

איך היא תראה בפועל:

```
console.log(f) // "return'a\154e\162\164 ...
```

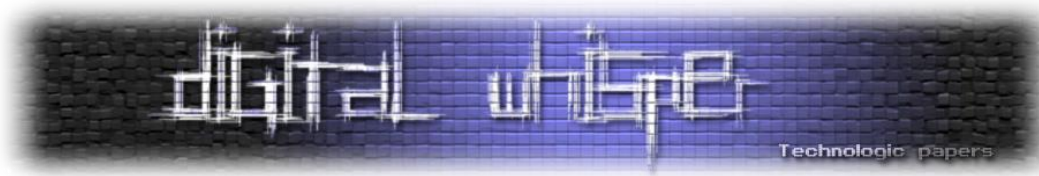
הדרך לפתרון:

1. כדי להתגבר על הבעיה עלינו לשמור את הקוד הפנימי שלנו כמחרוזת. בשביל זה יצרנו רמה שלישית של פונקציות, כאשר הכי פנימית מחזירה את המחרוזת של הקוד (כמו בצורה מהדוגמה הראשונה).
2. עכשיו אחרי שפתרנו את הבעיה הראשונה יצרנו בעיה חדשה, אין ב-javascript תמיכה במחרוזות המתפרשות על יותר משורה אחת, לכן במצב הנוכחי לא נוכל לעצב את הקוד שלנו בצורות. כדי לפתור את הבעיה נוסיף בסוף כל שורה סלאש-אחורי (\), רק כך ה-javascript מאפשר למחרוזת להתפרש על כמה שורות.
3. אך עדיין לא סיימנו, ברגע שנריץ את הקוד הפנימי נזהה שחסרים לנו שוב הסלאשים בסוף. הסלאשים לא נחשבים חלק מהמחרוזת ולכן כאשר המחרוזת נשמרת למשתנה, ה"רידות שורה" עדיין יהיו שם אבל הביטוי לא יהיה ביטוי תקין, שכן אסור לרדת שורה באמצע מחרוזת ב-javascript. בעצם במקום ירידת שורה וסלאש בסוף תהיה לנו רק ירידת שורה, כמו במצב 2.
4. את הבעיה האחרונה נפתור ע"י הוספת פונקציה פנימית שתדאג להוסיף כל פעם את סלאש בסוף השורה. אני לא אכנס למימוש של הפונקציה לאור העובדה שאני לא בדיוק זוכר מה הולך שם, נועם ואני כתבנו אותה ב-4 בבוקר. בכל זאת אזכיר שאנחנו בפונקציה ברמה 4 (בתוך 3 פונקציות קודמות) לכן כל עניין ה-escaping נהיה מסובך. אם נרצה לכתוב את התו \ נצטרך בפועל לכתוב אותו פעמיים עבור הרמה הראשונה ו-4 פעמים עבור הרמה השניה, 8 פעמים עבור הרמה השלישית וחוזר חלילה.

את הקוד לתמונה הסופית אפשר להשיג כאן. אתם מוזמנים להציץ בפרויקט ה"קומפיילר" ב-github. אני מקווה שנהניתם מהמאמר. לסיום, מצאתי חידה חביבה מרחבי האינטרנט, אני לא זוכר את המקור: תנסו להכניס למשתנה a ערך ככה שהחלון יקפוץ. בהצלחה!

```
a = <PUT_VALUE_HERE>;

// if something is not equal to itself =]
if (a !== a){
    alert("you win");
}
```



---

# חלק ג' - Hacking Games For Fun And (mostly) Profit

מאת d4d

---

## הקדמה

מטרת סדרת מאמרים זו הינה להציג את השלבים שעברנו בעת מחקר המשחק Worms World Party, במטרה לכתוב שרת פרטי למשחק זה. עד כה הצגנו את שלבי הקמת המעבדה לטובת ביצוע המחקר, ואף את שלבי המחקר המתקדמים:

- את ההצפנה בה מפתחי WWP השתמשו בכדי לבצע אימות משתמשים לשרת ה-IRC.
- את השלבים ואת תהליך הרברסינג למנגנון שבתוך המשחק בכדי לזייף את ה-Challenge Response.
- ניתחנו את שיטת ההצפנה שבה השתמשו לרשימת המשחקים והצגנו קוד שיודע להציג את המשחק ללא הצפנה.
- דיברנו על איך נראה מבנה המשחק ב-WWP בזיכרון.

