זיהוי התקפות סייבר בתקשורת בזמן אמת – NetSpect

מאת: שי חחיאשוילי ומקסים סובוטין

בהנחיית: גב' אלונה קוציי

SE-A-14

המכללה האקדמית להנדסה ע"ש סמי שמעון

מהנדסים לעולם טוב יותר!

בסביבת PROJECT ORIENTED

ספר הפרויקט

שם הפרויקט: זיהוי התקפות סייבר בתקשורת בזמן אמת - NetSpect

מנחת הפרויקט: גב׳ אלונה קוציי

שמות הסטודנטים:

שם: מקסים סובוטין

שם: שי חחיאשוילי

תאריך התחלה: 01/11/2024

תאריך סיום משוער: 23/06/2025

תאריך עדכון אחרון: 15/06/2025

קישור ל- GitHub Repository של הפרויקט
קישור לסרטון Demo של הפרויקט

תוכן עניינים

תקציר

NetSpect - detecting cyber attacks in network traffic in real time

SE-A-14

Shay Hahiashvili; shayha2@ac.sce.ac.il

Maxim Subotin; maximsu@ac.sce.ac.il

Advisor: Ms. Alona Kutsyy

SCE - Shamoon College of Engineering, Be'er-Sheva

In the modern era, the dependence on technology and the internet is growing, and cyber attacks in the field of network communications pose serious risks to businesses and users, and can lead to data theft, service disruptions, and financial losses. As part of the project, we developed a real-time Intrusion Detection System (IDS) that monitors network traffic, detects attack patterns, and issues alerts. The hybrid system integrates cyber attack detection algorithms with machine learning models to identify anomalies. It detects threats like Port Scanning, DoS, ARP Spoofing, and DNS Tunneling. Since existing datasets were ineffective for real-time detection, we manually collected optimal training data. Our system ensures high accuracy with minimal false alarms and features an intuitive and simple interface.

Keywords: ARP Spoofing, Denial of Service, DNS Tunneling, Intrusion Detection System, Machine Learning, Port Scanning

זיהוי התקפות סייבר בתקשורת בזמן אמת - NetSpect

SE-A-14

שי חחיאשוילי; shayha2@ac.sce.ac.il
מקסים סובוטין; maximsu@ac.sce.ac.il

בהנחיית: גב' אלונה קוציי

SCE - המכללה האקדמית להנדסה ע״ש סמי שמעון, באר שבע

בעידן המודרני, התלות בטכנולוגיה ובאינטרנט הולכת וגוברת, ומתקפות סייבר בתחום התקשורת מהוות סיכון משמעותי לעסקים ולמשתמשים פרטיים כאחד. מתקפות אלו עלולות להוביל לגניבת נתונים רגישים, שיבושים בשירותים ונזקים כלכליים חמורים. עם למעלה מ-16 מיליארד מכשירים המחוברים לאינטרנט, הצורך בפתרונות חדשניים לזיהוי איומי סייבר בזמן אמת על מערכות התקשורת קריטי מתמיד. במסגרת הפרויקט פיתחנו מערכת זיהוי חדירה IDS – Intrusion Detection System, אשר מנתרת את תעבורת התקשורת בזמן אמת, מזהה דפוסי התקפות תקשורת שונות, ומתריעה עליהם בזמן אמת. המערכת שלנו היא מערכת היברידית, אשר משלבת בין אלגוריתמים לזיהוי התקפות סייבר לבין מודלי למידת מכונה (ML) אשר מזהים דפוסי תקשורת חריגים. בזכות כך, המערכת מסוגלת לזהות את סוגי מתקפות הסייבר הנפוצות ביותר כיום, בהן, Port Scanning, Denial of Service (DoS), ARP Poisoning ו- DNS Tunneling. כחלק מלימוד המודלים, מצאנו כי הנתונים שמחקרים מתבססים עליהם כיום לא יעילים עבור זיהוי מתקפות בזמן אמת, ולכן מצאנו Feature Selection אופטימלי יותר ממה שקיים במחקרים כיום. ולכן, לימוד המודלים התבצע על הנתונים שאספנו באופן ידני, הנתונים מכילים תעבורת תקשורת תקינה וגם תעבורת התקפות שונות. המערכת שפיתחנו מספקת יכולות זיהוי בזמן אמת עם שיעור מינימלי של התראות שווא, ובנוסף, היא כוללת ממשק משתמש אינטואיטיבי וקל לשימוש המאפשרת למשתמש לראות את היסטוריית ההתראות בצורה ברורה ומסודרת.

מילות מפתח: ARP Spoofing, Denial of Service, DNS Tunneling, Intrusion Detection System, Machine Learning, Port Scanning

רקע ומוטיבציה

רקע

מתקפות סייבר בתקשורת מציבות סיכון משמעותי לעסקים ולפרטים בעידן המודרני, שבו התלות בטכנולוגיה ובאינטרנט הולכת וגוברת. הסיכונים[1] כוללים גניבת נתונים רגישים, כמו פרטי כרטיסי אשראי ומידע אישי, מה שעלול להוביל להונאות ולנזקים כלכליים. שיבוש שירותים, כמו מתקפות DoS - Denial of Service, יכול להשבית אתרי אינטרנט ושירותים מקוונים, מה שגורם לאובדן הכנסות ופוגע באמון הלקוחות. בנוסף, התקפות כמו DNS Tunneling מאפשרות הפצת תוכנות זדוניות כמו וירוסים ורוגלות יכולה לפגוע בתפקוד של מערכות ולחשוף מידע אישי, דבר שפוגע בפרטיות המשתמשים.

עם יותר מ-16 מיליארד מכשירים המחוברים לאינטרנט[2] בשנת 2024, חשיבותן של מערכות כגון מערכות זיהוי חדירה כמו IDS[3] (Intrusion Detection System) ו- EDR (Endpoint Detection and Response), היא קריטית לצורך הגנה על מידע רגיש ושמירה על תפקוד תקין של מערכות תקשורת. מערכות אלו מיועדות לנטר את תעבורת התקשורת בזמן אמת ולהצביע על דפוסי התקפות סייבר.

מערכת IDS מתחלקות לשלושה סוגים עיקריים והם Signature-Based IDS Anomaly-Based IDS ו- Hybrid IDS. השוני ביניהם הוא שמערכות Signature-Base מתמקדות על זיהוי תפוסים חריגים בתקשורת על ידי אלגוריתמים מורכבים המנטרים את תעבורת התקשורת בזמן אמת ומנתחים את הנתונים שהם רואים, ובכך מזהים דפוסי התקפה ידועים. לעומת זאת, מערכת Anomaly-Based שונות בכך שהם מחפשות אנומליות בתעבורת התקשורת ובכך מסוגלות לזהות גם התקפות ידועות וגם התקפות לא ידועות (zero day), כלומר התקפות שעדיין לא ידוע תפוס ההתקפה שלהם. ואילו מערכות Hybrid IDS משלבות את שני השיטות ובכך הם יותר אפקטיביות.

בעוד שבעבר מערכות Anomaly-Based IDS התבססו על אלגוריתמים מסורתיים כמו עצי החלטה ו- KNN. כיום (Laghrissi F et al. 2021, 2-5) נעשה שימוש נרחב במודלים של למידה מכונה כמו RNN, SVM ו- LSTM. מודלים אלו מסוגלים להתמודד

[1] Top 15 types of cybersecurity attacks and how to prevent them
[2] State of IoT 2024: Number of connected IoT devices growing 13% to 18.8 billion globally
[3] Host based intrusion detection systems, how do they work

עם כמות גדולה של נתונים ולסווגם במהירות , הודות ליכולות שלהם לאיתור תבניות מורכבות, מה שהופך אותם לאידיאליים להגנה על מערכות מחשב מפני התקפות שונות.

עם זאת, השימוש במודלים של למידה מכונה הוא תהליך מורכב, הדורש הרבה נתונים מדוייקים המצביעים על תבניות של התקפות ידועות. על מנת להשיג נתונים מדוייקים, פותחו שיטות ותהליכים כמו הקטנת ממדים (dimensionality reduction) ובחירת מאפיינים (feature selection) המסייעים להוריד את עומס הנתונים ולהתמקד במידע החשוב ביותר לאימון המודל (Laghrissi F et al. 2021, 5-6), מה שמוביל לביצועים מהירים ומדויקים יותר של המערכת ומאפשר למודל לזהות דפוסים קריטיים בנתונים.

מוטיבציה

ההתקדמות הטכנולוגית לזיהוי ותגובה מהירים למתקפות סייבר עשויה למנוע נזקים משמעותיים ולתרום למניעת אירועים מזיקים בעתיד. ככל שהעולם נעשה דיגיטלי יותר, כך גם הצורך במערכות יעילות לזיהוי ותגובה מתקדם הופך קריטי. השקעה בהכשרת עובדים, הגברת מודעותם לאיומי סייבר, ופיתוח פתרונות טכנולוגיים מתקדמים יחד יכולים לשפר את היכולת להגן על מערכות קריטיות ולהפחית את הסיכונים הנלווים.

נוסף לכך, ההתקדמות המתמשכת בתחומי הסייבר[4] יוצרת צורך גובר במודלים ותוכנות המסוגלים לזהות מתקפות בזמן אמת ולהגיב ביעילות. לכן המעבר לפיתוח מערכות Hybrid IDS (מערכות משולבות) מאפשר לשלב בין גישות שונות, כמו זיהוי מבוסס חתימות (Signature-Based), זיהוי מבוסס חריגות (Anomaly-Based) ושימוש בטכניקות של למידת מכונה. מערכות אלו מציעות איזון אידיאלי בין זיהוי התקפות ידועות בצורה מדויקת לבין יכולת ללמוד ולזהות התקפות חדשות או לא מוכרות. שילוב טכניקות למידת מכונה מאפשר למערכות להתאים את עצמן לדפוסים משתנים, לשפר את יכולת הסיווג והתגובה, ולהציע שכבת הגנה מתקדמת ומקיפה יותר לעומת שיטות מסורתיות.

לכן, קיים הצורך בפיתוח תוכנות מתקדמות לזיהוי התקפות סייבר בזמן אמת, תוך שימוש בכלים וטכנולוגיות חדשות ומתקדמות[5] על מנת להגן על מערכות מפני איומי סייבר פוטנציאלים במהירות ועם רמת דיוק גבוהה.

[4] The Evolution of Cyber Threats: Past, Present and Future
[5] Advancements in artificial intelligence and machine learning

סקר ספרות

הקדמה (רקע)

מתקפות סייבר על מערכות תקשורת מהוות איום משמעותי לעסקים וליחידים בעידן הדיגיטלי שבו התלות בטכנולוגיה ובאינטרנט גדלה במהירות. סיכונים אלו כוללים גניבת מידע רגיש, כגון פרטי כרטיסי אשראי ונתונים אישיים, שעלולה להוביל להונאות ונזקים כלכליים משמעותיים. שיבוש שירותים חיוניים באמצעות מתקפות כמו DoS עלול להשבית אתרי אינטרנט ושירותים מקוונים, לגרום לאובדן הכנסות ולפגוע באמון הלקוחות. בנוסף, מתקפות כמו DNS Tunneling מאפשרות הפצת תוכנות זדוניות כמו וירוסים ורוגלות, אשר פוגעות בתפקוד המערכות וחושפות מידע אישי – דבר המהווה פגיעה חמורה בפרטיות המשתמשים.

הצורך במערכות מתקדמות לזיהוי מתקפות בזמן אמת, כמו IDS ו-EDR, הפך קריטי כדי להגן על נתונים רגישים. מערכות אלו עוקבות אחר תעבורת הרשת ומתריעות על איומים בזמן אמת. כיום במחקרים ובמאמרים אקדמיים, נפוץ לבנות ולפתח מערכות IDS המשתמשות במודלים כמו RNN, SVM ו-LSTM, לצורך זיהוי התקפות תקשורת (Laghrissi F et al. 2021, 2-5). מודלים אלו מאפשרים זיהוי מדויק של דפוסי התקפה מורכבים וחדשים.

ההתקדמות הטכנולוגית בתחום הסייבר והשימוש במודלים של למידה מכונה, דורשים תהליכים של הקטנת מימדים ובחירת מאפיינים חשובים, כדי להתמודד עם עומס נתונים ולהפיק תובנות קריטיות בזמן אמת (Laghrissi F et al. 2021, 1). יחד עם העלאת מודעות והכשרת עובדים בתחום, מערכות אלו יוכלו לספק רמת הגנה מתקדמת ולמנוע נזקים פוטנציאליים.

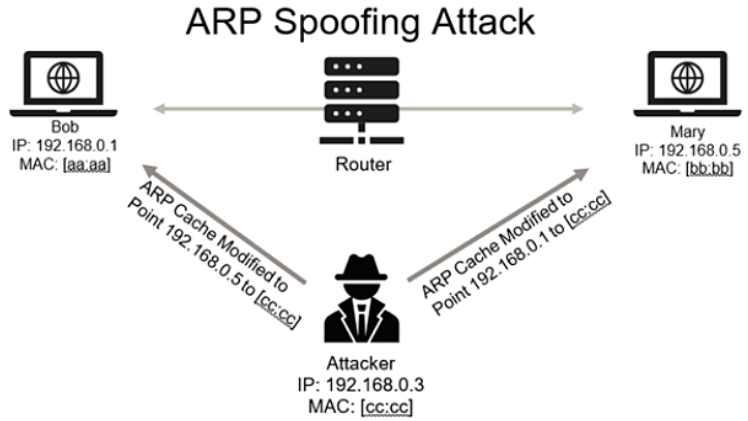
מטרת הפרויקט שלנו היא לפתח תוכנת Hybrid IDS שיודעת לזהות מתקפות סייבר נפוצות בתקשורת בזמן אמת תוך שימוש באלגוריתמי דפוסי חריגים ואלגוריתמי למידת מכונה פשוטים, אשר תוכל לזהות התקפות תקשורת ובכך תתרום למזעור נזקים בקרב המשתמשים.

התקפת ARP Spoofing

התקפה נפוצה על רשתות תקשורת לוקאליות LANs היא ARP Spoofing, זאת היא התקפת "אדם באמצע" (Man in the Middle[6]) אשר מתבצעת על ידי שליחת ARP Reply לכל המכשירים הקיימים ברשת הלוקאלית (LAN), כלומר כל מכשיר המקבל את הפאטקה הזו יודע ללכת לטבלת ה ARP הלוקאלית שלו ולעדכן את ה Mac-Address המופיע לו בטבלה ל- Mac-Address שקיבל ב ARP Reply. דבר זה גורם לכך שבמקום שכתובת ה-IP תיהיה משויכת לכתובת Mac המקורית, כעת היא תיהיה משויך לכתובת התוקף. בכך, התוקף מסוגל להאזין לכל התקשורת שעוברת באותם כתובות ה-IP שהושפעו על ידי ההתקפה.

התקפת ARP Spoofing מציבה סיכון משמעותי לרשתות תקשורת לוקאליות, שכן היא מאפשרת לתוקף להאזין, לשנות ואף למנוע תעבורת נתונים בין מכשירים מחוברים לרשת. כאשר המידע החשוב, כגון סיסמאות או נתונים רגישים, עובר דרך הרשת, התוקף יכול לתפוס אותו בקלות יחסית (Majumdar A, S, and T 2021, 1-2). בנוסף, ההתקפה יכולה להוביל להתקפות נוספות, כמו הפניית תעבורה לאתרים מזויפים, מה שמגביר את הסיכון להונאות ומעשי שוחד. בעידן שבו הגנת המידע חשובה יותר מתמיד, זיהוי מוקדם של ההתקפה יכול למזער את ההשלכות של מתקפת ARP Spoofing.

לצורך זיהוי של התקפת ARP Spoofing נדרש לבנות טבלת כתובות המכילה זוגות של כתובות IP ו- MAC, הטבלה הזאת נבנת על ידי שליחת ARP Requests לכל המכשירים ברשת הלוקאלית (Majumdar A, S, and T 2021, 8-9). לאחר מכן ניתן להשתמש בטבלה זו לצורך זיהוי של התקפות על ידי ניתוח חבילות ARP והצלבה בין כתובת ה-MAC שנשלחה בחבילה לבין הכתובת האמיתית הנמצאת בטבלה שלנו. במקרה של חוסר התאמה, המערכת תדע להתריע על מתקפה.



[6] Man In The Middle Attacks - IBM

התקפת Port Scanning

בעולם הסייבר, סריקות פורטים הפכו לכלי נפוץ בידי תוקפים. התוקפים משתמשים בטכניקות שונות כדי לסרוק את מערכת היעד ולאתר פורטים פתוחים שדרכם ניתן לחדור. קיימות הרבה טכניקות להתקפה זו אך לא משנה באיזו שיטה נוקט התוקף, סריקת פורטים היא אמצעי יעיל שמסייע לו להעריך את נקודות תורפה במערכת היעד.

אחת השיטות הנפוצות להתקפת Port Scanning היא התקפת TCP SYN Scan, התקפה זו מתמקדת על שליחת פאקטות TCP עם דגל SYN[7] מורם המצביע על בקשה ליצירת חיבור עם המכשיר הנתקף בפורט מסויים, אך לאחר שהתוקף מקבל תשובה מהמכשיר הנתקף, הוא לא ממשיך את תהליך יצירת החיבור על ידי זה שהוא לא שולח חזרה פאקטה עם דגל ACK מורם. התקפה זו מביא לתוקף מידע אם הפורט שאליו שלח בקשה פתוח או סגור על ידי זה שאם הוא קיבל חזרה פאקטה עם דגל ACK מורם, ואילו אם הוא קיבל תשובה עם דגל RST מורם התוקף יודע שהפורט אליו ניסה לגשת סגור. תהליך זה של שליחת הפאקטות מתבצע על כל הפורטים המיועדים לסריקה.

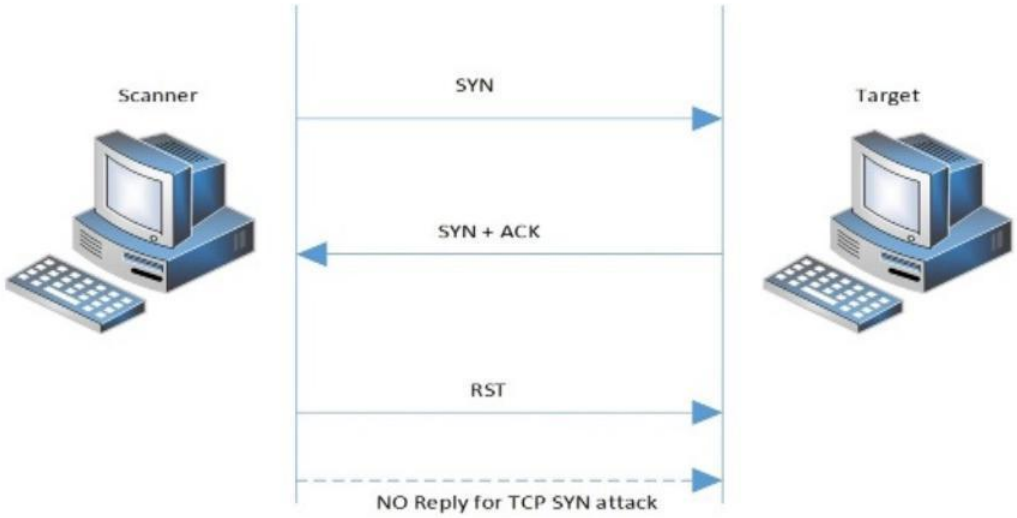
כדי להגן מפני סוגי התקפות אלה, צוות חוקרים החליט לפתח מערכת גילוי חדירה מבוססת Deep Belief Network - DBN[8]. המערכת נועדה לזהות סריקות פורטים בתעבורת רשת על ידי שימוש במודל למידת עומק הלומד לזהות דפוסים בהתנהגות זדונית. החוקרים אספו נתונים רבים על תעבורת רשת, בהם דוגמאות של תעבורה רגילה והתקפות סריקה, ואימנו את המודל שלהם כך שיוכל להבחין בין התעבורה השגרתית לבין התקפות סריקה (Viet HN et al. 2018, 118-119)

בזכות המערכת שפיתחו, החוקרים הצליחו להעניק פתרון יעיל ואמין יותר לזיהוי סריקות פורטים, גם כאשר נעשה שימוש בטכניקות מתוחכמות להימנעות מזיהוי (Viet HN et al. 2018, 120). המערכת מסוגלת לזהות במהירות דפוסים חשודים בתעבורת רשת ולסווג אותם כסריקות זדוניות, תוך שמירה על רמת דיוק גבוהה והפחתת התראות השווא.

עם זאת, קיימים מחקרים המעידים על כך שהשימוש ב Support Vector Machine, שהוא אלגוריתם למידת מכונה המשמש למטרות קלסיפיקציה או רגרסייה, מביא תוצאות טובות עבור זיהוי וקלסיפיקציה של התקפות Port Scanning ו- DDoS עם רמת דיוק של 99% עם שיעורי חיוביות שגויות נמוכים (Aamir M et al. 2021, 220-223).

[7] Tcp Protocol SYN-ACK - SYN Flag
[8] Deep Belief Neural Network - Wikipedia

בנוסף מחקרים אלו מראים כי שימוש באלגוריתמי סיווג פרימיטיבים כגון עצי החלטה ו-KNN מביאים תוצאות פחות טובות.