מאמר זה הינו החלק השלישי של סדרת מאמרים זו, מאמר זה מדבר על הנושאים הבאים:

- תיאור הפרוטוקול של WWP.
- יצירת אמולטור לשרת WormNET2.

## תיאור הפרוטוקול

הפרוטוקול של WWP דומה ל"פקודות" HTML ליצירת דפי אינטרנט, אך הפקודות מעט שונות. כדי למצוא את כל הפקודות הקיימות ב-WWP הסתכלנו ב-IDA Pro על הפונקציה שבה בודקים את סוג הפקודה שהתקבלה.



להלן צילום מסך של קטע מפונקציה זו:

```

; Attributes: bp-based frame

sub_4327DC proc near

var_14= dword ptr -14h
var_10= dword ptr -10h
var_C= dword ptr -0Ch
var_8= dword ptr -8
var_4= dword ptr -4
arg_0= dword ptr 8

55          push    ebp
8B EC      mov     ebp, esp
83 EC 14   sub     esp, 14h

68 90 70 5E 00 push   offset Str2      ; "<SHOWLOGIN>"
68 E0 EA 62 00 push   offset word_62EAE0 ; Str1
FF 15 28 77 5B+call  ds:_stricmp
83 C4 08   add     esp, 8
85 C0     test   eax, eax
75 1B     jnz   short loc_432814
    
```

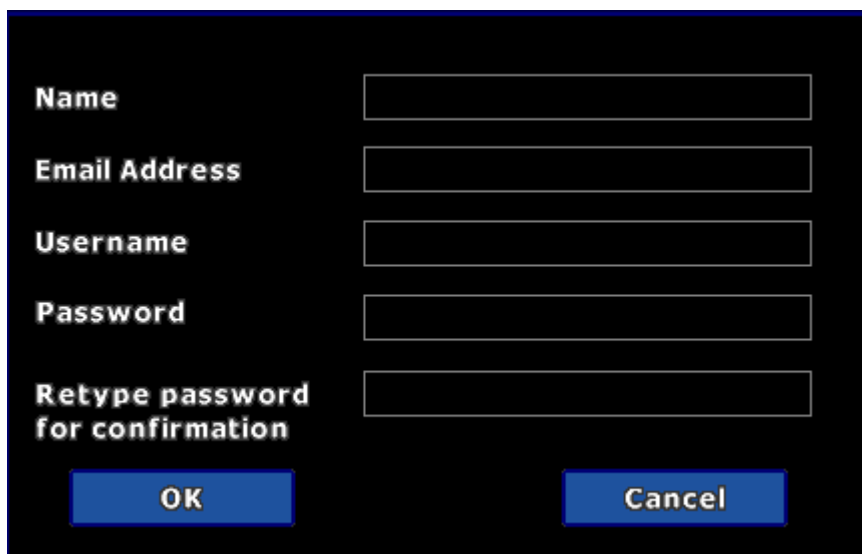
בדוגמא זו ניתן לראות השוואה לפקודה <SHOWLOGIN>, במידה וזו לא הפקודה שהתקבלה - הפונקציה תעבור לבצע השוואה עם הפקודה הבאה.

בכדי להגיע למסקנה מה כל פקודה עושה השתמשנו בסניפר (WireShark) כדי לבדוק מתי יש שימוש בפקודות החשובות. רשימת הפקודות החשובות הן:

- הפקודה <SHOWLOGIN> תציג במשחק את החלון הבא:



- הפקודה <SHOWNEWUSERENTRY> משמשת ליצירת משתמש חדש:



חלק מהשדות שמופיעים בתהליך ההרשמה אינם בשימוש באמולטור. נראה שמפתחי המשחק "התבלבלו" והם שולחים פרטים שלא מופיעים בהרשמה כלל.

כאשר נשלחת בקשת ה-GET לשרת, על מנת לבצע את ההרשמה נשלחים הארגומנטים הבאים:

```
[Username] => d4d
[Password] => 12345
[Email] => sdsdsd
[Surname] => abcddf
[Address] =>
```

- הפקודה לביצוע חיבור לשרת ה-IRC הינה הפקודה הבאה:

```
<CONNECT [IRCServer] IRCPORT=[port] IRCUSER=[user] IRCPASS=ELSLRACLHP>
```

שדה	תיאור
IRCServer	כתובת לשרת IRC
IRCPORT	הפורט
IRCUSER	שם משתמש
IRCPASS	סיסמא שניתן לשנות בפקודה



- הפקודה לביצוע Challenge Response היא הפקודה: ANSWER אותה ראינו בחלק א' והרחבנו עליה בחלק ב':

```
<ANSWER CVttpTpl5cd7dP+0Ae1WIr71jqpXllw2jXE1qmAyQb0gEwFZ>
```

- הפקודות להצגה של רשימת המשחקים הם הפקודות הבאות:

```
<GAMELISTSTART>
<GAME ... >
<GAMELISTEND>
```

תיאור	פקודה
פקודה המציגה את תחילת רשימת המשחקים	GAMELISTSTART
פקודה המציגה את סוף רשימת המשחקים	GAMELISTEND
כל משחק מופיע בפקודה חדשה והוא חייב להיות מוכל בין GAMELISTSTART ל GAMELISTEND	GAME

- הפקודה WEBADDRESS הינה הפקודה בה קובעים את התיקייה של השרת. לדוגמא:

```
<WEBADDRESS /test/>
```

בביצוע פקודה זו השרת ייגש לקבצים הנמצאים בתיקייה test, בכתובת זו: http://myurl.com/test.

- הפקודה SCHEME מתבצעת כאשר אנו נכנסים לערוץ בשרת. הסבר מפורט על הפרמטרים שמקבלת הפקודה SCHEME ניתן למצוא באתר [http://worms2d.info/WormNET\\_\(Worms\\_Armageddon\)](http://worms2d.info/WormNET_(Worms_Armageddon)) משום שקיים מידע על מה קורה ב-WA זה חסך קצת עבודה לבדוק מה כל מוד בפקודה זו עושה.

ישנן פקודות נוספות למשחק, אך הן לא חשובות לאמולטור שלנו.

בחלק א' הזכרנו שהיו דרגות ב-WA ושחברת Team17 הורידו אותן, בתחילה תכננו לעשות שרת שיהיה בו מימוש גם לדרגות ב-WWP אך החלטנו לא להפיץ אותו מהסיבות שיפורטו בחלק הבא.



## דרגות - אבטחת מידע

מנגנון הדרגות אינו מאובטח כלל, כל משתמש שיזייף פאקט ששולח ניצחון יקבל אותו, גם אם המשחק לא בוצע כלל. על מנת שהדרגות יעבדו בצורה שלא יהיה קל לכל ילד בן 10 לרמות צריך לשנות את המודול. הבעיה במודול של חברת team17 היא הבעיה הבאה:

team17 סומכים על המשתמש שישלח את התוצאה של מי ניצח לשרת. מי שולח את הבקשה? המשתמש שיצר את המשחק, הדבר דומה לקזינו שבו בעל הבית תמיד מנצח ולכן החלטנו לא לפרסם את המימוש לדרגות.