התקפת DoS

התקפות כמו Denial of Service ו- Distributed Denial of Service הן מההתקפות הנפוצות ביותר על ארגונים כיום, במיוחד כלפי שרתים מרכזיים וחשובים בהם נעשה שימוש נרחב. התקפות אלה פועלות על ידי שליחה מסיבית של חבילות מידע (packets) לשרתים בזמן קצר, מה שגורם לשימוש מוגזם במשאבי זיכרון של השרת, מה שיכול להוביל לקריסתו, ובכך מונע גישה למשתמשים לגיטימיים.

ההתקפות הנפוצות מסוג זה הם התקפות TCP SYN Flood ו- UDP Flood. התקפת TCP SYN Flood היא הנפוצה ביותר והיא מאופיינת על ידי שליחת פאקטות TCP עם דגל SYN מורם, לבקשת יצירת ערוץ תקשורת (connection) לפורט מסוים של המכשיר הנתקף בכמויות גדולות, אך בעת קבלת תשובה מן המכשיר הנתקף, התוקף ממשיך לשלוח עוד ועוד בקשות ליצירת ערוץ תקשורת. פעולה זו גורמת לריבוי connections פתוחים על השרת הנתקף ובכך מבזבזת זיכרון ויכולה לגורום לקריסות על המערכת הנתקפת. ההתקפה השניה הנפוצה ביותר היא UDP Flood שדומה מאוד להתקפה הקודמת אך בהתקפה זו שולחים פאקטות UDP בכמויות גדולות וגורמת לתוצאה דומה.

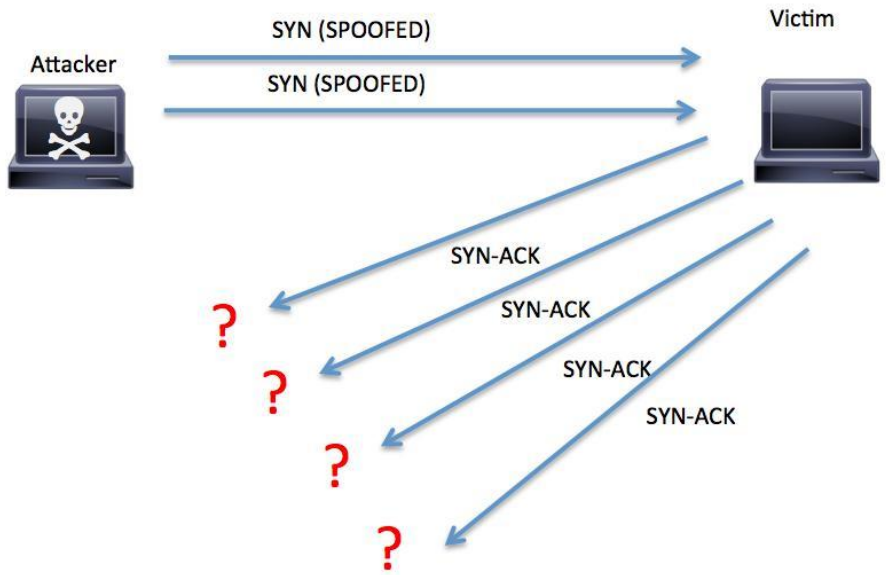
עם זאת, זיהוי התקפות אלה הפך לפשוט יותר בעקבות התפתחות אלגוריתמי למידת מכונה, המאפשרים ליצור מודלים שמבדילים בין תעבורה רגילה לתעבורה זדונית. מחקרים מראים כי רשתות נוירונים כמו Long Short Term Memory (LSTM) וגם אלגוריתמים קלאסיים יותר כמו עצי החלטה ו-Random Forests יכולים להגיע לרמות דיוק גבוהות בזיהוי התקפות, עם שיעורי חיוביות שגויות (False Positives) נמוכים יחסית (Muhuri PS et al. 2020, 12-15).

במחקר מסוים פותח מודל לזיהוי חדירות על בסיס שילוב של רשתות LSTM-RNN עם אלגוריתם גנטי(GA[9]) לבחירת תכונות אופטימליות (Muhuri PS et al. 2020, 9-10). המודל הוכיח יעילות גבוהה במיוחד במיון רב-קטגורי וביצועים טובים יותר בהשוואה ל-Support Vector Machine ו- Random Forest. בסיווג בינארי, דיוק המודל היה דומה לזה של RF וגבוה יותר מזה של SVM על אותם נתונים (Muhuri PS et al. 2020, 18). מחקר זה מראה את חשיבות בחירת התכונות הנכונות מתעבורת הרשת, מכיוון שתכונות שמזהות סוג אחד של התקפה עשויות להיות לא יעילות לזיהוי סוגים אחרים של התקפות.

סוג נפוץ נוסף של התקפות DoS הוא התקפת HTTP DoS. התקפות אלו מכוונות בעיקר לאפליקציות אינטרנטיות, כגון שרתים ואתרי אינטרנט, ומטרתן לשבש את פעילותם.

[9] Genetic Algorithms Introduction - What is a genetic algorithm

ההתקפה מנצלת את פרוטוקול HTTP על ידי שליחה מסיבית של בקשות GET או POST לאותם שרתים ואתרים. כתוצאה מכך, השרת מוצף בכמות עצומה של בקשות, מה שמוביל לעומס יתר ומונע מהשרת לטפל בבקשות אמיתיות שמגיעות ממשתמשים לגיטימיים. ניתן להשתמש בטכניקות ובשיטות דומות לאלו שהוצגו במאמר הנוכחי לצורך זיהוי ומניעה של סוג זה של התקפות DoS.

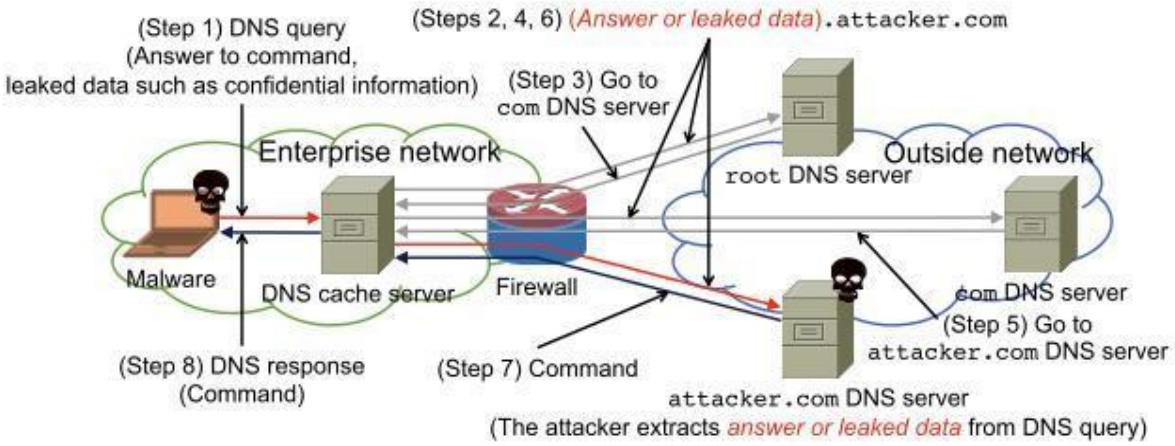


התקפת DNS Tunneling

DNS Tunneling היא מתקפת סייבר מתוחכמת בתחום התקשורת, בה נעשה שימוש בפרוטוקול ה- DNS - Domain Name System[10] ליצירת ערוץ תקשורת דו-כיווני מוסתר בין התוקף לבין מערכת הקורבן. הפרוטוקול, שתכליתו המוצהרת היא לתרגם שמות דומיין לכתובות IP ולהפנות את המשתמש לאתרים המבוקשים, מאפשר במקרים מסוימים לשלוח מידע מגוון בתוך תעבורת ה-DNS התקנית. ניצול זה מאפשר לתוקפים להחדיר פקודות זדוניות או לגנוב מידע רגיש מבלי לעורר חשד, תוך שימוש בתעבורה שאמורה להיות לגיטימית (Altuncu MA et al. 2021, 39-40)

התקפה זו יכולה להתבצע על רק על מכשירים פגועים על ידי נוזקה כלשהי כגון וירוס או תוכנה זדונית כלשהי. כאשר המכשיר פגוע, התוקף יכול ליצור DNS Tunnel, "צינור", שבאמצעותו יכול להקים ערוץ תקשורת דו-כיווני (C2[11] - Command and Control) אשר יכול לנצל את החולשות של פרוטוקול ה DNS על מנת לגנוב מידע מהמכשיר הנתקף ולהעביר אותו למכשיר התוקף, או לחילופין לשלוח פקודות זדוניות למכשיר התוקף ובכך לגרום לשיבושים במכשיר הנתקף היכולים לשמש לביצוע התקפות נוספות על המכשיר.

החולשות של פרוטוקול ה DNS מאפשר לתוקף לשלוח DNS Request מהמכשיר הנתקף לשרת של התוקף, לאחר מכן כל DNS Response החוזר מהשרת של התוקף יכיל פקודות בגוף פקאטת ה DNS. בעת קבלת הפאקטה המכשיר הנפגע יבצע את הפקודות וישלח DNS Request עם הפלט של הפקודה שקיבל חזרה לשרת התוקף.



[10] Domain Name System (DNS) - Wikipedia
[11] https://www.paloaltonetworks.com/cyberpedia/command-and-control-explained - C2

ההתקפה מציבה סיכון משמעותי למשתמשים ולארגונים, שכן היא יכולה להוביל לגניבת מידע רגיש, לפגיעות בהתקפות פישינג, ואף להתקפות זדוניות נוספות.
DNS Tunneling מספק לתוקף יכולת לשלוט על תעבורת הנתונים בין המשתמש לאתרים שהוא מנסה לגשת אליהם, ובכך הוא מגביר את הסיכון להונאות ולהתקפות סייבר נוספות. בעידן שבו אבטחת המידע קריטית מתמיד, זיהוי מוקדם של ההתקפה יכול לצמצם את הנזקים ולמנוע פגיעות חמורות.

בעידן הנוכחי, בו נעשה שימוש גובר בכלי בינה מלאכותית ולמידת מכונה, זיהוי מתקפות מסוג זה הפך ליעיל ומדויק מאי פעם. ניתוח מתקדם של שדות כגון request query ו-response data בתוך חבילות ה-DNS מאפשר לזהות בקשות חריגות או מידע שאינו כתובת IP, דבר המעיד על חשד למתקפה. כיום קיימות רשתות נוירונים מתקדמות, כמו Deep Feedforward Neural Networks, המאפשרות להשיג רמת דיוק יוצאת דופן של 99.91% בזיהוי מתקפות אלו, ואף לבצע זאת בזמן תגובה מהיר ביותר של 0.614 מילי-שניות (Altuncu MA et al. 2021, 46), מה שמקדם את מערכות ההגנה ומצמצם משמעותית את סיכוני האבטחה ברשתות תקשורת מודרניות.

הפתרון שאנחנו מציעים

על ידי שימוש במודלים של למידת מכונה ואלגוריתמים לזיהוי תפוסים חריגים בתקשורת, בפרויקט שלנו אנחנו נבנה תוכנת Hybrid Intrusion Detection System, אשר תתמקד בזיהוי דפוסים חריגים בתעבורת הרשת בזמן אמת, תתריע על התקפות סייבר בתקשורת בזמן אמת ותסייע במזעור הנזקים וההגנה על המשתמשים ובמערכות בהם התוכנה שלנו מותקנת.

התוכנה שלנו תוכל לזהות ולהתריע על סוגים שונים של התקפות כגון **ARP Spoofing**, שבא תוקף שולח הודעות ARP זדוניות על מנת להטעות את המכשירים ברשת ולהפנות אליהם את התעבורה. **DNS Tunneling**, שהיא טכניקת תקיפה בה נעשה שימוש בפרוטוקול DNS כדי להעביר נתונים מוסתרים ולבצע פעולות בלתי מורשות.
Denial of Service (DoS), שמטרתה לשבש את פעילות השרתים או המערכות על ידי הצפתן בבקשות יתר ו- **Port Scanning**, שהיא טכניקת סריקה בה תוקף בודק אילו פורטים פתוחים במחשב או בשרת כדי לאתר חולשות.

בניגוד לפתרונות הקיימים בשוק, התוכנה שלנו מציעה ערך מוסף ייחודי למשתמשים באמצעות שילוב של שיטות מתקדמות לזיהוי מתקפות סייבר בתקשורת עם ממשק משתמש ידידותי ופשוט לתפעול. התוכנה נועדה להפחית משמעותית את כמות ההתראות השגויות, תוך שיפור רמת הדיוק בזיהוי מתקפות בזמן אמת, ובכך להבטיח חוויית שימוש אפקטיבית ואמינה יותר.

ביבליוגרפיה

[1] Altuncu MA, Gulagiz FK, Ozcan H, Bayir OF, Gezgin A, Niyazov A, Cavuslu MA, Sahin S. Deep learning based DNS tunneling detection and blocking system. Adv. Electr. Comput. Eng. 2021 Aug 1;21(3):39-48.

[2] Majumdar A, Raj S, Subbulakshmi T. ARP Spoofing detection and prevention using Scapy. InJournal of Physics: Conference Series 2021 May 1 (Vol. 1911, No. 1, p. 012022). IOP Publishing.

[3] Muhuri PS, Chatterjee P, Yuan X, Roy K, Esterline A. Using a long short-term memory recurrent neural network (LSTM-RNN) to classify network attacks. Information. 2020 May 1;11(5):243.

[4] Viet HN, Van QN, Trang LL, Nathan S. Using deep learning model for network scanning detection. InProceedings of the 4th International Conference on Frontiers of Educational Technologies 2018 Jun 25 (pp. 117-121).

[5] Aamir M, Rizvi SS, Hashmani MA, Zubair M, Ahmad J. Machine learning classification of port scanning and DDoS attacks: A comparative analysis. Mehran University Research Journal Of Engineering & Technology. 2021 Jan 1;40(1):215-29.

[6] Laghrissi F, Douzi S, Douzi K, Hssina B. Intrusion detection systems using long short-term memory (LSTM). Journal of Big Data. 2021 May 7;8(1):65.

סקר שוק

התוכנה שאנו בונים במסגרת פרויקט הגמר היא מערכת זיהוי חדירה היברידית (HIDS) שתהיה מסוגלת להגן על המערכת מחשב מפני התקפות תקשורת נפוצות. התוכנה שלנו מיודעת עבור משתמשים פרטים, אשר מחפשים להגן על עצמם מפני התקפות סייבר, וגם עבור חברות אשר מחפשות מערכת שתגן על המכשירים של העובדים. המערכת שלנו תכלול ממשק משתמש אינטואיטיבי המאפשר למשתמשים להפעיל ולכבות את התוכנה, לנטר לוגים על התראות קודמות ועוד.

לצורך בניית התוכנה שלנו, ערכנו סקר שוק על מנת להבין אילו תוכנות קיימות בשוק כיום, מה החוזקות שלהם, ומה חסר להם. כחלק מהסקר שוק בחנו את ההתקפות עליהם מגנות התוכנות, באילו שיטות הם השתמשו עבור הזיהוי, ועד כמה פשוט הממשק משתמש שלהם.

<u>להלן התוכנות הקיימות בשוק:</u>

Trend Micro by TrippingPoint

Trend Micro של TippingPoint היא תוכנת IDS/IPS מתקדמת המשתמשת בטכנולוגיית DPI - Deep Packet Inspection לניתוח תוכן חבילות רשת במגוון שכבות. המערכת מזהה ומונעת מתקפות מתוחכמות כמו הזרקות SQL, מתקפות DoS/DDoS, ניצול חולשות זיכרון כמו buffer overflow ועוד. למערכת יכולות מובנות להגנה מפני מתקפות DDoS, המערכת מזהה דפוסים כמו SYN floods ו- ping floods, ומייצרת התראות בזמן אמת שמאפשרות תגובה מהירה. TippingPoint משלבת מנגנוני זיהוי מבוססי חתימות (Signature-Based), זיהוי אנומליות (Anomaly-Based) ומודיעין איומים בזמן אמת כדי להגן מפני מגוון רחב של איומי סייבר, תוך הפחתת הצורך בהתערבות ידנית ושמירה על רמת אבטחה גבוהה. בנוסף התוכנה מספקת ממשק משתמש פשוט וקל לשימוש המספק תובנות למשתמש לגבי המערכת.

Cisco Secure Firewall

Cisco Secure Firewall הוא פתרון חומת אש מדור חדש (NGFW) המציע מנגנוני מניעת חדירה (IPS) וזיהוי חדירה (IDS) מתקדמים לצד יכולות אבטחה נוספות להגנה מפני מגוון רחב של איומי סייבר. המערכת משלבת שליטה ברמת שכבת האפליקציה, זיהוי זהות משתמשים ו- DPI - Deep Packet Inspection, יחד עם בדיקות מצביות למניעת איומים ידועים ולא ידועים בזמן אמת. היא כוללת תכונות אבטחת DNS למניעת בקשות זדוניות והדלפת מידע רגיש על ידי שימוש בפרוטוקול ה- DNS.
המערכת מזהה ומונעת מתקפות כמו DoS, Port Scanning, SQL Injection Zero-Day, ו-XSS באמצעות ספריית חתימות רחבה וזיהוי אנומליות. יכולות אלו מאפשרות לה לזהות גם איומים שאינם תואמים דפוסים מוכרים, ובכך להגן על הרשת מפני מתקפות חדשות ומותאמות אישית.

Darktrace

Darktrace מציעה גישה ייחודית לאבטחת רשת המבוססת על בינה מלאכותית. בניגוד למערכות IDS/IPS מסורתיות, המשתמשות בחתימות או כללים מוגדרים מראש, Darktrace מתבססת על למידת מכונה וניתוח התנהגותי לזיהוי איומים ידועים ובלתי ידועים בזמן אמת. המערכת לומדת את "דפוס החיים" הייחודי של כל מכשיר ומשתמש ברשת, היא עוקבת אחר תעבורת הרשת ובונה base-line להתנהגות נורמלית בסביבה. כאשר מתגלות סטיות מקו הבסיס, המערכת מזהה איומים פוטנציאליים כמו מתקפות DDoS, איומים פנימיים, נוזקות, או גניבת מידע. בנוסף, Darktrace מספקת הגנה לרשתות מקומיות, סביבות ענן (AWS, Azure, Google Cloud), מכשירי IoT, ומערכות תעשייתיות (ICS), ומתאימה את עצמה באופן אוטומטי לשינויים ברשת ללא צורך בהגדרות ידניות, דבר ההופך אותה לפתרון יעיל לארגונים עם סביבות IT היברידיות.

Suricata

Suricata הוא כלי Open Source מתקדם המשלב מערכת זיהוי חדירות ברשת (NIDS), מניעת חדירות ברשת (NIPS), וכלי לניטור אבטחת רשת (NSM), המציע ניתוח תעבורה בזמן אמת וביצועים גבוהים. המערכת, שפותחה על ידי קרן OISF, ידועה בגמישותה וביכולתה לזהות מגוון רחב של איומים מבוססי רשת. Suricata מבצעת ניתוח מעמיק של חבילות, DPI - Deep Packet Inspection, ותומכת בפרוטוקולים כמו HTTP, FTP, DNS, SMTP, TLS ועוד, ומסוגלת לזהות איומים מבוססי SSL, כגון מתקפות MITM או SSL/TLS downgrade attacks. היא מנתחת תעבורה בזמן אמת ומספקת התראות בזמן אמת, כולל זיהוי פעילות חשודה בשכבת הרשת, כמו מתקפות ARP spoofing. הכלי Suricata משתלבת עם כלים נוספים כמו מערכות SIEM, חומות אש ופלטפורמות IDS/IPS, ומשמשת להזרמת נתונים לצורך ניתוח וקורלציה מתקדמים.

Vectra AI

Vectra AI מציעה גישה חדשנית לאבטחת רשת באמצעות זיהוי איומים מבוסס בינה מלאכותית (AI) וניתוח התנהגותי לזיהוי איומים ידועים ולא ידועים בזמן אמת.
הפלטפורמה מתמקדת בניתוח תעבורת רשת, תגובה אוטומטית, וזיהוי התנהגותי, ומאפשרת זיהוי התקפות מורכבות כמו APT - Advanced Persistent Threats, איומים פנימיים ונוזקות שלא מתגלות על ידי כלים מסורתיים. אין צורך בהתקנת התוכנה על גבי מכשירים בודדים ברשת, מה שהופך את הפלטפורמה מתאימה במיוחד לסביבות גדולות ומורכבות כמו מרכזי נתונים, תשתיות ענן וסביבות היברידיות. מערכת משתמשת בלמידת מכונה ליצירת base-line של "התנהגות נורמלית" ברשת, ומסוגלת לזהות סטיות המעידות על פעילות זדונית. המערכת מנתחת metadata מתעבורת הרשת, כולל נתוני זרימה ורמת חבילות, ומספקת זיהוי מתקדם של תנועות רוחביות,
malware infections , command-and-control (C2) communications וטכניקות תקיפה נוספות, תוך התאמה דינמית לשינויים בטקטיקות התוקפים.