### למה עשו אז הצפנה ל-WWP?

הסיבה שעשו את ההצפנה זה בכדי להקשות על משתמשים לא מורשים להיכנס לשרת משחק בלי עותק של המשחק ושלא יוכלו לקרוא את רשימת המשחקים. הסיבה שהשתמשו בהצפנה של RSA בכדי להצפין את המשחקים הייתה דרך טובה להקשות על הקמת שרת פרטי בקלות.

בכדי ליצור משחק אנו צריכים להחליף את המפתחות RSA של השרת על ידי מודול שנכתב שתפקידו לשנות את המפתח הפומבי שנמצא בקליינט (בשרת יהיה מפתח private שבעזרתו נחתום את המידע שמוצפן ב-Twofish)

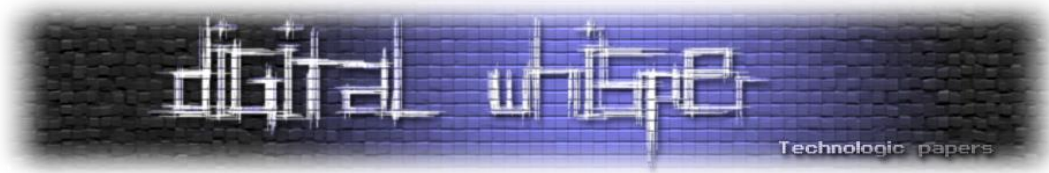
המפתחות של ה-Twofish לא שונים, המפתחות זהות בהצפנה סימטרית אז אין טעם לשנות גם אותם.

### יצירת אמולטור ל-WWP

לאחר שניתחנו את ההצפנה שהייתה בשימוש ב-WWP שנתנה לנו את היכולת להבין איך לפענח את רשימת המשחקים ולפתור את ה-Challenge Response לשרת IRC בלי עותק של המשחק הגענו לשלב המתבקש שהוא יצירת האמולטור.

כדי להבין את שאר החלקים החסרים ב-WWP היה שימוש ב-WireShark כדי לבדוק איפה להשתמש בכל פקודה.





ניתן לראות שהשרת מחזיר למשתמש הידר בשם SetGameId בשביל ליצור משחק, אם ה-Header הנ"ל לא יופיע, המשחק לא ייווצר.

כאשר כותבים אמולטור לשרת פרטי, צריך לנסות לשחזר את מה שמקבל השרת המקורי, ובמידה ומשהו לא עובד צריך להשוות את הקוד שלנו עם השרת המקורי או לקרוא בקוד איך המשחק ניגש למידע מסויים.

לדוגמא כאשר אנחנו מנסים להיכנס לערוץ, אם יהיה חסר לנו הדף RequestChannelScheme.php המשחק ייתקע, אנו חייבים להחזיר לו תשובה, אז במידה ולא הוגדר סוג של ערוץ אנו נקבע את הערך הדיפולטיבי:

```
<SCHEME=Pf, Be>
```

כדי לדעת איזה ארגומנטים הקליינט שולח לשרת כתבנו סקריפט ב-PHP שידפיס לתוך קובץ בשם logs.txt את כל מה שנשלח ב-GET על ידי קטע הקוד הבא:

```
$fp = fopen('logs.txt', 'a');  
fwrite($fp, "Login.php\r\n");  
fwrite($fp, print_r($_GET, TRUE));  
fclose($fp);
```





## סקיצה בסיסית של טבלאות מה-Database

כעת אנו צריכים ליצור Database שישמור את כלל הנתונים. ה-Database צריך להכיל את הדברים הבאים:

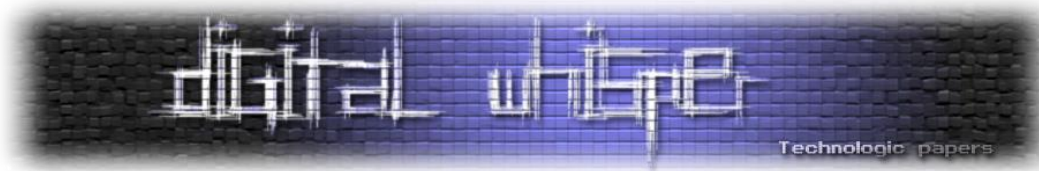
- נתונים על המשתמשים בשרת.
- רשימת המשחקים בכל הערוצים הקיימים בשרת.
- רשימת כל הערוצים שיש בשרת וסוג הערוץ.
- במידה ויהיו דרגות הצטרכו להוסיף את הנתונים מי ניצח והפסיד.

**הטבלה USERS** - טבלה זו שומרת פרטים על המשתמש, כגון ניקוד ובדיקה אם המשתמש חסום או לא לפני כניסה לשרת.

שם שדה	תיאור
UserID	מפתח ראשי לזיהוי השם משתמש
PlayerType	סוג מד החיים אשר יופיע למשתמש בזמן המשחק. אם הערך יהיה 2 למשתמש תיהיה אש כחולה. אם הערך הוא 1 תופיע למשתמש אש אדומה.
PlayerScore	נקודות שיש לכל משתמש, אם אין דרגות ניתן לוותר על השדה.
PlayerLevel	הדרגה שתהיה למשתמש בערוץ. לדוגמא, הערך: 5 יציג את הדרגה: 
Banned	במידה ונרצה לחסום את המשתמש. ערך זה יכול: 1, אחרת: 0.
UserName	הכינוי של המשתמש (שדה זה מוגדר כ-UNIQUE).
Password	הסיסמא של המשתמש
IPAddress	כתובת ה-IP של המשתמש
Email	המייל של המשתמש (שדה זה מוגדר כ-UNIQUE)
Access	1 במידה ולמשתמש יש גישה לפאנל ניהול של המערכת

**הטבלה HostGames** - בטבלה זו נשמרים כל המשחקים שנוצרו בשרת:

שם שדה	תיאור
GameID	מפתח ראשי לזיהוי השם משתמש
UserID	ה-ID של המשתמש
GuestID	ID שנקבע במשחק עם דירוג, ניתן לוותר על השדה הזה אצלנו
Nick	השם של יוצר המשחק
HostIP	כתובת ה-IP של יוצר המשחק
Name	השם של המשחק



Loc	המדינה של המשתמש
Type	המספר יכיל 0, במוד של דרגות, מכיל מספר 1-4
ServerUsage	1 אם ניתן ללחוץ על המשחק, 0 אם אי אפשר.
Chan	הערוץ שבו נוצר המשחק

**הטבלה EncryptedHosts** - בטבלה זו נשמרים כל המשחקים שנוצרו ואחרי 2 דקות הם נמחקים מהרשימה ולא ניתן להיכנס למשחק.



שם שדה	תיאור
Id	מפתח ראשי לזיהוי המשתמש המוצפן
EncryptedGame	המידע של המשחק עם הצפנה ומקודד ב-base64 כפי שהוסבר בחלק ב'
GameID	ה-ID של המשחק
Channel	הערוץ בו המשחק נוצר
Time	זמן שבו נוצר המשחק

**טבלה SCHEMES** - בטבלה זו נשמרים המודים לכל ערוץ שקובעים את סגנון המשחק.

שם שדה	תיאור
Id	מפתח ראשי לזיהוי הערוץ
Channel	השם של הערוץ (שדה זה מוגדר כ-UNIQUE)
Modes	המודים שקובעים את סגנון המשחק

על מנת לנהל את השרת עצמו, כתבנו פאנל ניהול, הוא נראה כך:

Set Channel modes

 main
 view

Channel name:

Set channel modes:

- Amount of blood a ▼
- Rope pushing power level a ▼
- Set ranked channel a ▼
- Set minimum rank restriction a ▼
- Set maximum rank restriction a ▼
- Worms per team b ▼
- Force Scheme
- Special weapons

בחלק זה אנו קובעים את המודים בהם הערוץ שלנו יתמוך, המוד לדרגות גם מופיע אך נתונים של מי ניצח לא יישמרו, כי ה-database לא טורח לשמור אותם. בתמונה הבאה מופיעים סוג ותיאור המודים:

Channel modes information

**AnythingGoes**  
Pf,Be

**PartyTime**  
Ba

**RopersHeaven**  
Pf

a-z is 0 to 25 unless otherwise stated.