הפרויקט שלנו

התוכנה שנפתח תהיה Hybrid IDS ותתמקד בזיהוי התקפות תקשורת מסוגים שונים על ידי שימוש בטכנולוגיות מתקדמות של למידת מכונה וגם על ידי שיטות rule-based הנפוצות בשוק. התוכנה שלנו תספק פעולת סריקת הרשת עבור חיפוש מתקפות סייבר בתקשורת, המערכת תתריע למשתמש על התקפות בזמן אמת. בנוסף המערכת תספק ממשק משתמש פשוט וקל לשימוש, אשר יספק למשתמשים יכולת להפעיל ולכבות את שירות הסריקה, לנטר לוגים של התראות קודמות ועוד.
על ידי מיזוג של שיטות הקיימות בשוק כמו DPI - Deep Packet Inspection הנפוצות במערכות Signature-Based IDS ואלגוריתמי למידת מכונה כמו SVM והנפוצות ב Anomaly-Based IDS, המערכת שלנו תדע לסווג את תעבורת הרשת בזמן אמת ולהתריע למשתמש בעת זיהוי סכנה. על סמך סקר הספרות אנו מצפים כי השימוש בשיטות מתקדמות יוביל אותנו לקבלת כמות נמוכה מאוד של התראות שווא ורמת דיוק גבוהה מאוד בזיהוי ההתקפות בזמן אמת, מה שיתרום למשתמשים של התוכנה שלנו.

מסמך ייזום ואפיון

תמצית מנהלים:

1. תקציר מנהלים
2. לקוחות
3. הגדרת הבעיה
4. יעדים ומטרות
5. יתרונות צפויים
6. יישום ותוכן הפרויקט
7. סביבת פיתוח, כלים ודיאגרמות
8. דרישות המערכת
9. מימוש המערכת המשוער (GANTT)
10. מדדי הצלחה
11. ניהול סיכונים
12. עלות צפויה

תקציר מנהלים

עבור חברות מכל הסוגים, כמו גם עבור משתמשים פרטיים הגולשים ברשת על בסיס יומיומי, אשר שמים דגש על אבטחת המידע שלהם, ורוצים למזער את הסיכונים שלהם מפני התקפות סייבר פוטנציאליות הנפוצות כיום בתקשורת, התוכנה שלנו "NetSpect" תספק הגנה מפני התקפות נפוצות בתקשורת על ידי כך שתנטר את התעבורה בתקשורת בזמן אמת ותתריעה למשתמשים על התרחשות התקפות פוטנציאליות על התקשורת שלהם. בניגוד לתוכנות הקיימות כיום בשוק, לתוכנה שלנו תהיה יכולת טובה בזיהוי התקפות תקשורת נפוצות בזמן אמת וגם תכלול ממשק גרפי ידידותי למשתמש.
המוצר שלנו היא תוכנת Hybrid Intrusion Detection System[12] אשר משתמשת באלגוריתמי למידת מכונה בשילוב עם אלגוריתמי זיהוי תפוסים חריגים בתקשורת וכך תוכל להתריע בזמן אמת על התרחשות התקפות סייבר בתקשורת אותה התוכנה מנטרת.

[12] https://www.stamus-networks.com/blog/what-are-the-three-types-of-ids

לקוחות

הלקוחות העיקריים של התוכנה הם:

- אנשים פרטיים אשר חשובה להם אבטחת המידע ומעוניינים בתוכנה שמסוגלת לזהות איומי סייבר נפוצים בתקשורת תוך ידי הענקת ממשק פשוט וקל לשימוש.
- חברות מכל הגדולים אשר מספקים מחשבי עבודה לעובדים, ומעוניינים בתוכנה שתעזור להם להגן על מכשירי הקצה של העובדים והמידע שלהם בכך שתתריע מפני איומי סייבר בתקשורת על בסיס יום יומי.

הגדרת הבעיה

מתקפות סייבר בתקשורת מציבות איום גובר ומשמעותי על עסקים ופרטים, בעיקר בעידן שבו התלות בטכנולוגיה ובאינטרנט הולכת ומתרחבת. איומים אלו כוללים גניבת נתונים רגישים, כגון פרטי כרטיסי אשראי ומידע אישי, אשר מנוצלים לצרכי הונאה וגורמים לנזקים כלכליים חמורים, לצד פגיעה בפרטיות המשתמשים ואובדן אמון הלקוחות. בנוסף, מתקפות כמו DoS (Denial of Service) משבשות שירותים קריטיים, משביתות אתרי אינטרנט ומערכות מקוונות, ומובילות לאובדן הכנסות משמעותי עבור עסקים, בעוד שמתקפות מתוחכמות כמו DNS Tunneling מאפשרות הפצה של תוכנות זדוניות, רוגלות ווירוסים, אשר פוגעות בתפקוד התקין של מערכות ומסכנות את פרטיות המידע. בעידן שבו יותר מ-16 מיליארד מכשירים מחוברים לאינטרנט, החל ממכשירים אישיים ועד מערכות תעשייתיות, הצורך בהגנה אפקטיבית על מערכות תקשורת הופך לקריטי מאי פעם, מה שמדגיש את הצורך בפתרונות חדשניים ואפקטיביים יותר, אשר יאפשרו זיהוי איומים בזמן אמת והפחתת הסיכון לנזקים חמורים.

יעדים ומטרות

1. מטרות

מטרת הפרויקט שלנו היא לפתח תוכנת Hybrid Intrusion Detection System המסוגלת להתריע בזמן אמת על מגוון התקפות סייבר נפוצות בתקשורת בשילוב ממשק ידידותי למשתמש. התוכנה צריכה לעבוד באופן שוטף ברקע של המכשיר בו היא מותקנת. כמו כן התוכנה שלנו תספק זיהוי של התקפות כמו DoS, Port Scan, DNS Tunneling ו- ARP Spoofing בזמן אמת על ידי שימוש באלגוריתמי למידת מכונה בשילוב עם אלגוריתמי זיהוי תפוסים חריגים בתקשורת. בנוסף לכל משתמש, פרטי או עובד חברה, תהיה אפשרות להפעיל ולכבות את הסריקה של תעבורת התקשורת, לצפות בהיסטוריית ההתראות שלו וגם לשמור מידע אודות ההתראות קודמות שהתרחשו על התקשורת. התוכנה תאפשר למשתמשים שלה לזהות התקפות סייבר בזמן אמת ובכך להגן על המידע שלהם בכל רגע נתון ובכל רשת אינטרנט אליה היו מחוברים. אנו שואפים שהתוכנה תנצל באופן מיטבי את יכולות החישוב המקבילי של המכשיר שבו היא מותקנת, על ידי שימוש בריבוי תהליכונים. גישה זו תאפשר ניצול יעיל של משאבי המערכת, תספק ביצועים מהירים יותר, ותשפר את חוויית המשתמש באמצעות תגובתיות גבוהה ותפעול חלק.

2. יעדים

- לבצע מחקר מעמיק של התקפות תקשורת שהוגדרו, כולל ARP Spoofing, Port Scan, DoS, ו-DNS Tunneling. המחקר יתמקד בהבנת דפוסי הפעולה של כל התקפה, המנגנונים שהיא מנצלת, והדרכים לזיהוי והתגוננות מפניה. התהליך יתבסס על קריאת מאמרים אקדמיים, סקירת פרסומים מקצועיים ואיסוף מידע עדכני ממאגרים מוכרים כדי לבנות תשתית ידע רחבה ומעמיקה.

- לבנות אלגוריתם זיהוי תפוסים חריגים בתקשורת המסוגל לזהות את דפוס ההתקפה של ARP Spoofing על ידי ניטור תעבורת ARP בתקשורת, ותבדוק בזמן אמת האם זוג הכתובות IP-Mac תואם את המידע אשר נמצא בטבלת ARP אשר הכין מראש. שיטה זו תאפשר לזהות Spoofing של כתובות Mac בתקשורת ובכך לזהות התקפה פוטנציאלית, כלומר Mac Address משתייך ליותר מכתובת IP אחת בתקשורת. לאחר מכן נרחיב את האלגוריתם לעבוד במצבים בהם למשתמש יש יותר מכתובת IP אחת ונמצא ביותר מ- subnet אחד.

- לאסוף נתוני תעבורה מגוונים, כולל תעבורת רשת תקינה ותעבורת רשת נגועה בהתקפות סייבר כגון Port Scan, DoS, ו-DNS Tunneling. הנתונים ייאספו ממאגרי נתונים ציבוריים באינטרנט, וגם באמצעות תעבורת תקשורת לוקאלית על מכשירי חברי הצוות גם ברשת הביתית וגם ברשת המכללה. הנתונים אשר ייאספו באופן ידני יעובדו, ויאוחדו לקבוצות נתונים המיועדות לאימון מודלים ייחודיים לכל סוג התקפה.

- לפתח מודלים מבוססי למידת מכונה על בסיס הנתונים שנאספו. כל מודל יתוכנן בהתאם לדפוסי ההתקפה הייחודיים, תוך ביצוע Feature Selection קפדני המבטיח זיהוי מדויק ומזעור של התראות שגויות (False Positives). המודלים ייבחנו בתרחישי תקיפה מגוונים כדי להבטיח דיוק וביצועים טובים.

- לשלב את המודלים והאלגוריתמים שפותחו במערכת תוכנה אחידה שתנצל את יכולות החישוב המקבילי של המכשיר בו היא פועלת. השימוש בריבוי תהליכונים יבטיח ניצול מיטבי של משאבי המערכת, ביצועים מהירים ויכולת תגובה בזמן אמת לזיהוי התקפות תקשורת שונות.

- לבנות ממשק משתמש גרפי אינטואיטיבי ומתקדם לסביבת Desktop הכולל חיבור לבסיס נתונים. הממשק יאפשר שליטה קלה במערכת, כולל הפעלה וכיבוי, צפייה בהיסטוריית ההתראות ושמירת קבצי log, תוך מתן דגש על חוויית משתמש ידידותית ופשטות תפעול גם עבור משתמשים שאינם בעלי ידע טכני מתקדם.

יתרונות צפויים

המערכת מספקת יכולת זיהוי מתקדמת של התקפות סייבר בתקשורת, כולל ARP Spoofing, DoS, DNS Tunneling ו-Port Scanning, באמצעות ניתוח תעבורה חכם בזמן אמת. היא כוללת ממשק גרפי אינטואיטיבי וקל לשימוש, המיועד לכל סוגי המשתמשים ומאפשר הפעלה, כיבוי וצפייה בהתראות בצורה פשוטה וברורה. התוכנה מספקת התראות בזמן אמת על כל התקפה שזוהתה, תוך פירוט סוג ההתקפה, מקורה והפרטים הרלוונטיים. בנוסף, היא מנצלת את יכולות החישוב המקבילי של המערכת, מה שמבטיח ניטור מהיר ואפקטיבי של תעבורת התקשורת, גם במצבים של עומס כבד. המבנה המודולרי של המערכת מאפשר להוסיף מודלים ואלגוריתמים חדשים לתוכנה, ובכך להתאים אותה לאיומים מתפתחים ולשמור על הגנה מתקדמת לאורך זמן.

שימוש בתוכנה שלנו תציג יתרונות משמעותיים לעסקים ולמשתמשים פרטיים, אשר מעוניינים להישאר מוגנים מפני התקפות סייבר פוטנציאלים בתקשורת. התוכנה מציעה פתרון מקיף להתראה על התקפות נפוצות בתקשורת, תוך יכולת התאמה מתמדת לאיומים חדשים. היכולת לקבל התראות בזמן אמת, יחד עם ממשק ידידותי למשתמש, מקנה למשתמשים שליטה וניהול יעיל של הגנה תקשורתית. השימוש במערכות המבוססות על יכולות חישוב מקבילי מבטיח ביצועים גבוהים ואמינים, מה שמאפשר שמירה על תפעול שוטף של כל המערכות. באמצעות כלי ניתוח היסטוריית ההתראות וקבצי log, ניתן לבצע אבחונים מעמיקים של תקלות וניסיונות תקיפה, ולהשיג תובנות חשובות מכך.

בנוסף, היתרון המרכזי של מערכת Hybrid IDS הוא היכולת לשלב את היתרונות של שתי הגישות המרכזיות, Signature-based ו- Anomaly-based, ובאותו זמן להפחית את המגבלות המאפיינות כל אחת מהן. המערכת מזהה איומים ידועים במהירות באמצעות חתימות (Signature-based), ומאתרת איומים חדשים או מתוחכמים בעזרת זיהוי חריגות (Anomaly-based). שילוב זה מפחית את שיעור ההתראות השגויות (False Positives) הנפוץ במערכות מבוססות חריגה, ומספק הגנה משופרת בסביבות דינמיות ומורכבות. בנוסף, השיטה ההיברידית מספקת מענה רחב יותר, המותאם לאיומים מגוונים, ומבטיחה כיסוי אבטחתי מלא יותר בהשוואה לשימוש בגישה אחת בלבד.

יישום ותוכן הפרויקט

1. אופי המערכת

- ❖ מערכת המסוגלת לזהות התקפות סייבר בתקשורת בזמן אמת (Hybrid IDS).
- ❖ המערכת מיועדת גם עבור מכשירי Desktop PC פרטיים וגם עבור חברות.
- ❖ למערכת ממשק משתמש פשוט וקל לשימוש.
- ❖ המערכת מסוגלת לזהות התקפות סייבר נפוצות כמו Port Scan, DoS, ARP Spoofing ו- DNS Tunneling ובכך מקנה הגנה רחבה עבור מערכת המשתמש.
- ❖ המידע של כל משתמש נשמר בבסיס נתונים מאובטח אשר יאפשר למשתמשים לצפות במידע כמו היסטוריית התראות.
- ❖ ההתחברות למערכת תהיה מאובטחת על ידי אלגוריתמי הצפנה המובילים בשוק.

2. אילוצים

- ❖ המערכת יכולה להיות מותקנת רק על מכשירי Desktop PC לכל מערכת הפעלה (Windows, MacOS, Linux), אך לא תעבוד על מכשירים סלולריים וטאבלטים.
- ❖ על מנת לקבל גישה לבסיס הנתונים יש להתקין SQL Server.
- ❖ במערכת Windows התוכנה דורשת התקנה של תוכנת Npcap המאפשרת ניטור התקשורת בזמן אמת.
- ❖ על מנת שהתוכנה תעבוד נדרש להתקין Python 3.13.0 ומעלה.

3. מגבלות

- ❖ המערכת שלנו מבצעת זיהוי בלבד, היא לא תומכת בהגנה מפני התקפות אותן היא מזהה. זאת מכיוון שאנחנו מפתחים מערכת זיהוי IDS ולא EDR.
- ❖ המערכת דורשת פריבילגיית ריצה "admin" על מנת לנטר את תעבורת התקשורת בזמן אמת.

4. משתמשים

המשתמש	סוג הרשאה	הפעולות במערכת
משתמש אורח	גישה מוגבלת	הפעלת התוכנה, כיבוי התוכנה, קבלת התראות בעת זיהוי התקפה בזמן אמת וצפייה בהיסטוריית ההתקפות מאז ההפעלה האחרונה של התוכנה.
משתמש מחובר	גישה מלאה	אותן פעולות כמו משתמש אורח אך בנוסף לכך יכול גם להתחבר ולהתנתק מהמערכת, שמירת מידע אודות התראות, הוצאת דוחות ועוד.

5. התהליכים

<u>תהליכים של משתמש מסוג אורח:</u>

התליך	תיאור
הפעלה וכיבוי של התוכנה	המשתמש יכול להפעיל את התוכנה כדי לנטר את תעבורת התקשורת ובכך לזהות התקפות סייבר פוטנציאליות בזמן אמת. למשתמש היכולת להפעיל ולעצור את התוכנה בכל רגע נתון על ידי לחיצה פשוטה על כפתור.
קבלת התראות על התקפות בזמן אמת	המשתמש מקבל באופן אוטומטי התראות ברגע בו המערכת תזהה התקפה פוטנציאלית בתעבורת התקשורת שלו, ההתראה תוצג למשתמש על המסך התוכנה ותספק למשתמש את כל המידע הרלוונטי על ההתקפה.
הרשמה למערכת	כל משתמש אורך יכול בכל רגע נתון לבצע הרשמה למערכת ולפתוח משתמש חדש. פעולה זו מאפשרת למשתמש לקבל גישה לפיצ׳רים נוספים שאין למשתמש אורח.
צפייה בהיסטוריית ההתראות זמנית	המשתמש מסוגל לצפות ברשימה של כל ההתראות על התקפות סייבר בתקשורת שזוהו על המכשיר שלו מהפעם האחרונה שהפעיל את התוכנה, זאת מפני שלמשתמש אורח אין יכולת זיכרון ולכן הכל נשמר באופן לוקאלי.
קבלת מידע על המערכת	המשתמש יכול לקבל מידע כללי (metadata) על המערכת שלו כמו כתובת IPv4, IPv6, Mac-Address ועוד.

שמירת היסטוריית התראות	המשתמש יכול לשמור את היסטוריית ההתראות שלו מאז ההפעלה האחרונה של המערכת (שמירה כקובץ / הוצאת דוח).
מיון התראות לפי סוג ותקופה	המשתמש יכול לבצע מיון התראות לפי סוג התקפה ולפי התקופה בא ההתראה התקבלה.
צפייה במידע אנליטי אודות ההתראות לפי שנה	המשתמש יכול להסתכל על מידע גרפי אשר ממפה את ההתראות שלו לאורך השנה בצורה גרפית.
ביצוע פעולות בתוכנה דרך ה- Tray Icon	המשתמש יכול לבצע פעולות בסיסיות באפליקציה כמו להפעיל ולכבות את הסריקה, לגשת לעמוד מסויים בלחיצת כפתור בעזרת ה- Tray Icon.
שחזור סיסמה בעזרת הודעה הנשלחת לכתובת המייל	המשתמש יכול לשחזר את הסיסמה שלו בעזרת שליחת הודעה לכתובת מייל והזנת הקוד הסודי לצורך יצירת סיסמה חדשה.
הוספת כתובת ל- MAC Blacklist למניעת התראות	המשתמש יכול להוסיף כתובות MAC לרשימה אשר תמנע קבלת התראות על התקפות מאותה כתובת MAC.
החלפה בין מצבים Light Mode ו- Dark Mode	המשתמש יכול לשנות את נראות האפליקציה על ידי בחירה של ערכת נושא שונה כאשר יש בחירה בין מצב בהיר למצב כהה.

תהליכים של משתמש מסוג מחובר:

התליך	תיאור
התחברות והתנתקות מהמערכת	המשתמש יכול להתחבר ולהתנתק מהמערכת בכל רגע נתון, פעולה זו היא אופציונלית ומספקת למשתמש גישה מלאה לשאר הפיצ׳רים הקיימים במערכת שלמשתמש אורח אין.
שינוי פרטי המשתמש	משתמש המחובר למערכת יכול בכל רגע נתון לערוך את הפרטים האישיים שלו כמו שם משתמש, סיסמה ועוד.
הפעלה וכיבוי של התוכנה	המשתמש יכול להפעיל את התוכנה כדי לנטר את תעבורת התקשורת ובכך לזהות התקפות סייבר פוטנציאליות בזמן אמת. למשתמש היכולת להפעיל ולעצור את התוכנה בכל רגע נתון על ידי לחיצה פשוטה על כפתור.
קבלת התראות על התקפות בזמן אמת	המשתמש מקבל באופן אוטומטי התראות ברגע בו המערכת תזהה התקפה פוטנציאלית בתעבורת התקשורת שלו, ההתראה תוצג למשתמש על המסך התוכנה ותספק למשתמש את כל המידע הרלוונטי על ההתקפה.
צפייה בהיסטוריית ההתראות המלאה	המשתמש מסוגל לצפות ברשימה של כל ההתראות על התקפות סייבר בתקשורת שזוהו על המכשיר שלו מהרגע הראשון שמתחבר למערכת, כלומר היסטוריה מלאה של התראות כפי שנשמרה בבסיס

	הנתונים.
קבלת מידע על המערכת	המשתמש יכול לקבל מידע כללי (metadata) על המערכת שלו כמו כתובת IPv4, IPv6, Mac-Address ועוד.
הוצאת דוחות על התראות קודמות	משתמש יכול להוציא דוחות מהמערכת על התראות קודמות, סטטיסטיקות על התראות קודמות ועוד.
מחיקת משתמש	המשתמש בכל רגע נתון יכול לבחור למחוק את המשתמש שלו מהמערכת וכך להסיר את כל הנתונים שלו מן השרת.
מחיקת התראות	המשתמש בכל רגע נתון יכול למחוק את היסטוריית ההתראות שלו.
שמירת היסטוריית התראות	המשתמש יכול לשמור את היסטוריית ההתראות שלו מאז ההפעלה האחרונה של המערכת (שמירה כקובץ / הוצאת דוח).
מיון התראות לפי סוג ותקופה	המשתמש יכול לבצע מיון התראות לפי סוג התקפה ולפי התקופה בא ההתראה התקבלה.
צפייה במידע אנליטי אודות ההתראות לפי שנה	המשתמש יכול להסתכל על מידע גרפי אשר ממפה את ההתראות שלו לאורך השנה בצורה גרפית.
ביצוע פעולות בתוכנה דרך ה- Tray Icon	המשתמש יכול לבצע פעולות בסיסיות באפליקציה כמו להפעיל ולכבות את הסריקה, לגשת לעמוד מסויים בלחיצת כפתור בעזרת ה- Tray Icon.
שחזור סיסמה בעזרת הודעה הנשלחת לכתובת המייל	המשתמש יכול לשחזר את הסיסמה שלו בעזרת שליחת הודעה לכתובת מייל והזנת הקוד הסודי לצורך יצירת סיסמה חדשה.
הוספת כתובת ל- MAC Blacklist למניעת התראות	המשתמש יכול להוסיף כתובות MAC לרשימה אשר תמנע קבלת התראות על התקפות מאותה כתובת MAC.
החלפה בין מצבים Light Mode ו- Dark Mode	המשתמש יכול לשנות את נראות האפליקציה על ידי בחירה של ערכת נושא שונה כאשר יש בחירה בין מצב בהיר למצב כהה.
החלפה בין מצבים Detection Mode ו- Collection Mode	המשתמש יכול לבחור בין מצבים שונים כמו מצב Detection שבו התוכנה מזהה התקפות ואילו במצב Collection התוכנה אוספת נתונים אודות תעבורת התקשורת ושומרת אותם בקובץ CSV.

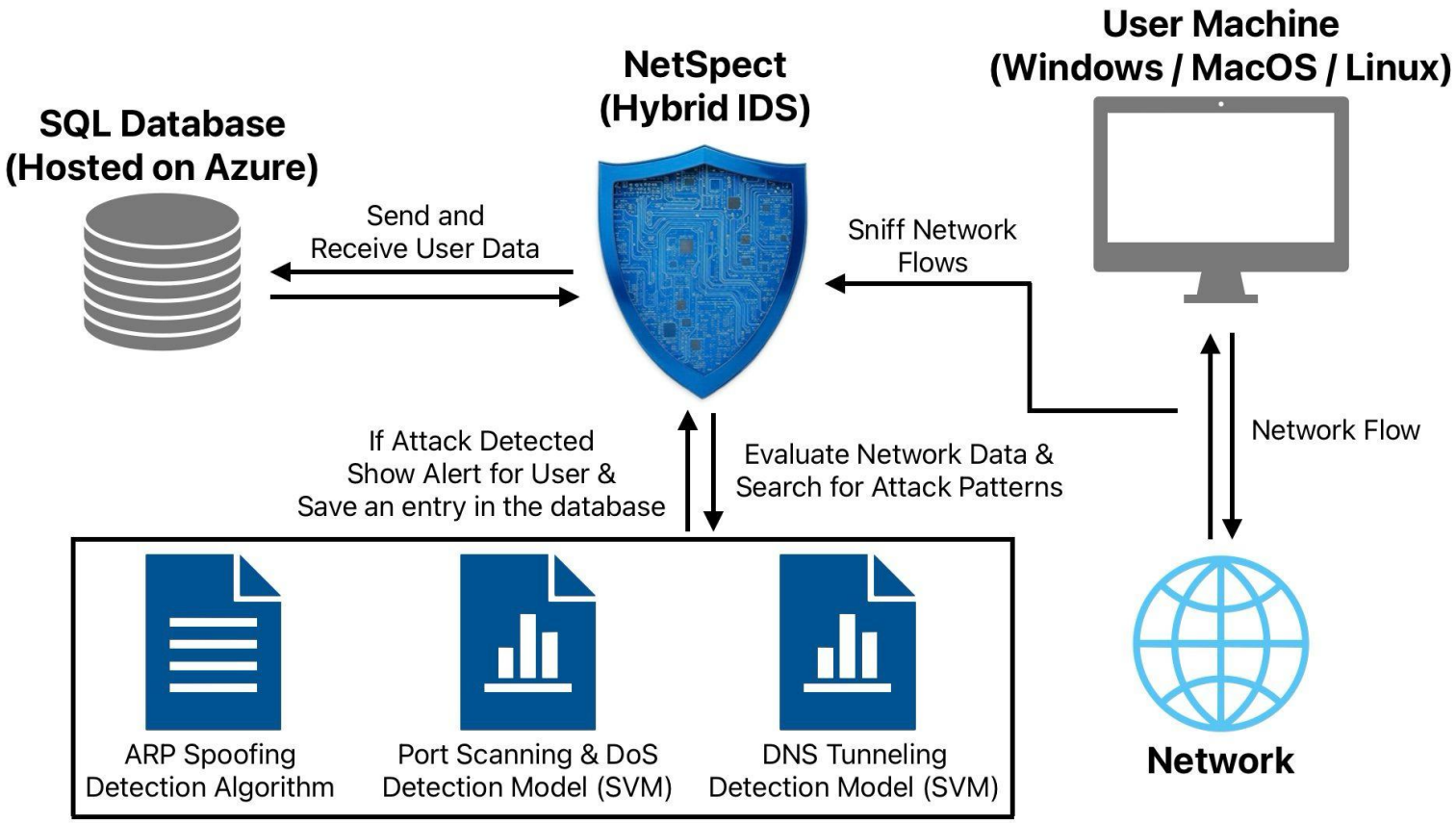
6. דוחות

<u>משתמש מחובר יכול להוציא את הדוחות הבאים:</u>

- ❖ דוח המכיל רשימה וכמות התראות שהתקבלו ביממה האחרונה
- ❖ דוח המכיל רשימה וכמות התראות שהתקבלו בשבוע האחרון
- ❖ דוח המכיל רשימה וכמות התראות שהתקבלו בחודש האחרון
- ❖ דוח המכיל רשימה וכמות התראות שהתקבלו בשנה האחרונה
- ❖ דוח המכיל סטטיסטיקה לבחירת המשתמש אודות ההתראות שהתקבלו במערכת בתקופה מסוימת עבור התקפה מסוימת, למשל:
  - ➢ מספר התראות על התקפה מסוג Port Scan בחודש האחרון לעומת מספר התקפות DNS Tunneling באותה תקופת זמן.
  - ➢ כמות התראות על התקפה מסוג Port Scan ו- DoS בשבוע האחרון.
- ❖ דוח המכיל מידע על המערכת כמו פרטים על המערכת הפעלה, נתוני תקשורת ועוד.

סביבת פיתוח, כלים ודיאגרמות

1. ארכיטקטורת מערכת



2. סביבת פיתוח

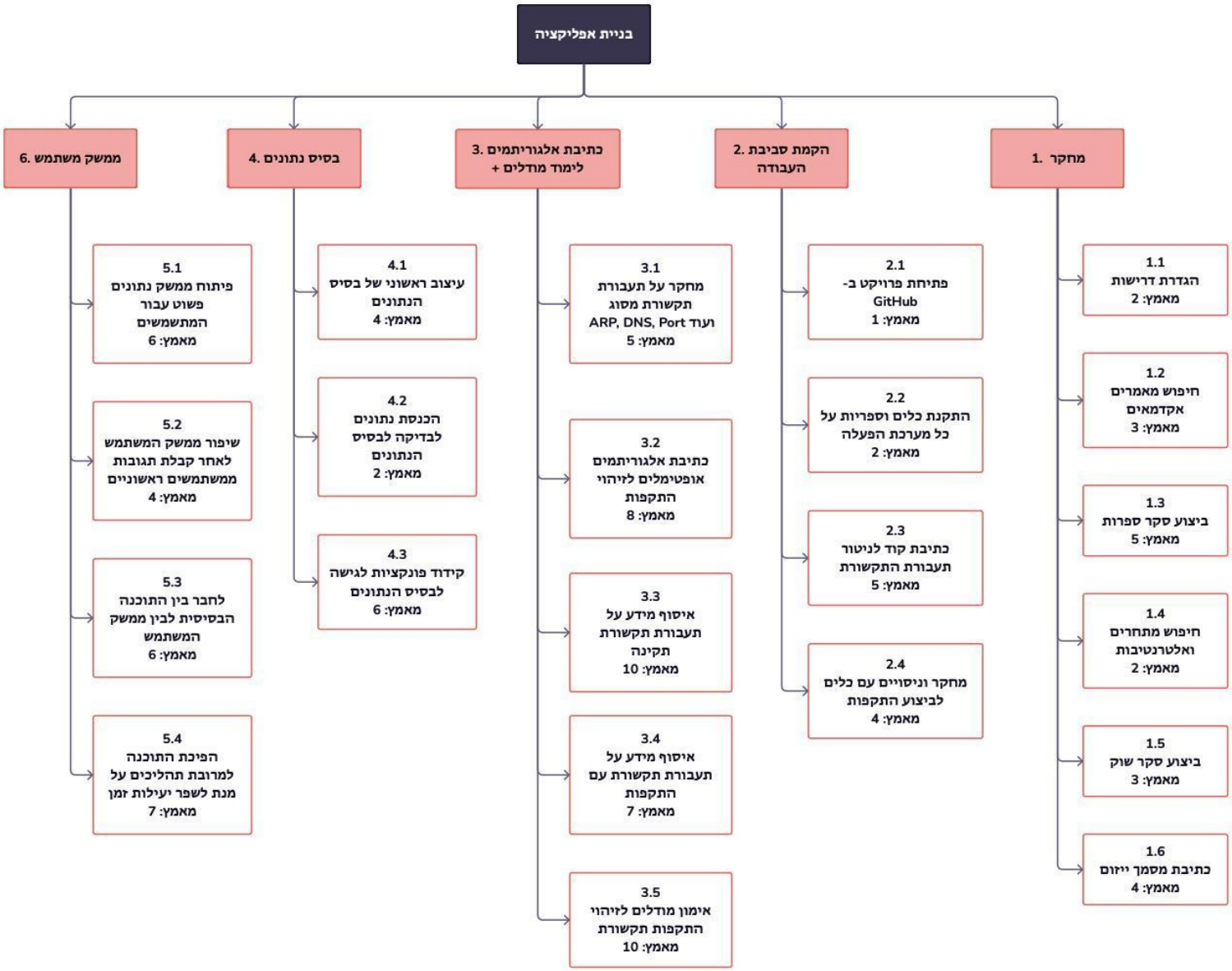
המערכת תפותח על המחשבים האישיים של חברי הצוות, מחשבים אלו כוללים את שלושת מערכות ההפעלה העיקריות, Windows 11, MacOS Sequoia 15.1 ו- Linux Ubuntu 22.04.
המערכת תפותח בשפת Python בגרסה 3.13.0. לצורך כתיבת הקוד ואימון המודלים נשתמש ב- VS Code. עבור בסיס הנתונים נשתמש ב- SQL Server. לצורך הדמיית ההתקפות נשתמש בכלים מוכנים כמו DoS Hulk, Nmap ועוד.
הקוד יועלה ל- repository פרטי ב- GitHub בקישור הבא.

3. כלים וספריות

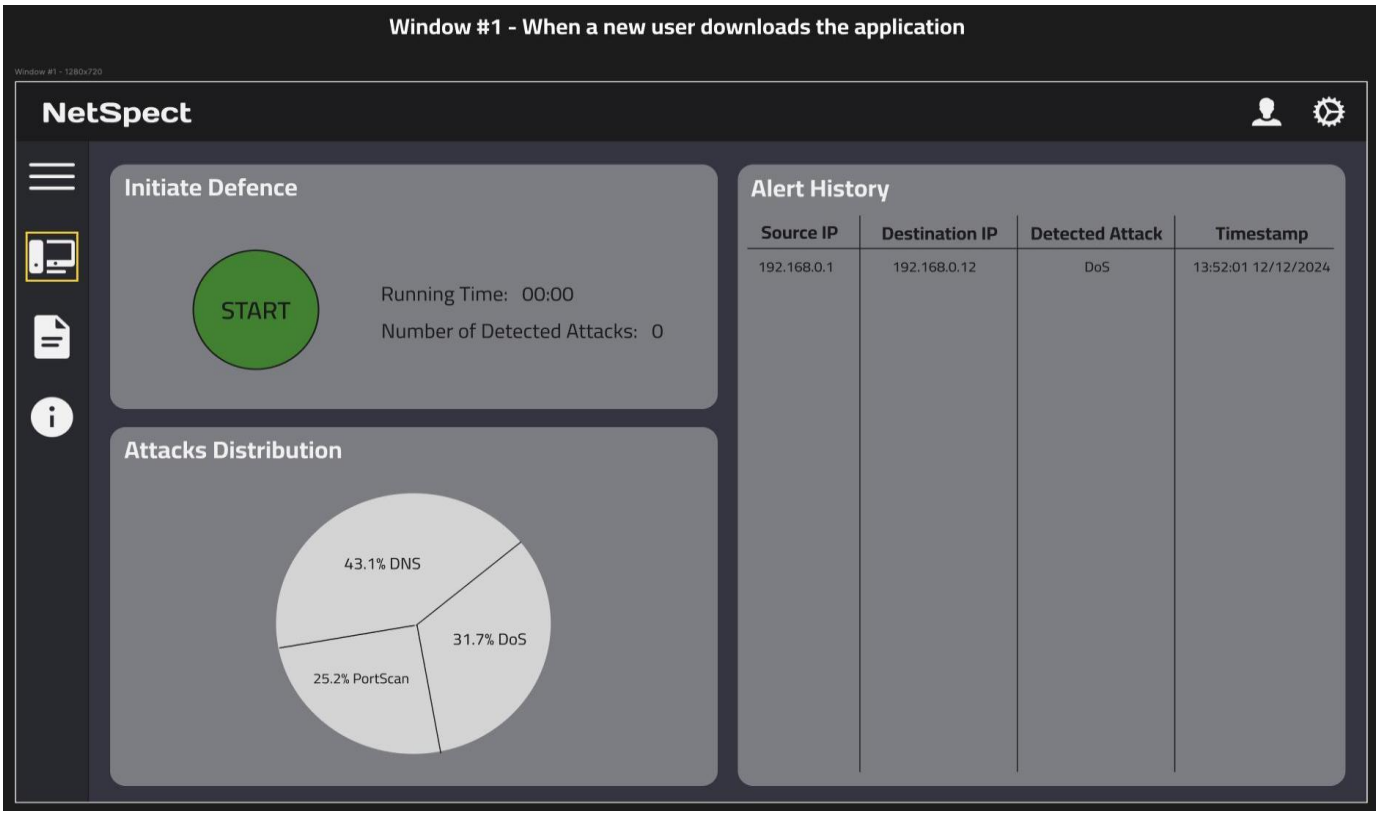
- **Python 3.13.0**
- **Scapy** - ספרייה המאפשרת לנטר את תעבורת התקשורת המקומית, לאיסוף מידע על כל מה שמתרחש בתקשורת, לשלוח פקטות (packets) בתקשורת במגוון דרכים וגם לשמור נתונים בקבצים ואובייקטים בפורמט pcap במידת הצורך. נשתמש בספרייה זו ככלי מרכזי אשר יאפשר לנו לאיסוף ולנתח מידע על תעבורת התקשורת בזמן אמת.
- **Npcap** - ספרייה עבור מערכת הפעלה Windows המאפשרת לתוכנות צד שלישי (כמו התוכנה שלנו) לנטר את תעבורת התקשורת. ללא הורדת ספרייה זו, מערכת ההפעלה תחסום כל ניסיון גישה לתעבורת התקשורת.
- **PySide6** - ספרייה המאפשרת לייצר ממשק משתמש איכותי בשפת Python אשר מותאם למכשירי Desktop מכל סוגי מערכות ההפעלה ולא רק, ספרייה זו נפוצה בשוק והרבה חברות גדולות משתמשות בה לכל מני צרכים. נשתמש בספרייה זו כדי לפתח ממשק משתמש בצורה פשוטה.
- **Scikit-Learn** - ספרייה המספקת אלגוריתמים לבניית מודלי למידת מכונה כמו RandomForest, SVM, KNN ועוד. ספרייה זו תשמש אותנו לבניית מודלים לזיהוי מתקפות סייבר בתקשורת.
- **Pandas & Numpy** - ספריות מוכרות אשר עוזרות למתכנתים לעבור עם BigData וטבלאות על זה שמספקות פונקציונליות נוחה וקלה לשימוש. בנוסף בעזרת ספריית Numpy ניתן לשפר את הביצועים של כל תוכנת Python מכיוון שהספרייה זו מספקת מימושים המבוססים על שפת C++ ולכן יעילים יותר.
- VS Code
- GitHub
- SQL Server
- Microsoft Word
- **Flask** - ספרייה המאפשרת להקים שרתי HTTP לוקאלים על המחשב בצורה פשוטה, ספרייה זו מאפשרת לשלב Frontend ו- Backend הרשום ב Python בצורה נוחה.
- **DoS Hulk** ו-**DoS Goldeneye** - כלים נפוצים אשר מאפשרים לבצע התקפת DoS על שרתי HTTP במגוון דרכים. כלים אלו משומשים בעיקר על ידי pen-testers כדי לחפש חולשות בשרתי HTTP.

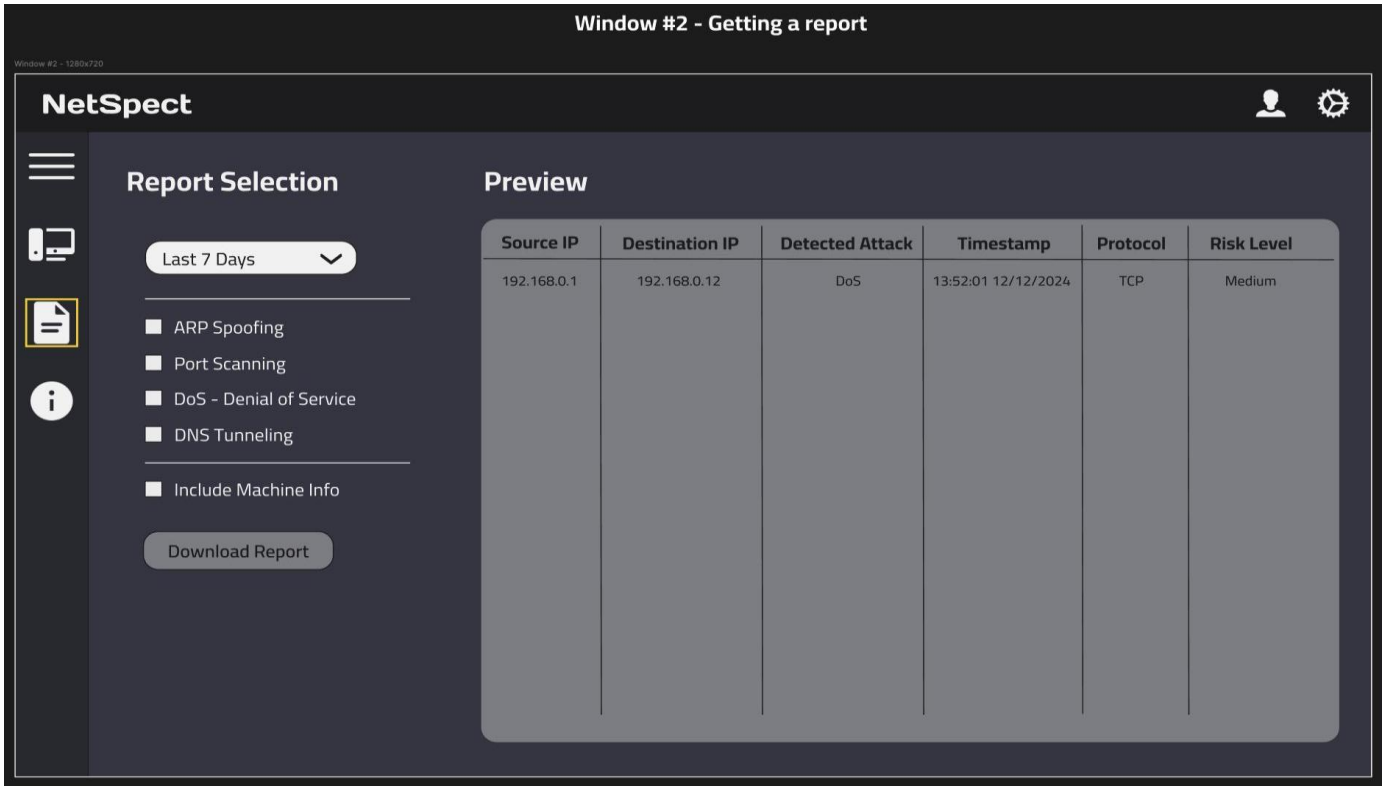
❖ **Nmap** - כלי המאפשר לבצע התקפות מסוג Port Scan על מכשירים שונים המחוברים לתקשורת, כלי זה משומש בעבור איסוף מידע וריגול על מכשירים ברשת וגם עבור חיפוש חולשות במכשירים אלו.