Code	Arguments	Effect
B	a-e	Amount of blood.
D	a-z × ...	String of enforced game/weapon options.
G	a-z × ...	String of super-weapon options.
N	(text)	Set the scheme name that will appear.
P	a-k	Set rope pushing power level, defaults to 0, f is normal.
R	a-m a-m	Set rank restriction (minimal and maximal rank)
T	a-f	Hosting: 0=allowed, 1=ranked, 2=ranked, 3=ranked, 4=ranked, 5=disallowed This is also used as the type value of the game when reporting the result.
W	b-i	Worms per team, counting from b=1. If adding a team with this many would exceed the worm limit, that team will be added with less worms. Any values out of range count as b (1).



Special weapons information			
a-z is 0 to 25 unless otherwise stated.			
k in Amount for unlimit weapons			
k in delay for disable weapon unless otherwise stated			
Code	Amount	Delay	Effect
ha	a-k	a-b	Crate shower, for unlimit Crate shower turns it would be set to k.
fs	a-b	a-k	Crate spy, how much turns Crate spy would be set, k for always.
fq	a-k	a-k	Invisibility.
eq	a-k	a-k	Armageddon.
cq	a-k	a-k	Girder pack.
fb	a-k	a-k	Carpet bomb.
dy	a-k	a-k	Donkey.
ba	a-k	a-k	Earth quake.
ga	a-k	a-k	Freeze.
gb	a-k	a-k	Magic bullet.
dq	a-k	a-k	MB bomb.
da	a-k	a-k	Mine strike.
dt	a-k	a-k	Ming vase.
db	a-k	a-k	Mole Squadron.
dz	a-k	a-k	Nuclear Test.
bt	a-k	a-k	Mail strike.
cz	a-k	a-k	Salvation Army.
eb	b-k	a-k	Scales Of Justice.
et	a-k	a-k	Select worm.
fa	a-k	a-k	Sheep strike.
bj	a-k	a-k	Suicide Bomber.
ec	a-k	a-k	Super banana bomb.

כל 2 אותיות מרכיבות נשק מיוחד אותו לא ניתן לקבוע דרך המשחק. שתי אותיות האלה מחשבות את האינדקסים במבנה שקובע את הנשקים במשחק, במקום לנחש אינדקסים נכתב קוד בפייתון שימצא לנו את הצירוף אותיות הנכון שיתן לנו את הנשק המבוקש. הקוד יצורף בסוף המאמר.

בפאנל ניהול נכתב כלי שיודע להמיר לקודים הדרושים את הנשקים שאנו מבקשים וידע להביא את הנשקים. דוגמא לבחירת חלק מהנשקים בפאנל:



בשביל שהשרת יעבוד כמו שצריך היינו צריכים להקים שרת IRC אליו הוא יוכל להתחבר, בגדול כל שרת IRC מתאים אך רצינו להכניס לשרת שלנו את ה-Challenge Respons כמו שעשו בחברת team17. אנו השתמשנו ב-hybrid-ircd בתור השרת IRC שלנו והוספנו לשם מודול ושינינו כמה core files.

אחד המפתחים הראשיים של הפרויקט הסכים לתת לנו תמיכה ולומר איזה קבצי core כדאי לשנות.

הסיבה שלא השתמשנו ב-unrealIRCd זה בגלל שאין תמיכה למי שמעוניין לכתוב מודולים, ניסינו לפנות אליהם במשך חודשיים ואף אחד לא היה זמין.