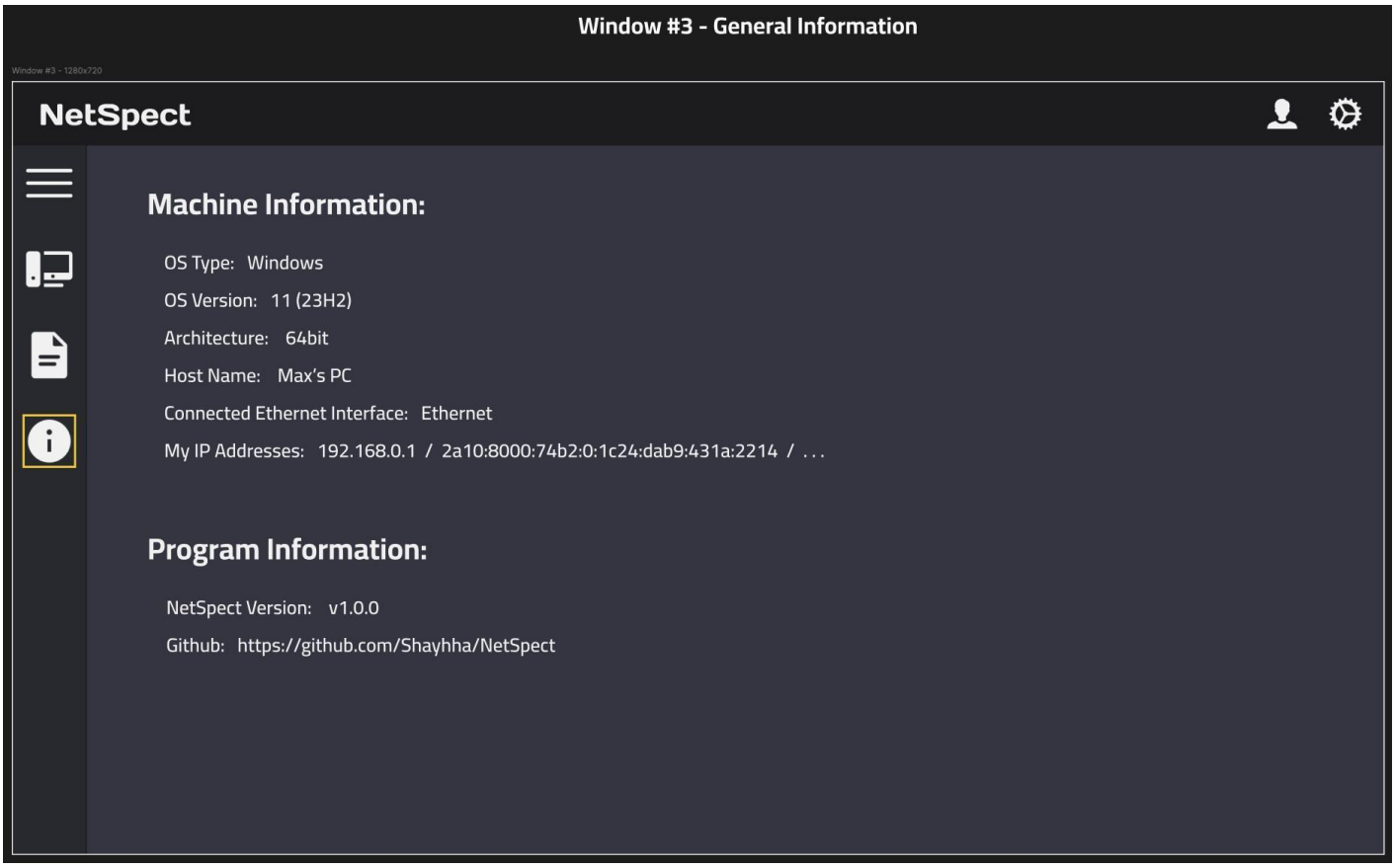
4. דיאגרמת **WBS**:

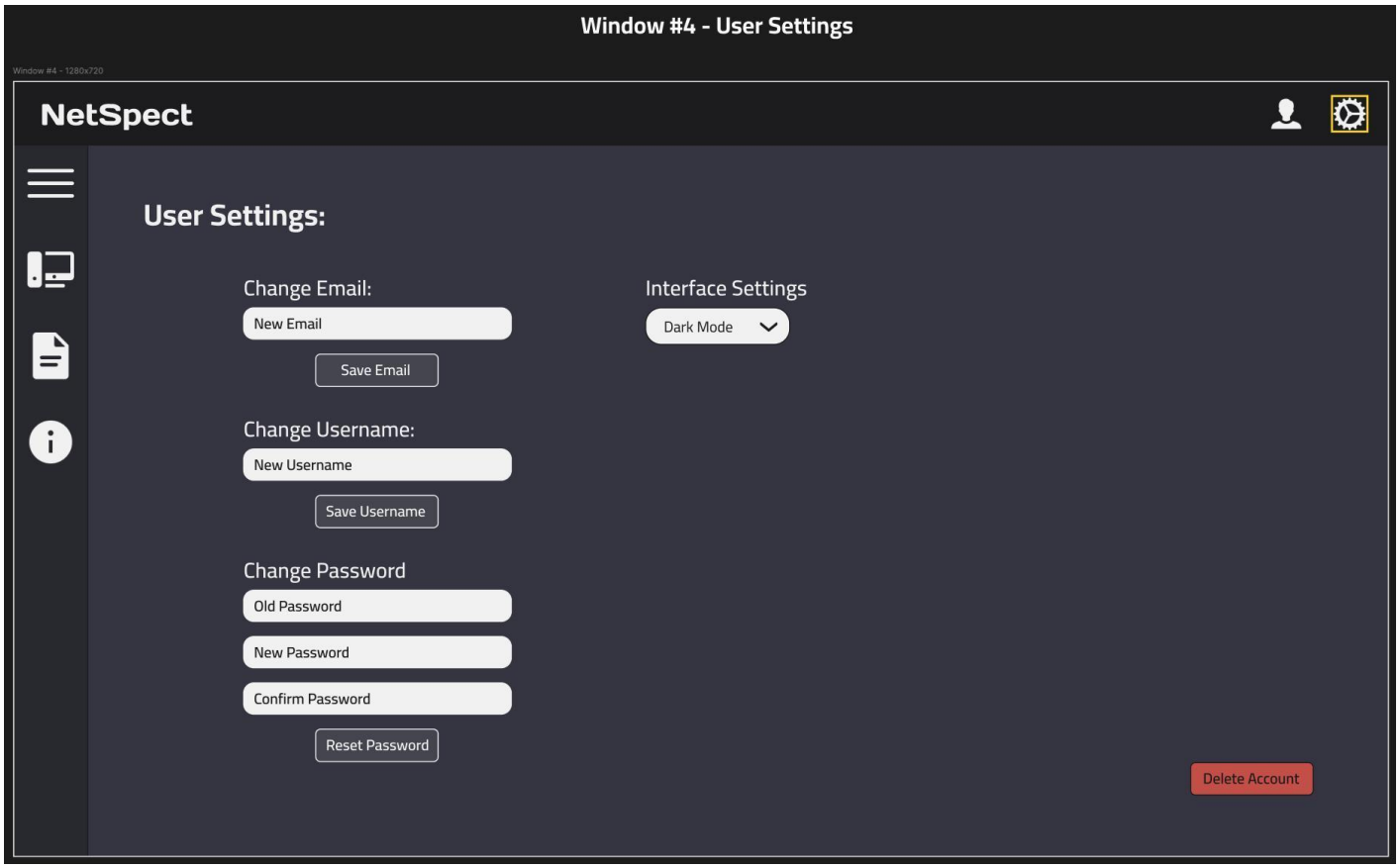


5. תרשימי מסכים:

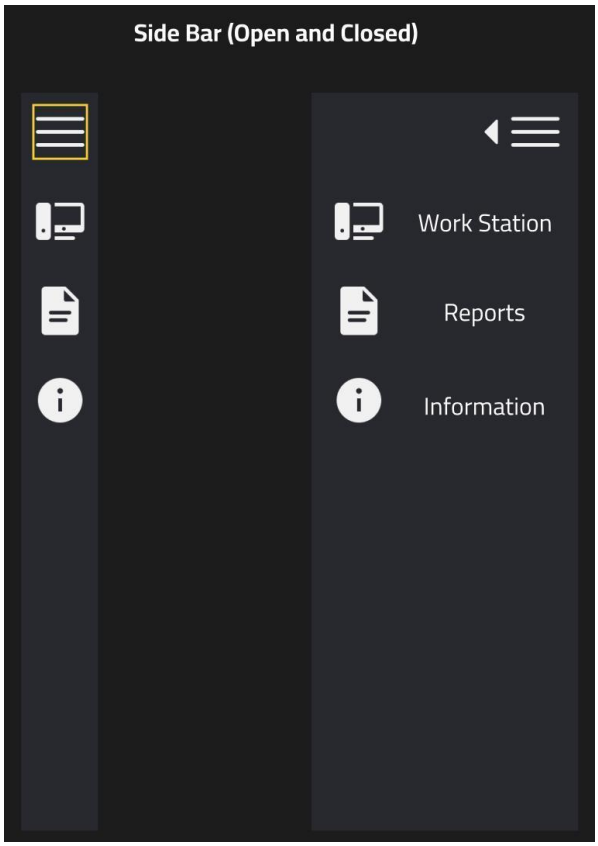


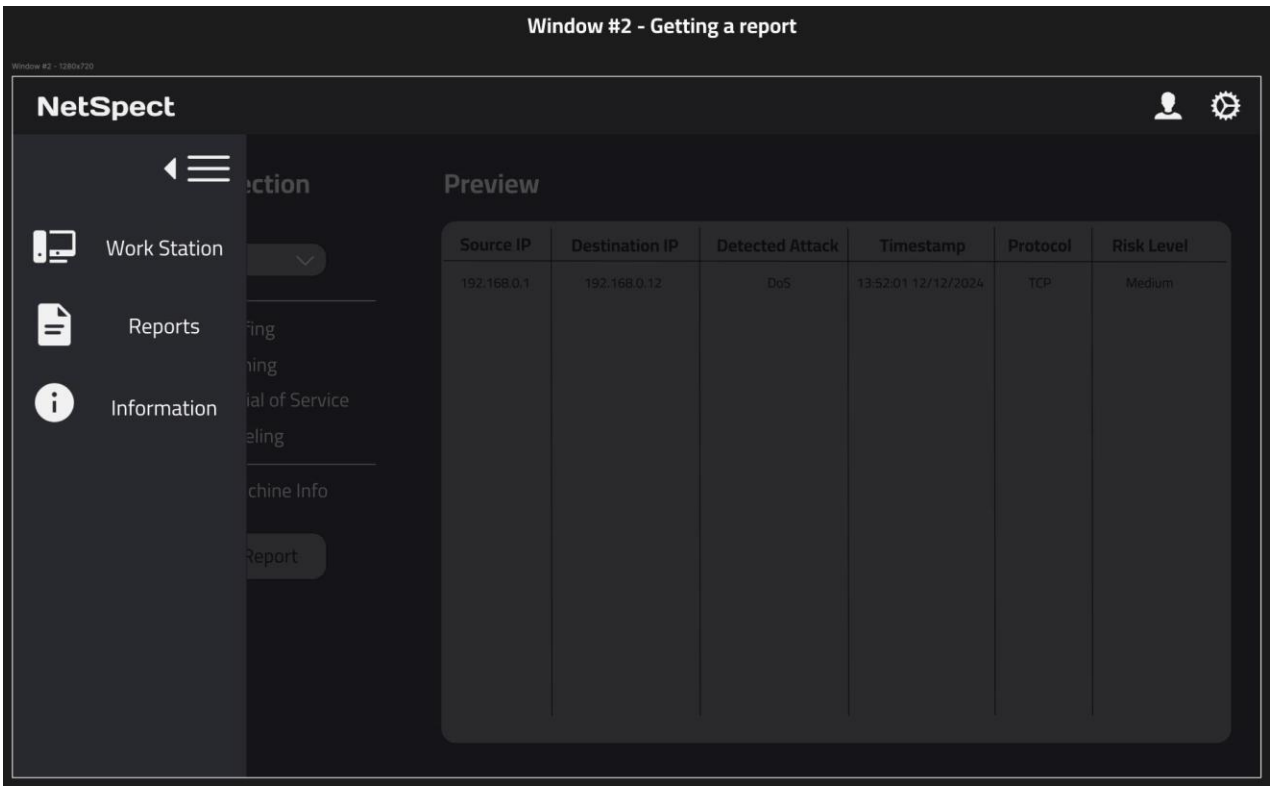


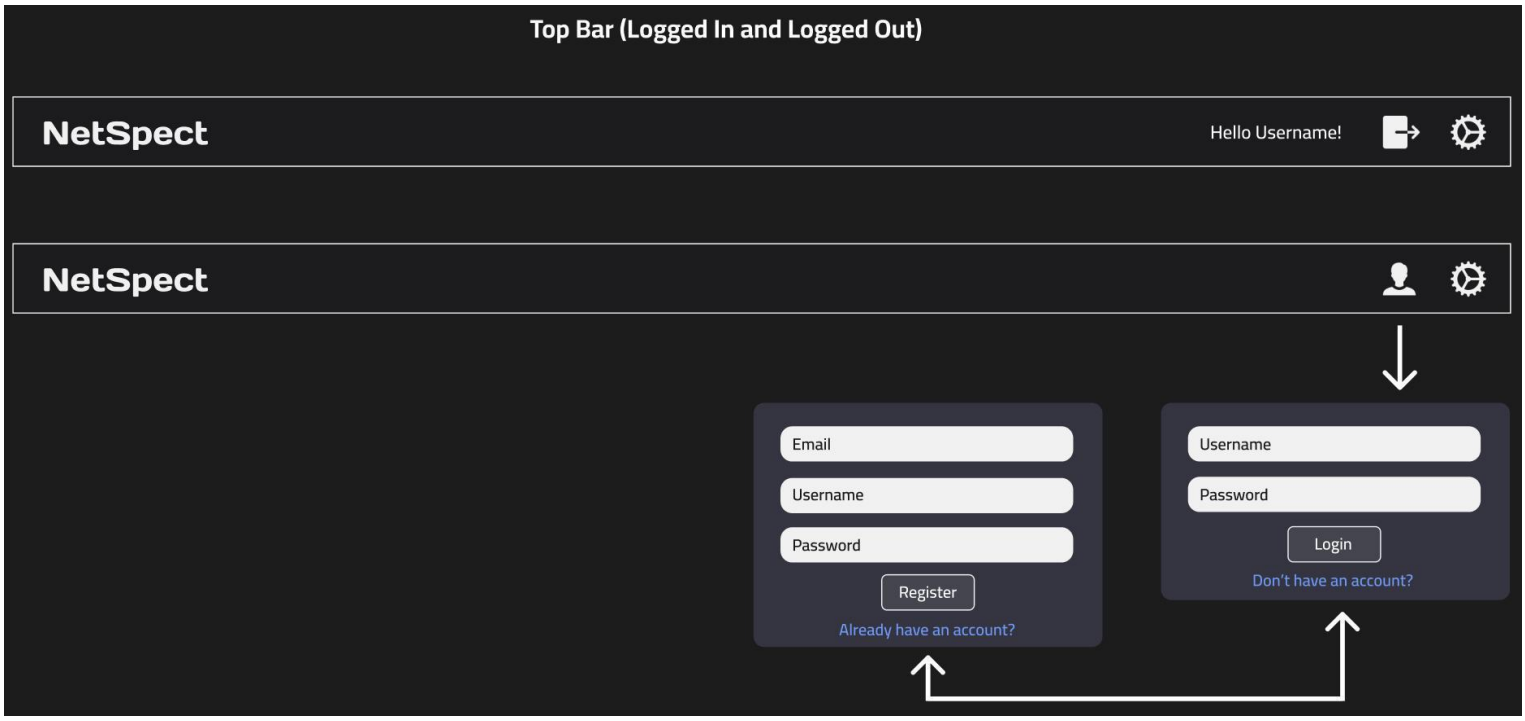

Window #3 - General Information
Window #3 - 1280x720
NetSpect
Machine Information:
OS Type: Windows
OS Version: 11 (23H2)
Architecture: 64bit
Host Name: Max's PC
Connected Ethernet Interface: Ethernet
My IP Addresses: 192.168.0.1 / 2a10:8000:74b2:0:1c24:dab9:431a:2214 / . . .
Program Information:
NetSpect Version: v1.0.0
Github: https://github.com/Shayhha/NetSpect


Window #4 - User Settings
Window #4 - 1280x720
NetSpect
User Settings:
Change Email:
New Email
Save Email
Change Username:
New Username
Save Username
Change Password
Old Password
New Password
Confirm Password
Reset Password
Interface Settings
Dark Mode
Delete Account

להלן השימוש בכפתור הרשמה וכפתור התפריט:






Top Bar (Logged In and Logged Out)
NetSpect
Hello Username!
NetSpect
Email
Username
Password
Register
Already have an account?
Username
Password
Login
Don't have an account?

דרישות המערכת

משתמש אורח:

1. הפעלת מערכת הזיהוי
2. כיבוי מערכת הזיהוי
3. הרשמה למערכת כמשתמש חדש
4. קבלת התראות על התקפות סייבר בתקשורת בזמן אמת
5. צפייה בהיסטוריית התראות מאז ההפעלה האחרונה של המערכת
6. קבלת מידע כללי על המערכת
7. הוצאת דוח על היסטוריית התקפות
8. מיון התראות לפי סוג ותקופה
9. צפייה במידע אנליטי אודות ההתראות לפי שנה
10. ביצוע פעולות בתוכנה דרך ה- Tray Icon
11. שחזור סיסמה בעזרת הודעה הנשלחת לכתובת המייל
12. הוספת כתובת ל- MAC Blacklist למניעת התראות
13. החלפה בין מצבים Light Mode ו- Dark Mode

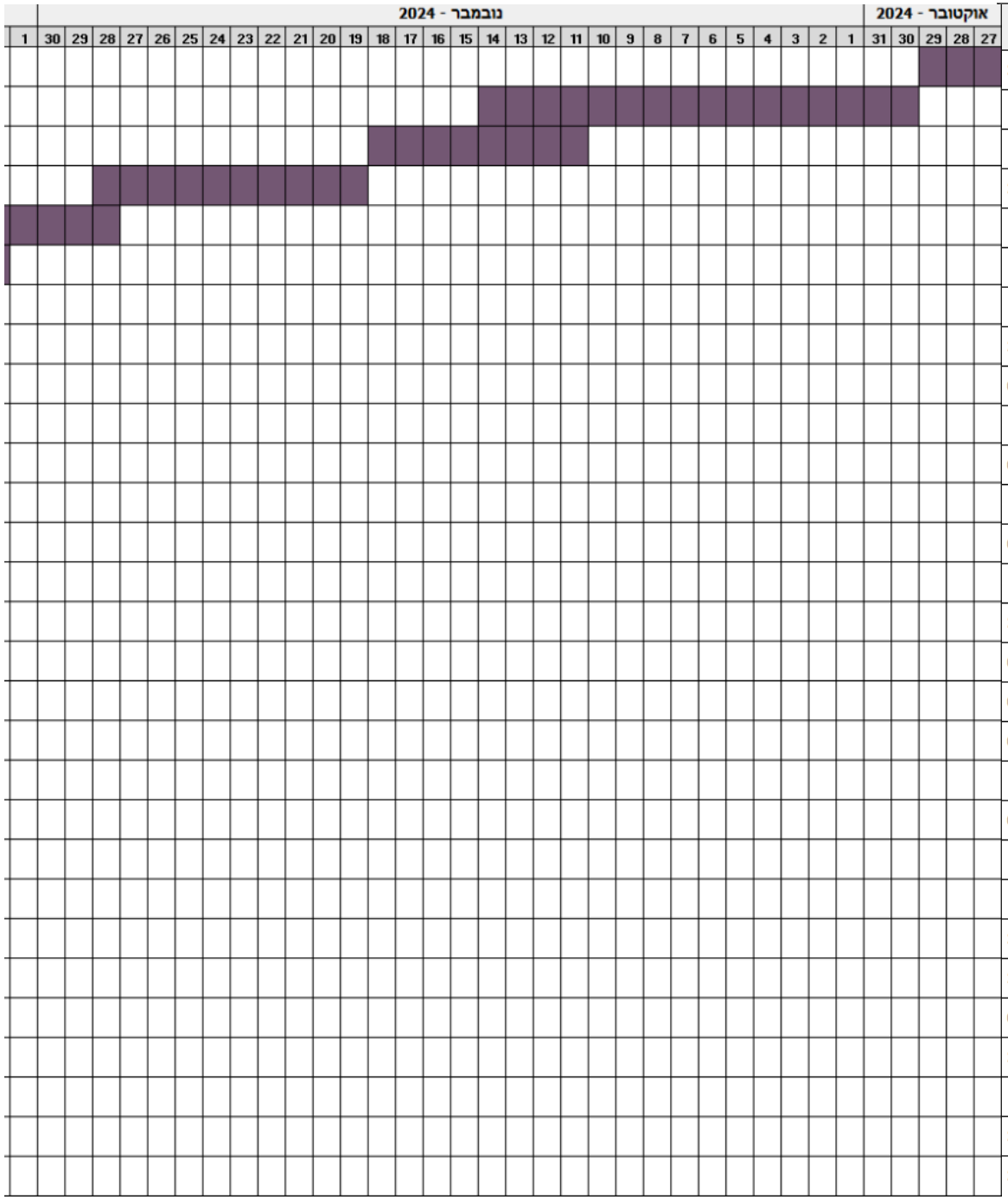
משתמש מחובר:

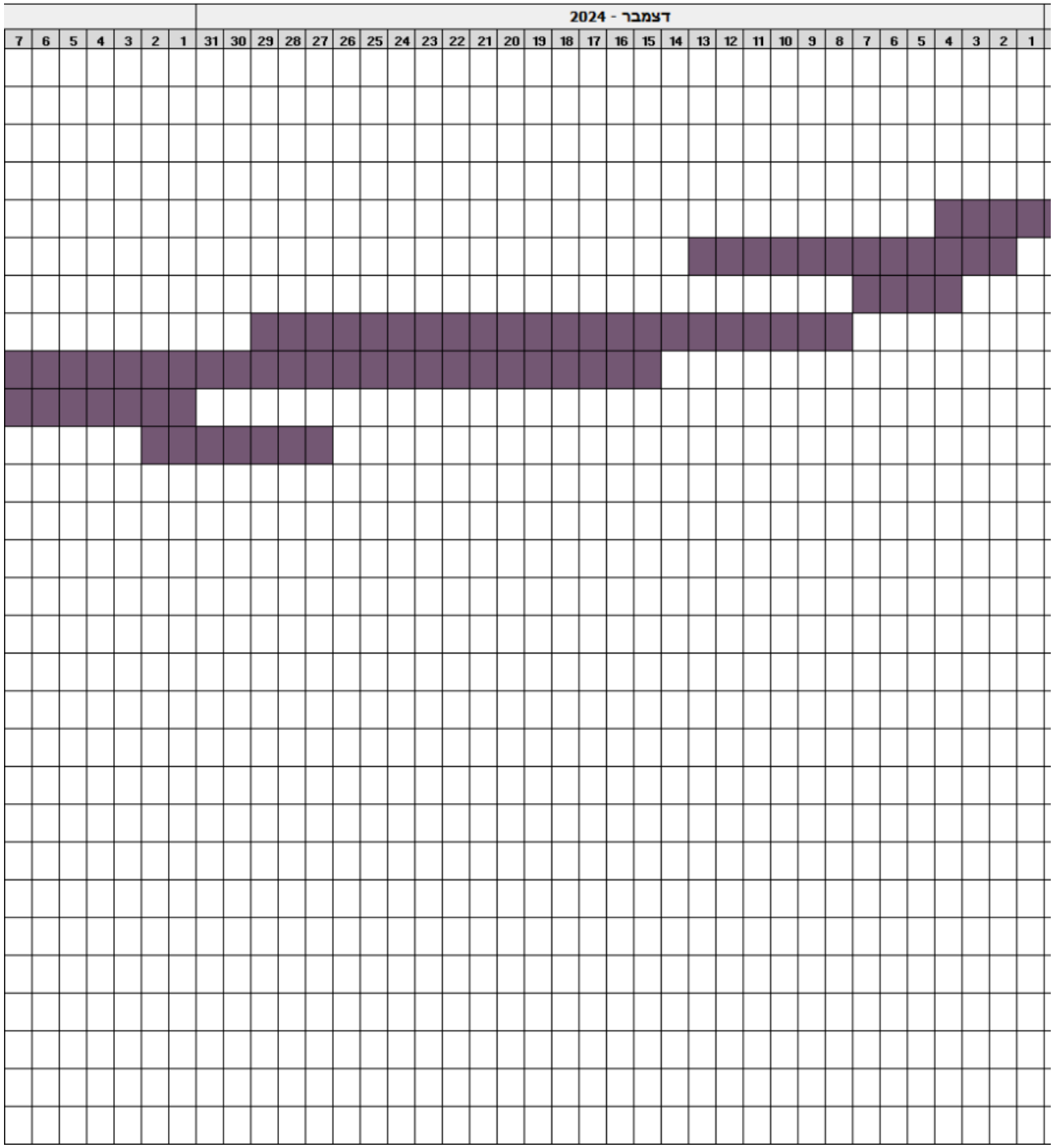
14. התחברות למשתמש
15. התנתקות מהמשתמש
16. שינוי פרטים אישיים בפרופיל
17. מחיקת משתמש
18. מחיקת התראות
19. הפעלת מערכת הזיהוי
20. כיבוי מערכת הזיהוי
21. קבלת התראות על התקפות סייבר בתקשורת בזמן אמת
22. צפייה בהיסטוריית מלאה של התראות
23. קבלת מידע כללי על המערכת
24. הוצאת דוח על היסטוריית התקפות
25. הוצאת דוח המכיל סטטיסטיקות על התראות קודמות
26. הוצאת דוח על נתונים המערכת ואופן השימוש בתוכנה
27. מיון התראות לפי סוג ותקופה
28. צפייה במידע אנליטי אודות ההתראות לפי שנה

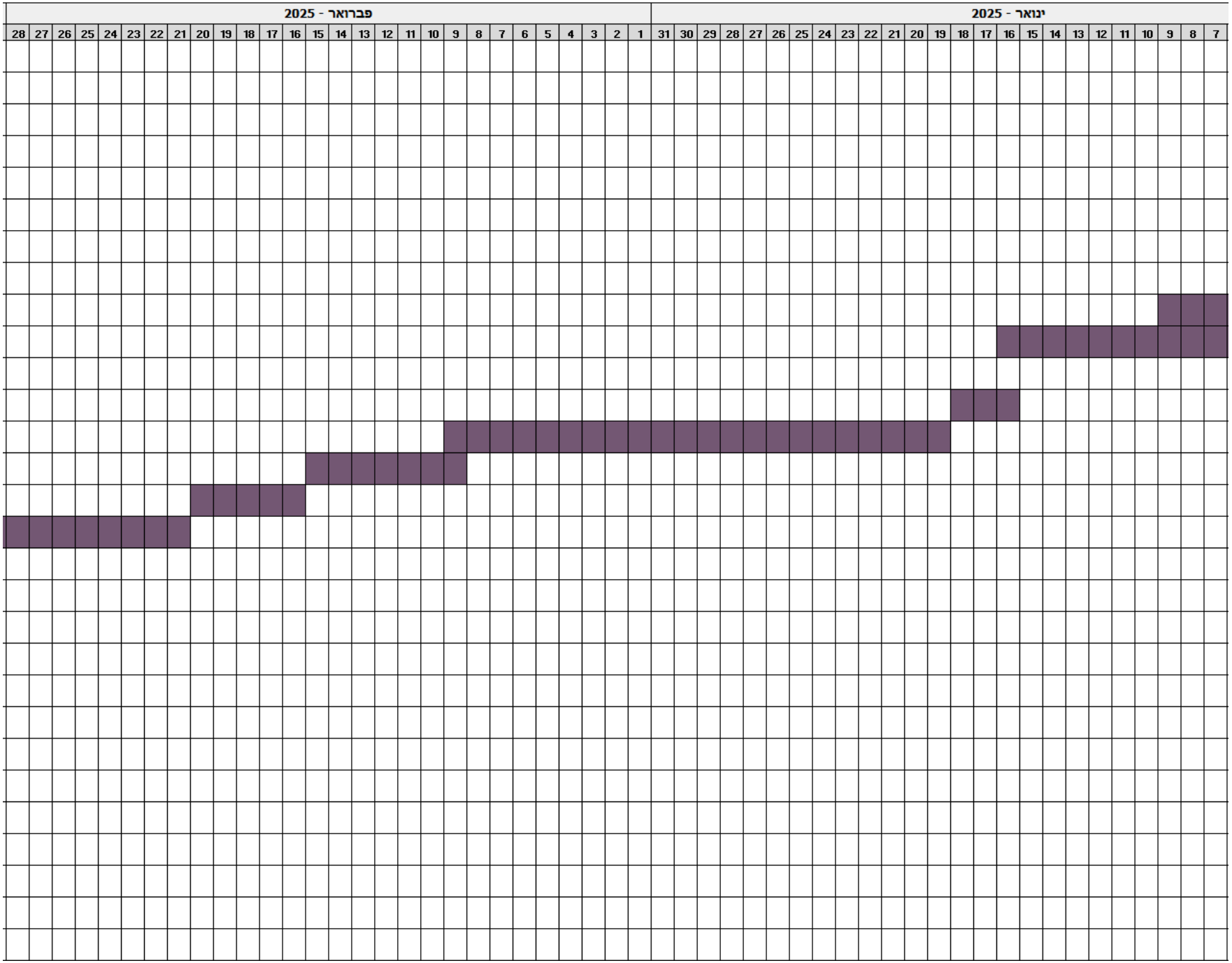
29. ביצוע פעולות בתוכנה דרך ה- Tray Icon
30. שחזור סיסמה בעזרת הודעה הנשלחת לכתובת המייל
31. הוספת כתובת ל- MAC Blacklist למניעת התראות
32. החלפה בין מצבים Light Mode ו- Dark Mode
33. החלפה בין מצבים Detection Mode ו- Collection Mode

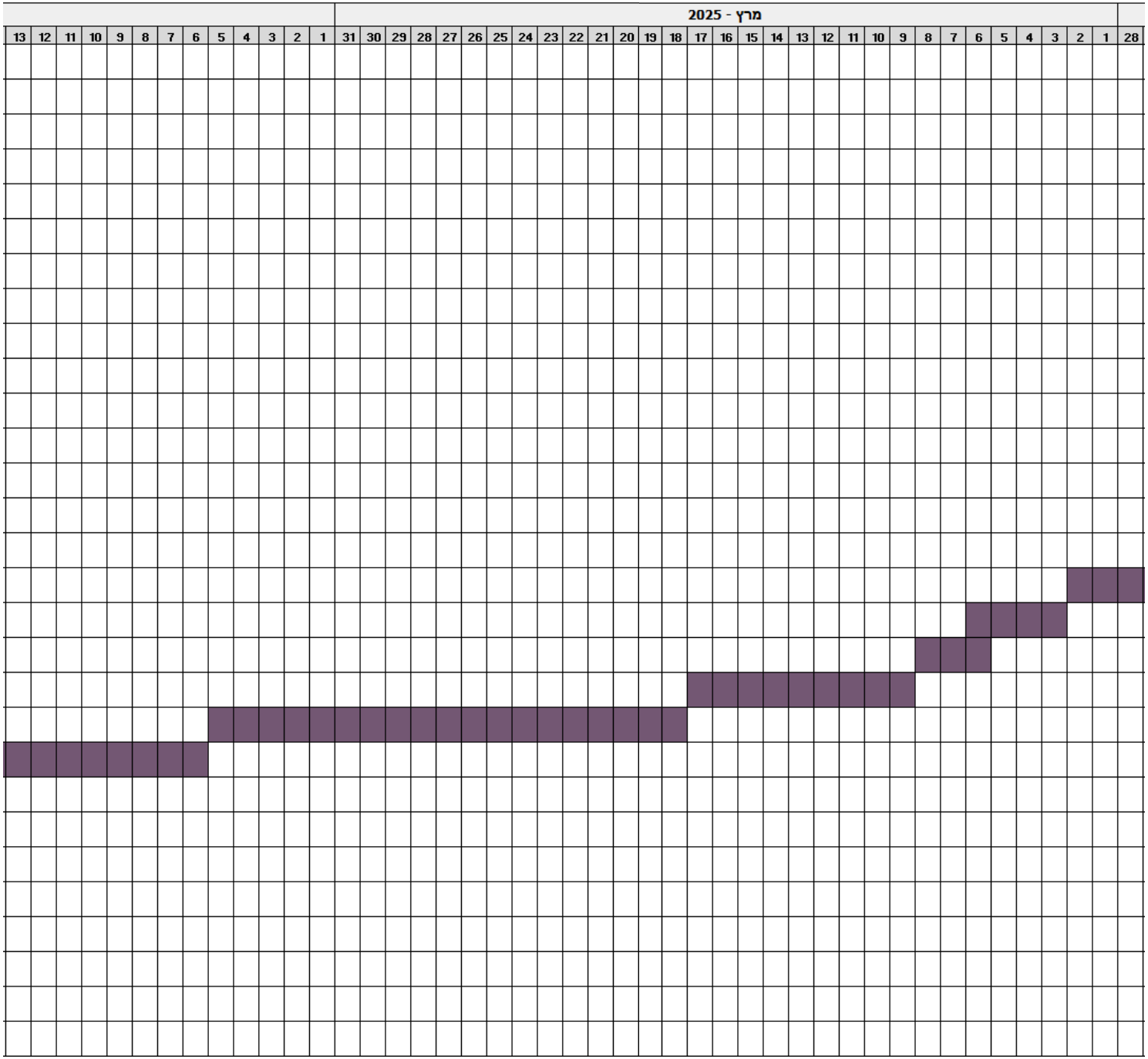
תרשים Gantt

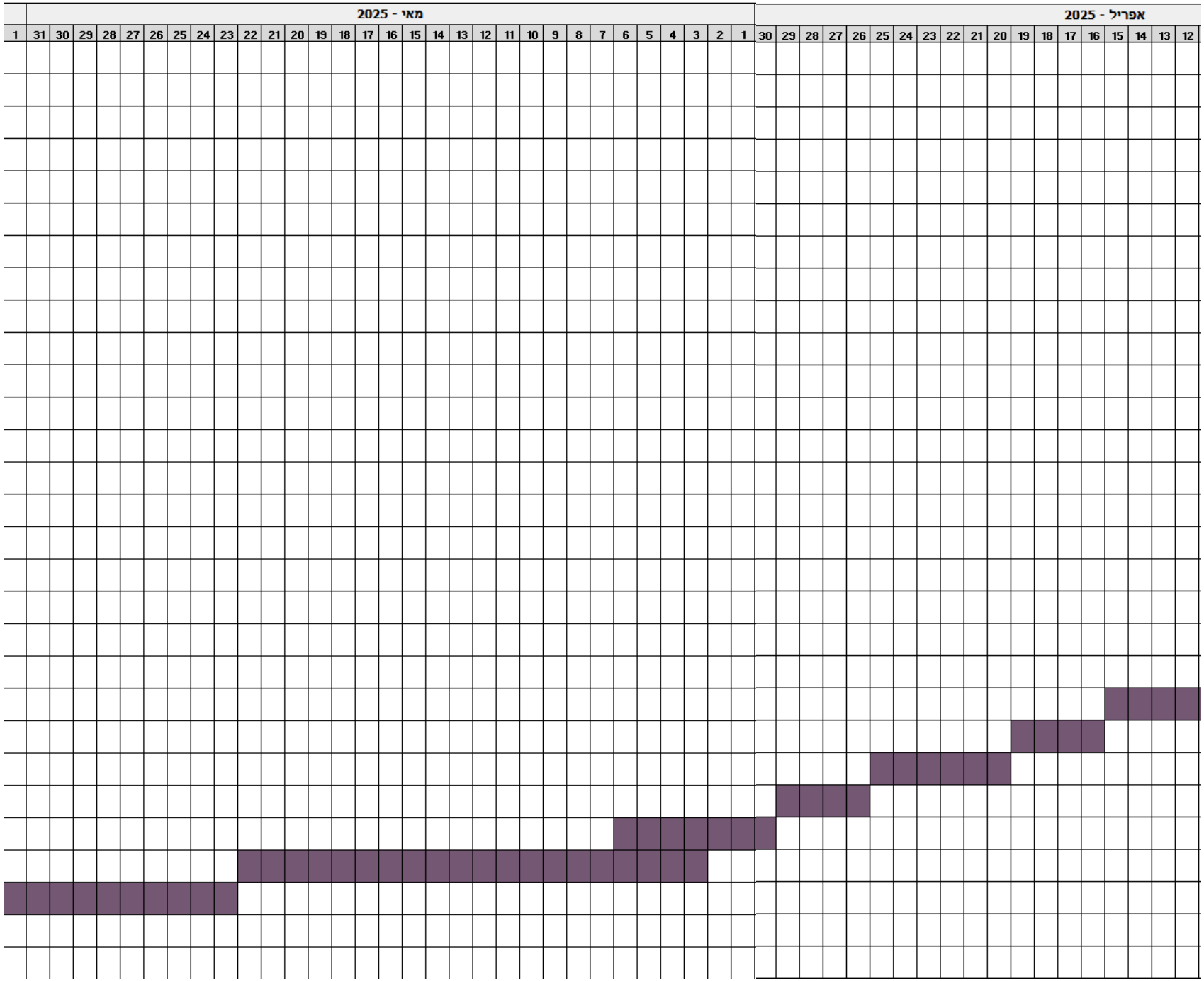
פעילות	תאריך התחלה	תאריך סיום
בחירת הנושא לפרויקט והגדרת תוכלת הפרויקט	27/10/2024	29/10/2024
ביצוע סקר ספרות	30/10/2024	14/11/2024
ביצוע סקר שוק	11/11/2024	18/11/2024
כתיבת סמך ייזום + דרישות + אפיון + ניהול סיכונים	19/11/2024	28/11/2024
פיתוח הדרישה לניטור תעבורת התקשורת בזמן אמת	28/11/2024	4/12/2024
פיתוח הדרישה של אלגוריתם לזיהוי התקפת ARP Spoofing	2/12/2024	13/12/2024
איסוף datasets מוכנים על התקפות סייבר בתקשורת	4/12/2024	7/12/2024
איסוף נתונים שלנו לשיפור הלמידה של המודלים	8/12/2024	29/12/2025
פיתוח מודל לזיהוי התקפת Port Scanning	15/12/2024	09/01/2025
פיתוח מודל לזיהוי התקפת DoS	1/1/2025	16/1/2024
הכנת מצגת לועדה מלווה 1	27/12/2025	02/01/2025
תיקון והגשה של הדו"ח לאחר קבלת הערות בוועדה	16/01/2025	18/01/2025
פיתוח מודל לזיהוי התקפת DNS Tunneling	19/01/2025	09/02/2025
פיתוח מסך ראשי בתוכנה (הפעלה וכיבוי המערכת)	09/02/2025	15/02/2025
הקמת בסיס נתונים עבור משתמשים	16/02/2025	20/02/2025
פיתוח מערכת התחברות, הרשמה, שינוי פרטים, מחיקה	21/02/2025	02/03/2025
פיתוח מסך למידע כללי על המערכת	03/03/2025	06/03/2025
כתיבה והגשה של תקציר הפרויקט באנגלית	06/03/2025	08/03/2025
פיתוח מסך להצגת התראות בזמן קבלת התראה על התקפה	09/03/2025	17/03/2025
פיתוח מסך להצגת היסטוריית התראות	18/03/2025	05/04/2025
אינטגרציה בין המודלים והאלגוריתמים לממשק משתמש	06/04/2025	15/04/2025
שימוש בריבוי תהליכים על מנת לשפר זמני ריצה	16/04/2025	19/04/2025
פיתוח מסך להוצאת דוחות	20/04/2025	25/04/2025
פיתוח פונקציונליות בסיס נתונים להוצאת דוחות	26/04/2025	29/04/2025
הכנת מצגת לועדה מלווה 2	30/04/2025	06/05/2025
ביצוע בדיקות פונקציונליות על המערכת בכל מערכת הפעלה	02/05/2025	22/05/2025
כתיבה והגשה של פוסטר	23/05/2025	10/06/2025
כתיבה והגשה של דוח מסכם + עותק לארכיון המחלקתי	11/06/2025	22/06/2025
השתתפות בכנס הפרויקטים המחלקתי	23/06/2025	23/06/2025

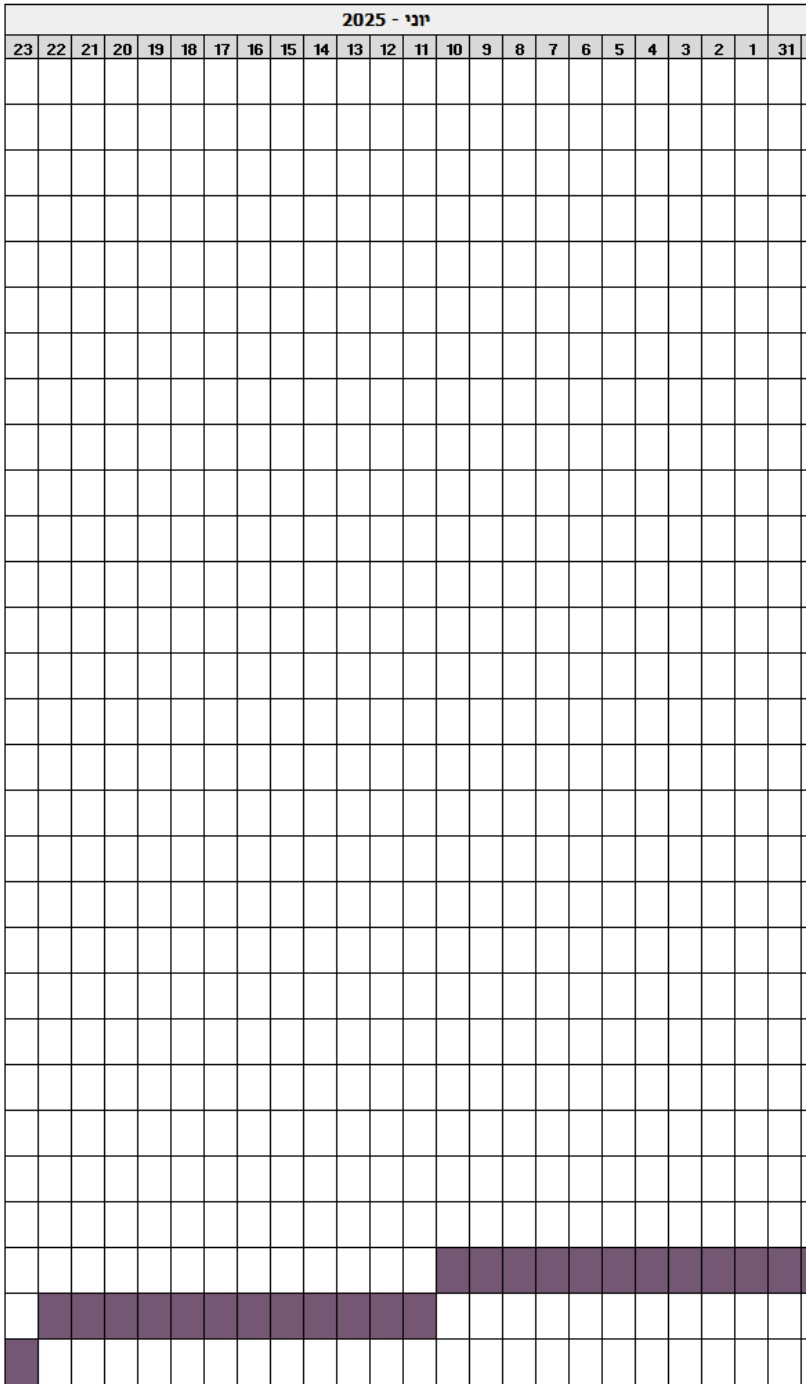



דצמבר - 2024
31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1
7 6 5 4 3 2 1


פברואר - 2025
ינואר - 2025
28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1
31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7


מרץ - 2025
13 12 11 10 9 8 7 6 5 4 3 2 1 31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 28


מאי - 2025
אפריל - 2025
1 31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1
30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12


יוני - 2025
23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 31

מדדי הצלחה

ממדי ההצלחה שלנו בפרויקט כוללים:

- סיום הפרויקט בזמן המוקצב לו
- היכולת לזהות התקפות בזמן אמת
- היכולת לזהות את סוג התקפת התקשורת במדויק
- ממשק משתמש פשוט וקל לשימוש
- עמידה בדרישות שהוגדרו עבור הפרויקט
- קוד קריא, ברור ומסודר
- יציבות המערכת בזמן עומס
- אבטחת מידע גבוהה של נתוני המשתמשים

במידה ונשיג את כל המטרות האלו נוכל לדעת שהצלחנו בפרויקט.