השרת IRC לא צריך Challenge Response, אך אם כבר מבינים איך המנגנון עובד אז אין טעם לא להוסיף את האופציה הנ"ל. זה רק משפר את האבטחה וכמובן לנסות להיות כמה שיותר קרוב למקור. אנו ערכנו את קבצי ה-core הבאים ב-hybrid-ircd:

- client.c
- user.c
- client.h

ב-client.h נוספו הדגלים הבאים, בהם יש שימוש למימוש המודול שלנו m\_authpong.c

```
#define SetAuthPing(x) ((x)->flags |= FLAGS_AUTHPING_SENT)
#define HasAuthPing(x) ((x)->flags & FLAGS_AUTHPING_SENT)
#define ClearAuthPingSent(x) ((x)->flags &= ~FLAGS_AUTHPING_SENT)
```

ב-client.h הוספנו עוד שדות למבנה Client בשם:

```
char *authping;
char *url;
```

ב-client.c הוספנו את השדות שיש לשחרר להם את הזיכרון בפונקציה free\_client():

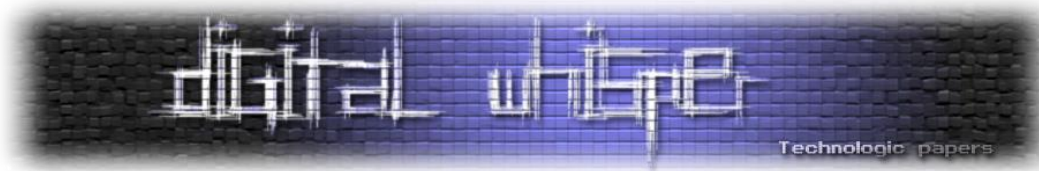
```
MyFree(client_p->authping); // added authping to free
MyFree(client_p->url); // added
```





ב-c.user הוספנו בפונקציה register\_local\_user את הקטע קוד הבא:

```
if(!HasAuthPing(source_p))
{
    char url[5][100] = {"wormnet.team17.com$1",
"wormnet.team17.com$2", "wormnet.team17.com$3", "wormnet.team17.com$4",
"wormnet.team17.com$5"};
    uint8_t key[] =
"\xD5\x2D\x31\x08\FB\x0F\x54\x9E\x6D\x7A\x0F"
"\xFD\xEE\xDC\x21\x9A\xD4xA6\x84\x24\x6D\x61\xFA\x8A\xAE\x98\x96xA1\xF
1\x63\x1A\x8D";
    uint8_t text[100];
    uint8_t iv[16];
    uint8_t hash[20];
    // choose the seed for the user
    memset(iv, '\0', 16);
    srand(time(NULL));
    for(int i = 0;i < 16;i++)
        iv[i] = rand() % 0xFF;
    // choose the secret value for the challenge response
    srand(time(NULL));
    int pickUrl = rand()%5;
    source_p->authping = MyCalloc(100);
    source_p->url = MyCalloc(100);
    // produce the encryption and hash it
    memcpy(text, iv, 0x10);
    memcpy(text + 0x10,url[pickUrl], strlen(url[pickUrl]));
    MCRYPT td = mcrypt_module_open("twofish", NULL, "cfb", NULL);
    mcrypt_generic_init(td, key, 0x20, iv);
    mcrypt_generic(td, text, 36);
    mcrypt_generic_deinit(td);
    mcrypt_module_close(td);
    computeRIPEMD160(text, 36, hash);
    memset(text, '\0', 100);
    for(int i = 0;i < 20;i++)
        sprintf((char*)text + i*2, "%02X", hash[i]);
    memcpy(source_p->url, url[pickUrl], strlen(url[pickUrl]));
    memcpy(source_p->authping, text, strlen(text));
    sendto_one(source_p, "AUTHPING %s %s ", source_p->url, source_p-
>authping);
    return;
}
```



ב-WWP יש אפשרות ליצור ערוצים עם סיסמאות ככה שרק אנשים ספציפיים עם סיסמא יוכלו להיכנס, זה פיצר אותו אין ב-WA (אין אפשרות להכניס סיסמא בכלל בקליינט). במידה ולא קיבלנו Response מהמשתמש על ה-Challenge הוא מתנתק מהשרת אחרי חצי דקה.