עלויות צפויות

במידה והיה מדובר בפרויקט אשר יוצא לשוק, העלויות הצפויות בפיתוח תוכנת IDS זו אשר מכילה מודלים של למידת מכונה, היות כוללים עלויות שכר למפתחים ומהנדסים עם מומחיות בלמידת מכונה ואבטחת מידע, עלויות תשתית ועיבוד נתונים כמו שרתים ומשאבי ענן לאימון המודלים ועיבוד בזמן אמת, רישוי של ספריות וכלים חיצוניים ללמידת מכונה ותמיכה ברב-פלטפורמות, בדיקות ואימות במערכות הפעלה שונות, וביצוע סימולציות של התקפות מגוונות בתקשורת מחשבים. בנוסף, יש להביא בחשבון עלויות תחזוקה ועדכון שוטף של המודלים, התאמתם להתקפות חדשות, והוצאות עקיפות כמו שיווק, תמיכה טכנית, ורישוי תוכנה. עם זאת, מכיוון שמדובר בפרויקט גמר שלנו, והיקף הפרויקט מצומצם בהשוואה לפיתוח תוכנות על-ידי חברות סייבר, העלויות הצפויות נמוכות משמעותית.

ניהול סיכונים

טבלת רמות חומרת הסיכון:

רמה	דירוג	פירוט
נמוכה מאוד	1	עלול לגרום נזק מינימלי, לא משפיעה על תפעול המערכת. קשור לחלק לא קריטי של המערכת. יכול להיות שתהיה דרישה שלא תסופק בזמן.
נמוכה	2	נזק קל, פוגע בחלק מסויים וקטן של המערכת כך שעדיין ניתן להשתמש בה. לא כל הדרישות יסופקו בזמן.
בינונית	3	נזק בינוני, פוגע בצורה ברורה בתפעול המערכת, מפריע לפעילות תקינה ומלאה במערכת ממבט המשתמשים. לא כל הדרישות יסופקו בזמן.
גבוהה	4	נזק חמור. ייתכן חריגה מהמועד המתוכנן לסיום פיתוח המערכת. המוצר לא תקין. ניתן יהיה לעבוד רק עם חלק מהמערכת.
גבוהה מאוד	5	נזק חמור מאוד, המועד המתוכנן לסיום פיתוח לא יוסג, דרישות קריטיות לא תסופקנה ולא יתאפשר לעבוד עם חלק גדול מהמערכת.

טבלת רמות סבירות הסיכון:

רמה	סבירות (%)	פירוט
נמוכה מאוד	0.0-0.2	נדיר שיקרה
נמוכה	0.21-0.4	יכול לקרות בסיכוי נמוך
בינונית	0.41-0.6	עלול לקרות
גבוהה	0.61-0.8	כנראה יקרה
גבוהה מאוד	0.81-1.0	בלתי נמנע שיקרה

טבלת הסיכונים:

#	פירוט הסיכון	ההשפעות של הסיכון על הפרויקט	רמת הנזק שהסיכון יגרום	רמת הסבירות שהסיכון יתרחש	תוחלת הסיכון (חומרה x סבירות)	ההתמודדות
1	אחד המודלים לא מצליח לזהות התקפה מסוג כלשהו למרות שאומן על נתונים של התקפה זו.	המערכת לא תזהה התקפות מסוג זה על המכשירים של הלקוחות וזה יגרום להם לעבור למוצר מתחרה.	5	0.45	2.25	עלינו למצוא או להכין מודל חדש לגמרי או לבצע fine-tuning למודל הקיים כך שיהיה מסוגל לזהות את ההתקפה, או לחילופין למוצא data חדש וללמד עליו את המודל כדי לשפר את יכולת הזיהוי שלו. אם כל אלו לא יעבדו ניתן לעבור ל-signature-based algorithms עבור התקפה זו.
2	אחד המודלים מזהה התקפה וגם מזהה באופן שגוי תעבורת תקשורת תקינה.	התראות חוזרות על התקפות למרות שאין התקפות בפועל יגרמו לכך שמשתמשים יאבדו אמון במוצר ויפסיקו להשתמש בו.	4	0.75	3	לשפר את המודל הקיים על ידי הוספה או הסרת features, ללמד את המודל על data אחר או לנסות מודל למידת מכונה אחד. אם כל אלו לא יעבדו ניתן לעבור ל-signature-based algorithms עבור התקפה זו.
3	אחד האלגוריתמים לא מצליח לזהות את ההתקפה	המערכת לא תזהה התקפות מסוג זה על המכשירים של הלקוחות וזה יגרום להם לעבור למוצר מתחרה.	5	0.5	2.5	עלינו לחקור שוב על אופן ביצוע ההתקפה ולזהות למה האלגוריתם לא תקין, לנסות לתקן אותו או לחילופין לנסות להחליף אותו במודל.
4	איסוף הנתונים מתבצע באופן שגוי (עבור ה-datasets)	המודלים שנבנה לא ילמדו טוב איך לזהות התקפות או יובילו לרמה גבוהה של False Positive	4	0.55	2.2	לחקור על המידע הרלוונטי לנו לפי מאמרים, לבדוק שהמידע שנאסף הוא תקין ולא חסר בו כלום וששיטת האיסוף נכונה.
5	ממשק משתמש לא אינטואיטיבי	משתמשים עלולים	2	0.35	0.7	עלינו להשתמש בעקרונות מוכרים

	ולא ברור למשתמש קצה	להתקשות בהבנת אופן השימוש במערכת, מה שיכול להוביל לתסכול, שימוש שגוי, או זניחת המערכת.				לעיצוב, כגון עקביות, פשטות, והפחתת עומס מידע. בנוסף ביצוע שיפור מתמיד של הממשק בהתאם למשוב משתמשים.
6	זמן התגובה של המערכת לא ברמת "זמן אמת"	משתמשים עשויים לחוות חווית שימוש שלילית מפני שלא יקבלו התראה מאוחרת ובעקבות זאת יחשפו להתקפת סייבר לאורך זמן ממושך.	3	0.5	1.5	אופטימיזציית הקוד על ידי שיפור אלגוריתמים ומבני נתונים כדי לייעל את מהירות הביצוע, וביצוע בדיקות סימולציה בתנאי עומס.
7	אי עמידה בדרישות וציפיות של הפרויקט	משתמשים עלולים לחוות חוויות כמו תכונות חסרות וביצועים לא מספקים מה שיגרום להם לעבור למתחרים.	2	0.30	0.6	קביעת פגישות חוזרות כל שבוע עם חברי הצוות על מנת להבין איפה תהליך הפיתוח עומד, מה הם הבעיות ומה ניתן לעשות על מנת לפתור אותם.
8	אי עמידה בזמנים עבור פיתוח התוכנה	המשתמשים יקבלו תוכנה לא גמורה אשר לא נבדקה ב100%, מה שיכול להוביל להרבה בעיות ובאגים ולכך שהמשתמשים יפסיקו להשתמש בתוכנה שלנו.	1	0.25	0.25	עלינו לפתח את התוכנה על בסיס שבועי, לקבוע לעצמנו deadlines על מנת לסיים את הפרויקט בטווח הזמנים שהוגדר, במידה ואנחנו נתקעים עלינו מייד לבקש עזרה ולמנוע בזבוז זמן יקר.
9	בעיה באינטגרציה של התוכנה לסוגים השונים של מערכות ההפעלה	חוסר היכולת להשתמש בתוכנה במערכות הפעלה מסויימות יוביל	3	0.4	1.2	עלינו בדיקות מקיפות על מערכות הפעלה שונות בשלבי הפיתוח. בנוסף עלינו להשתמש בממשקים וספריות

		לאובדן משתמשים או חוויית משתמש שלילית במידה ויש חוסר פונקציונליות.				התומכים ברב-פלטפורמות.
10	ניטור תקשורת בצורה לא נכונה, למשל בחירת flows בכיוון שונה לכיוון השליחה	קיים סיכוי להפחתת היכולת לזהות את ההתקפות או לחילופין הגברת כמות זיהוי שווא של האלגוריתמים והמודלים.	3	0.5	1.5	לנסות בחירת כיוון ה-flows בכיוון הפוך ובדיקת התוכנה שוב, במידה לא עוזר כנראה הבעיה לא בכיוון ה-flows.

תוצאות הפרויקט

לפרויקט שלנו שני חלקים, החלק הראשוני עסק במחקר של מספר התקפות סייבר בתקשורת, ביניהן ARP Spoofing, Port Scanning, DoS ו- DNS Tunneling. ואילו החלק השני עוסק בבניית תוכנה שתדע לזהות ולהתריע בזמן אמת על אותם התקפות סייבר. במסגרת הפרויקט ערכנו מחקר מעמיק בטכניקות קיימות לזיהוי המתקפות האלו, השיטות והכלים לביצוע המתקפות ואופן הפעולה שלהם.
בנוסף, כפי שנרחיב בהמשך, את הנתונים ששימשו לאימון המודלים שלנו לזיהוי המתקפות האלו נאספו על ידינו. בנוסף לכך, תהליך העיבוד המקדים על תעבורת התקשורת ופעולות ה-Feature Selection שכללו חישובים קריטיים, גם כן נעשה ונבנה על ידינו ללא שימוש בכלים חיצוניים. מעבר לכך, התוכנה שפיתחנו פועלת באופן עצמאי, ללא תלות בכלים חיצוניים או שירותים צד שלישי.

<u>מהמחקר שלנו עלו מספר נקודות חשובות:</u>

- ❖ רוב המחקרים שסקרנו עוסקים בשיטות לזיהוי התקפות אלו, אך לא שמים דגש על זיהוי בזמן אמת.
- ❖ מאגרי נתונים קיימים אינם טובים מספיק עבור לימוד מודלים אשר מסוגלים לזהות את התקפות אלו בזמן אמת.
- ❖ זיהינו מספר בעיות מרכזיות במאגרים קיימים כמו פיצ'רים שחושבו בצורה לא נכונה, שכפול מידע בין מספר גדול של פיצ'רים וחלק מהשדות תמיד חסרים.
- ❖ ההתקפות שבחרנו עבור פרויקט זה הם ההתקפות משמעותיות אשר יכולות לגרום לנזקים כבדים בקלות ולכן הזיהוי שלהם אכן קריטי.
- ❖ מצאנו כי ניתן לשפר את האלגוריתמים הקיימים לזיהוי התקפת ARP Spoofing ולהפוך אותם לעמידים יותר ברשתות תקשורת מורכבות.

<u>זיהוי ARP Spoofing:</u>

מחקר זה אפשר לנו לפתח אלגוריתם זיהוי מבוסס חתימות להתקפות ARP Spoofing, אשר מנתר את תעבורת התקשורת ומזהה אי התאמה בין כתובות MAC ל- IP וליהפך. אי התאמה בין הכתובות מהווה אינדיקציה על התרחשות ההתקפה אך אינה ודאית. על מנת לזהות את ההתקפה במדויק, האלגוריתם שלנו משלב מנגנון אימות כתובות IP באמצעות שליחה וקבלה של פאקטות ARP. מטרת המנגנון היא לאמת את התאמת כתובת ה- IP לכתובת MAC יחידה, וזיהוי של יותר ממכשיר אחד המשיב עבור אותה כתובת IP, או מענה מכתובת MAC שאינה תואמת לזו שנאספה במהלך סריקת הרשת,

מהווים סימן לחשד להתחרשות התקפת ARP Spoofing. כלומר לכל כתובת IP חייבת להיות משוייכת לכתובת MAC אחת בדיוק.

כדי להימנע מ- False Positives, האלגוריתם שלנו תומך בסיטואציות בהן כתובת IP של מכשיר מסוים משתנה על ידי ה- DHCP Server, נפוץ עבור כתובות IP דינמיות, על ידי כך שהוא משתמש בטבלאות ARP הפוכות כדי לבדוק האם כתובת Source MAC מסוימת הייתה משוייכת בעבר לכתובת IP אחרת, במידה וכן אנו מוחקים את השיוך הישן של ה Source MAC ומשייך אותו לכתובת IP החדשה רק לאחר ביצוע אימות לכתובות ה- IP הזאת.

בנוסף, האלגוריתם שפיתחנו יודע לזהות התקפות ARP Spoofing על כמה Subnets בו זמנית בצורה נכונה על ידי סיווג של כתובות IP ל- Subnet המתאים. ולכן אפילו ברשתות גדולות ומורכבות, אשר משתמשות ב- Network Segmentation, שבהם מכשירים יכולים לקבל יותר מכתובת IP אחת, ובכך להיראות כמו התקפת ARP Spoofing, האלגוריתם שלנו יסווג את כתובות ה- IP בצורה נכונה לפי Subnet וימזער את כמות ה- False Positives.

פסאודו-קוד לאלגוריתם:

```
InitArpTables:
    for each subnet in network_interface:
        arp_tables[subnet] <- ArpTable(subnet)


DetectArpSpoofing:
    for each arp_packet in network_flows:
        subnet <- getSubnetForIP(arp_packet.source_ip)

        if subnet in arp_tables:
            arp_table_obj <- arp_tables[subnet]
            if arp_packet.source_ip not in arp_table_obj.arp_table:
                if arp_packet.source_mac in arp_table_obj.inverted_arp_table:
                    del arp_table_obj.arp_table[arp_packet.source_ip]
                    del arp_table_obj.inverted_arp_table[arp_packet.source_mac]
                answered_mac_addresses <- ValidateIpAddress(arp_packet.source_ip)
                if answered_mac_addresses != NULL:
                    if (len(answered_mac_addresses) == 1) && (arp_packet.source_mac == answered_mac_addresses[0]):
                        arp_table_obj.arp_table[arp_packet.source_ip] <- arp_packet.source_mac
                        arp_table_obj.inverted_arp_table[arp_packet.source_mac] <- arp_packet.source_ip
                    else:
                        DetectArpSpoofing(arp_packet.source_ip, arp_packet.source_mac)
            else:
                if arp_table[arp_packet.source_ip] != arp_packet.source_mac:
                    DetectArpSpoofing(arp_packet.source_ip, arp_packet.source_mac)

```

זיהוי Port Scanning ו- Denial of Service:

מהמחקר שביצענו עלה כי התקפות Port Scanning ו- DoS מאוד קרובות בהתנהגותן, ואיך שהן באות לידי ביטוי בתעבורת הרשת, וניתן לזהות אותן באמצעות מודל Multi-Class Classifier יחיד.

תחילה חקרנו על מאגרי נתונים (Datasets) קיימים, בדקנו את שיטת האיסוף שלהם, את התוכן שלהם ואיך השתמשו בהם במחקרים מדעיים, ומצאנו כי לרוב משתמשים בהם לצורך פיתוח מודלים של למידה עמוקה, מכיוון שהנתונים נאספים באופן סדור ומאפשרים זיהוי תבניות המתפתחות לאורך זמן. לכן מרבית המחקרים מתמקדים בשיטות זיהוי, ולא בזיהוי בזמן אמת.

מאחר שמטרתנו בפרויקט היא זיהוי התקפות בזמן אמת, בחרנו להשתמש באלגוריתמי למידת מכונה קלאסים, הידועים במהירותם וביעילותם ביחס למודלי למידה עמוקה. אך בניגוד למודלי למידה עמוקה, אלגוריתמים כמו SVM לא יודעים לעבוד עם נתונים סדורים, וזאת אחת הסיבות לכך המאגרים שהמובילים אינם מתאימים לפרויקט שלנו.

בנוסף, האופן שבו מחלקים את תעבורת התקשורת ל-flows במאגרים הקיימים מקשה על גילוי ההתקפות. כל flow, המייצג זוג מכשירים בתקשורת, מזוהה על־ידי מפתח ייחודי הכולל את כתובות ה-IP של שני הצדדים וגם מספר ה-Port שדרכו עוברת התקשורת. בגישה זו מתקבל ריבוי שורות רב, מכיוון שבעת סריקת פורטים נרשמת שורה נפרדת עבור כל פורט, ומכאן קושי לאתר דפוסי התקפה עקב פיזור המידע.

החוקרים בחרו בשיטה זו כדי לבנות מאגר נתונים ממותג בזמן (timestamped), המאפשר למודלי למידת עמוקה לזהות תבניות דינמיות של התקפה לאורך זמן. אולם, כפי שציינו קודם, למרות יעילותה, למידת עמוקה אינה תומכת בזיהוי מיידי בזמן אמת כפי שאנחנו מחפשים, ולכן שגם הנתונים הקיימים במאגרים המובילים אינם מתאימים לצרכי המודל שלנו. לאומת זאת, אנחנו בחרנו לייצג כל flow לפי כתובות ה-IP של שני הצדדים וגם כתובות ה- MAC, ואנחנו לא מתשמשים ב- Port.

סיבה נוספת לכן היא שזיהינו מספר בעיות מרכזיות במאגרים קיימים כמו: חלק משיטת האיסוף מידע אינה תקינה, חלק מהשדות תמיד חסרים (מסומנים כאפס), וקיים שכפול מידע בין מספר גדול של פיצ'רים (כלומר מספר עמודות עם שמות שונים וערכים זהים).