בנוסף, ערכנו גם את ה-topic שלא יציג את המודים בכותרת בגלל שב-WWP/WA המספרים בין 00 ל-06 קובעים את האייקון של הערוץ במשחק, במידה ולא יוצגו בהתחלה מספרים אלה, ייקבע כברירת מחדל האייקון 06. הקוד ל-m\_authpong.c יצורף בסוף המאמר.

## סיכום

ביצוע מחקר לטובת הקמת שרת פרטי למשחק מחשב רציני זו בהחלט לא עבודה קלה. ראינו כי ראשית יש צורך לבצע ניתוח להצפנה שיש בפרוטוקול איתו מדברים השרת והלקוח. רק לאחר שמבינים את ההצפנה ויודעים איך לפענח את התקשורת ניתן לגשת לשלבים האחרים שהם ניתוח הפרוטוקול עצמו.

לפני המימוש של השרת, ישנו הצורך לכתוב Packet Logger שידע לקחת את כל חבילות המידע שהמשחק שולח לשרת ואז לנסות לסווג אותם לפי סוג ותפקיד, בכל משחק זה שונה.

במאמר זה הראינו את השלבים הבסיסיים אותם צריך בשביל לממש שרת פרטי, קודם הקמנו את סביבת העבודה וכתבנו קבצי DLL שיאיצו את העבודה, ניתחנו את ההצפנה והבנו איך היא פועלת, לאחר מכן בדקנו עם Sniffer את שאר הפקודות בשרת מה שלא היה ידוע מהגירסה הקודמת WA. ולבסוף כתבנו את הקוד של השרת וביצענו טסטים לראות שהקליינט מגיב למה שהשרת שלנו שולח.

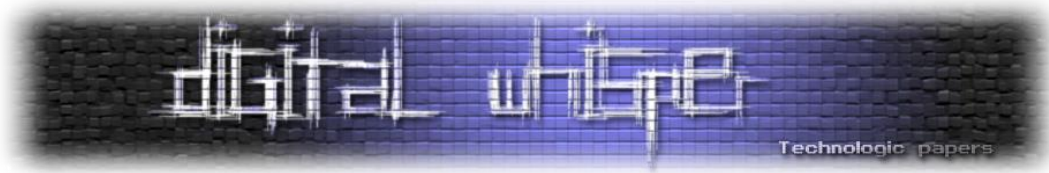
אני מקווה שנהנתם מסדרת מאמרים זו ושלמדתם ממנה רבות. הקוד המלא לשרת יפורסם ב-github וב-bitbucket בהמשך, כרגע סגור מכיוון שאנו מעוניינים להוסיף עוד מספר פקודות. את הקוד שהוצג בחלק זה ניתן להוריד מהקישור הבא:

<http://www.digitalwhisper.co.il/files/Zines/0x3B/WWP3.rar>

## על המחבר

d4d עוסק ב-Reverse Engineering ואוהב לחקור משחקי מחשב והגנות. לכל שאלה או יעוץ ניתן לפנות אליו בשרת ה-IRC של NIX, בערוץ: [#Reversing](https://www.twitch.tv/Reversing). בכתובת האימייל: [llcashall@gmail.com](mailto:llcashall@gmail.com). או דרך

האתר: <http://www.cheats4gamer.com>



---

## דברי סיכום

---

בזאת אנחנו סוגרים את הגליון ה-59 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](mailto:editor@digitalwhisper.co.il).

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

[www.DigitalWhisper.co.il](http://www.DigitalWhisper.co.il)

*"Talkin' bout a revolution sounds like a whisper"*

הגליון הבא ייצא ביום האחרון של חודש מרץ 2015.

אפיק קסטיאל,

ניר אדר,

28.02.2015