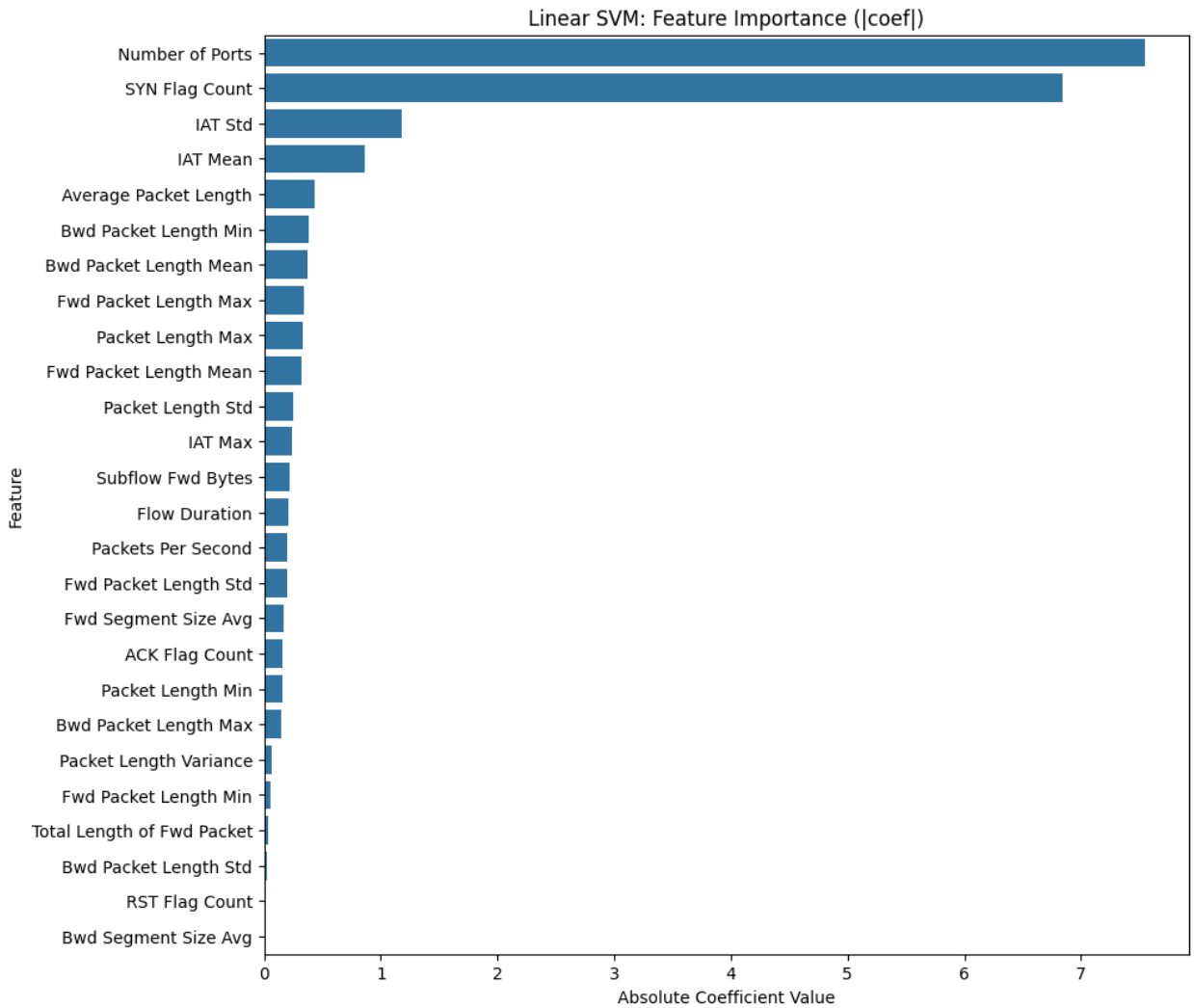
בהתאם למסקנות האלו, הסקנו שכדי לבנות מודל אופטימלי לזיהוי התקפות Port Scanning ו- DoS, היה עלינו לאסוף נתונים אודות תעבורת התקשורת באופן ידני, כלומר לאיסוף תעבורה תקינה ונגועה. לצורך איסוף תעבורת תקשורת נגועה יצרנו סביבת מעבדה שבה הרצנו התקפות ובנינו קוד שמייצר נתונים חדשים מתוך מאגר קטן של דגימות מקור. עבור התקפות Port Scanning השתמשנו בכלי Nmap, ועבור DoS השתמשנו ב- Hping3, DoS Hulk, DoS Goldeneye

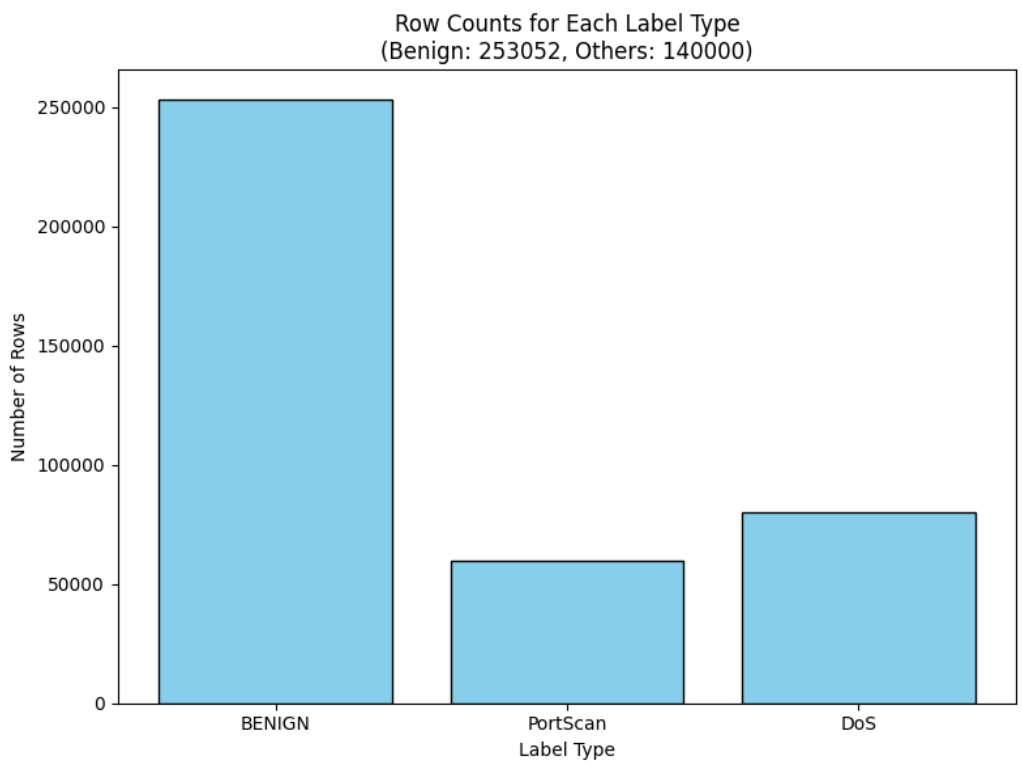
הקוד מייצר וריאציות אקראיות על גבי הנתונים המקוריים במטרה לדמות תרחישי תקיפה מגוונים, תוך שימוש בחישובי קורלציה לינארית בין פיצ'רים שונים על מנת להבטיח שהנתונים החדשים יהיו קרובים ככל האפשר לנתונים אמיתיים.

לאחר איסוף תעבורה תקינה ונגועה מהרשת הלוקאלית והמכללתית, ערכנו Feature Selection על סמך המחקר שלנו, בדקנו והשוונו כל אחד מהפיצ'רים שהוצעו בספרות, הוספנו פיצ'רים ומדדים נוספים, וגם תיקנו את השיטות לאיסוף חלק מהפיצ'רים. ה- Feature Selection שלנו מבוסס על מחקרים קיימים, ומכיל נתונים על תעבורת התקשורת כמו כמות פאקטות בשניה, כמות מידע שהועבר בכל כיוון של התקשורת, כמות דגלים ועוד. אך בנוסף לכך הוספנו מספר פיצ'רים נוספים כמו כמות פורטים אשר ניגשו אליהם באותה תקשורת, כמות דגלי RST ועוד. ובנוסף, חלק מן הפיצ'רים שלא חושבו נכון במחקרים ובמאגרים קיימים, תיקנו והשתמשנו בהם גם כן, אלו פיצ'רים כמו Subflow Fwd Bytes, Average Packet Length ועוד.

עם סיום איסוף הנתונים ובניית המודל, השגנו יכולת זיהוי של 100% של התקפות Port Scanning ו- DoS במודל SVM אחד אשר יודע לזהות התקפת Port Scanning רגילה וגם שקטה וגם התקפות DoS מסוג TCP SYN Flood ו- HTTP GET Flood.
מודל ה- SVM שלנו למד על 26 פיצ'רים שונים ומגוונים.
המודל משתמש ב- Linear Kernel עם C = 1, כאשר הנתונים שעליו אומן המודל, כמו הנתונים שהוא מסווג, עוברים נרמול בעזרת StandardScaler.

בהתבסס על תוצאות המודל, ביצענו ולידציה לאיתור תופעת Overfitting באמצעות שיטת K-Fold Cross Validation עם K = 10. תוצאות הבדיקה הראתה כי המודל מתאפיין בלמידה אופטימלית ואינו מציג סימנים ל-Overfitting.
המודל של Port Scanning ו- DoS אומן על 393,000 flows של פאקטות UDP ו- TCP, כאשר כפי שציינו כל flow הוא תקשורת בין שני מכשירים. מתוכם כ- 253,000 flows אשר מתוייגים כ- Benign, ו- 80,000 flows של התקפת DoS ו- 60,000 flows של Port Scanning.


Linear SVM: Feature Importance (|coef|)
Number of Ports
SYN Flag Count
IAT Std
IAT Mean
Average Packet Length
Bwd Packet Length Min
Bwd Packet Length Mean
Fwd Packet Length Max
Packet Length Max
Fwd Packet Length Mean
Packet Length Std
IAT Max
Subflow Fwd Bytes
Flow Duration
Packets Per Second
Fwd Packet Length Std
Fwd Segment Size Avg
ACK Flag Count
Packet Length Min
Bwd Packet Length Max
Packet Length Variance
Fwd Packet Length Min
Total Length of Fwd Packet
Bwd Packet Length Std
RST Flag Count
Bwd Segment Size Avg
Feature
0
1
2
3
4
5
6
7
Absolute Coefficient Value


Row Counts for Each Label Type
(Benign: 253052, Others: 140000)
250000
200000
150000
100000
50000
0
Number of Rows
BENIGN
PortScan
DoS
Label Type

```
Train Accuracy: 1.00000
Validation Accuracy: 1.00000
Test Accuracy: 1.00000

Confusion Matrix:
[[38004     0     0]
 [    0  9025     0]
 [    0     0 11929]]

Metrics for each class:
Class 0 -> TP: 38004, FP: 0, FN: 0, TN: 20954
Class 1 -> TP: 9025, FP: 0, FN: 0, TN: 49933
Class 2 -> TP: 11929, FP: 0, FN: 0, TN: 47029

Classification Report:
              precision    recall  f1-score   support

           0       1.00      1.00      1.00     38004
           1       1.00      1.00      1.00      9025
           2       1.00      1.00      1.00     11929

    accuracy                           1.00     58958
   macro avg       1.00      1.00      1.00     58958
weighted avg       1.00      1.00      1.00     58958
```

Port Scanning - DoS SVM Model K-Fold Cross Validation :

Fold	Train Accuracy	Validation Accuracy	Precision	Recall	F1-Score	Samples
1	1.0000	1.0000	1.00	1.00	1.00	25,436
2	1.0000	1.0000	1.00	1.00	1.00	25,436
3	1.0000	1.0000	1.00	1.00	1.00	25,436
4	1.0000	1.0000	1.00	1.00	1.00	25,436
5	1.0000	1.0000	1.00	1.00	1.00	25,436
6	1.0000	1.0000	1.00	1.00	1.00	25,436
7	1.0000	1.0000	1.00	1.00	1.00	25,435
8	1.0000	1.0000	1.00	1.00	1.00	25,435
9	1.0000	1.0000	1.00	1.00	1.00	25,435
10	1.0000	1.0000	1.00	1.00	1.00	25,435

זיהוי DNS Tunneling:

בדומה לגישתנו בזיהוי התקפות Port Scanning ו- DoS, גם ב- DNS Tunneling בחנו וחקרנו מאגרי נתונים קיימים וניתחנו את שיטות האיסוף שלהם, את התוכן והיישום שלהם במחקרים מדעיים. אך גם פה עלה כי רוב מאגרי הנתונים הקיימים עבור DNS Tunneling נבנו בעיקר לצורכי פיתוח מודלים של למידה עמוקה, ופחות מתאימים לזיהוי בזמן אמת.

המחקר שערכנו חשף מספר פיצ'רים חשובים שניתן לאסוף מתעבורת DNS, אך אינם קיימים במאגרים הנפוצים, הפיצ'רים הללו כוללים: כמות דגלי DF שנמצאים ב- IPv4 Headers, כמות פאקטות מסוג A, AAAA ו- MX אשר נמצאים ב- DNS Header וגם פיצ'רים הנגזרים מ- Sub Domain Names של DNS כמו Number of Sub Domain Names, Avg Sub Domain Name Length ועוד.

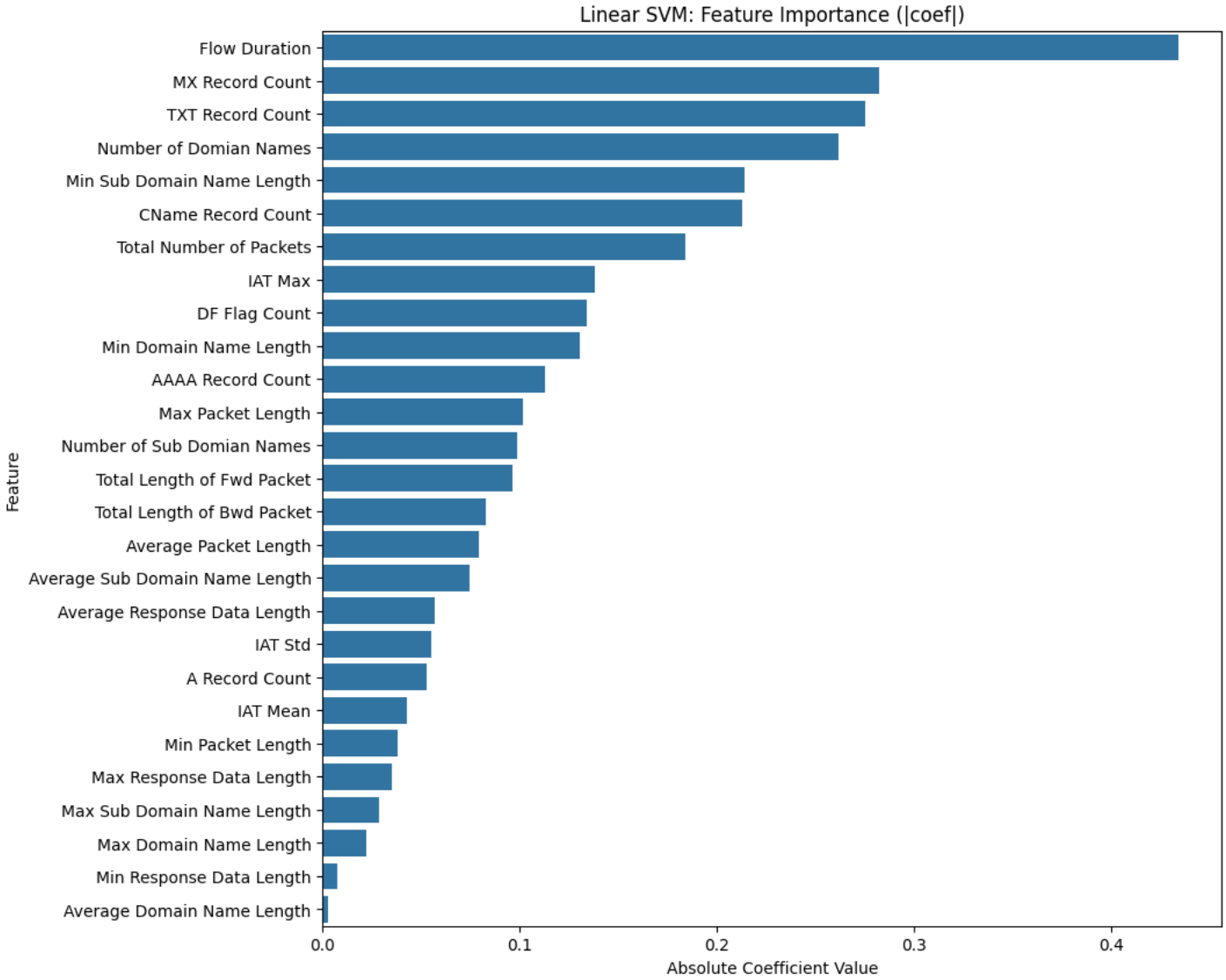
בהתאם לכך, גם עבור התקפה זו בחרנו לבצע איסוף נתונים ידני מתוך הרשת הלוקאלית והמכללתית. תוך כדי תהליך האיסוף ביצענו Feature Selection, וראינו כי חלק מהפיצ'רים הנוספים שבחרנו לאיסוף, תורמים משמעותית ליכולת הזיהוי של המודל.
איסוף תעבורת DNS הוא תהליך ארוך בגלל שאין הרבה פאקטות DNS בתעבורה רגילה. על מנת לייעל את התהליך, פיתחנו כלי שמייצר תעבורת DNS על גבי הרשת הלוקאלית בעזרת שליחה של פאקטות בגדלים שונים, בעלי תוכן משתנה, לשרתי DNS ברחבי העולם.

הכלי שתכננו מייצר תעבורת תקשורת המדמה את המציאות, כאשר תוכן הפאקטות, סוג הדגלים, כמות הדגלים ושרת היעד של הפאקטות נבחרים באופן אקראי ומבוקר. כלי זה המאפשר לנו לייצר תעבורת DNS מגוונת המדמה את המציאות ככל שניתן.
במקביל לאיסוף תעבורת התקשורת הסינתטית, אספנו גם תעבורה תקינה מהרשת, ובמהלך הבדיקות מצאנו כי התעבורה שנוצרת על ידי הכלי שלנו דומה מאוד לתעבורת אמיתית.

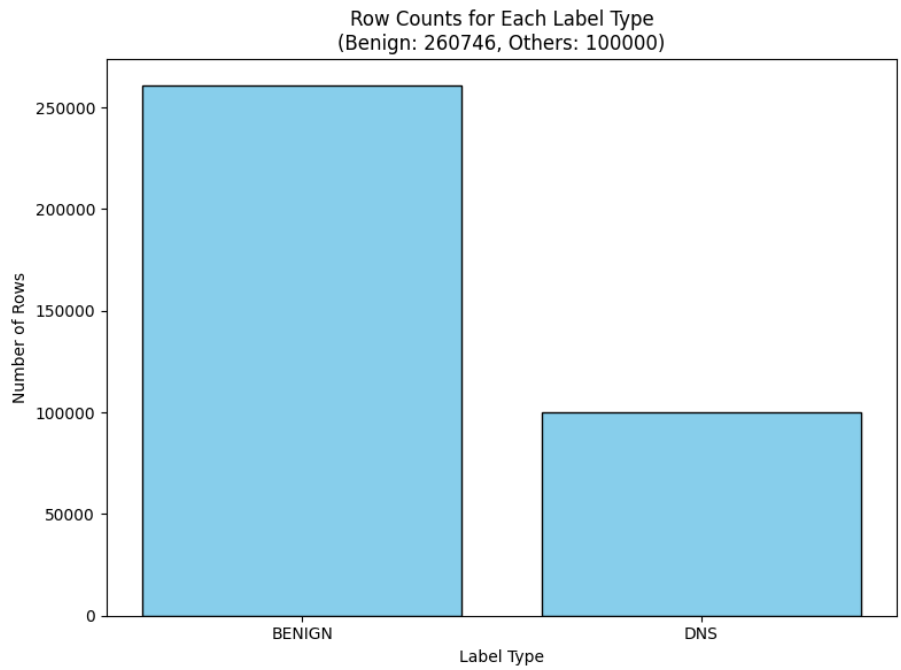
כפי שעשינו עבור התקפות Port Scanning ו- DoS, גם כאן הרצנו התקפות DNS Tunneling בסביבת מעבדה ובנינו קוד שמייצר נתונים חדשים מתוך מאגר קטן של דגימות מקור, תוך חישובי קורלציה לינארית והוספת וריאציות מבוקרות. תעבורת התקשורת הנגועה כללה התקפת DNS Tunneling אקטיבית ופסיבית, תוך שימוש בכלי DnsCat2 המאפשר יצירת ערוץ תקשורת בין מכשיר התוקף לנתקף לצורך העברת פקודות, גניבת מידע ועוד.

עם סיום איסוף הנתונים ובניית המודל, השגנו יכולת זיהוי של 100% של התקפת DNS Tunneling בעזרת מודל SVM אשר יודע לזהות התקפת DNS Tunneling אקטיביות ופסיביות. מודל ה- SVM שלנו למד על 27 פיצ'רים שונים ומגוונים.
המודל משתמש ב- Linear Kernel עם C = 1, כאשר הנתונים שעליו אומן המודל, כמו הנתונים שהוא מסווג, עוברים נרמול בעזרת StandardScaler.

בהתבסס על תוצאות המודל, ביצענו ולידציה לאיתור תופעת Overfitting באמצעות שיטת K-Fold Cross Validation עם K = 10. תוצאות הבדיקה הראתה כי המודל מתאפיין בלמידה אופטימלית ואינו מציג סימנים ל-Overfitting. המודל של DNS Tunneling אומן על 360,000 flows של פאקטות DNS, כאשר כפי שציינו כל flow הוא תקשורת בין שני מכשירים. מתוכם כ- 260,000 flows אשר מתוייגים כ- Benign, ו- 100,000 flows של התקפת DNS Tunneling.



```
Train Accuracy: 1.00000
Validation Accuracy: 1.00000
Test Accuracy: 1.00000

Confusion Matrix:
[[39246     0]
 [    0 14866]]

Metrics for each class:
Class 0 -> TP: 39246, FP: 0, FN: 0, TN: 14866
Class 1 -> TP: 14866, FP: 0, FN: 0, TN: 39246

Classification Report:
              precision    recall  f1-score   support

           0       1.00      1.00      1.00     39246
           1       1.00      1.00      1.00     14866

    accuracy                           1.00     54112
   macro avg       1.00      1.00      1.00     54112
weighted avg       1.00      1.00      1.00     54112
```



DNS SVM Model K-Fold Cross Validation :

Fold	Train Accuracy	Validation Accuracy	Precision	Recall	F1-Score	Samples
1	1.0000	1.0000	1.00	1.00	1.00	25,253
2	1.0000	1.0000	1.00	1.00	1.00	25,253
3	1.0000	1.0000	1.00	1.00	1.00	25,252
4	1.0000	1.0000	1.00	1.00	1.00	25,252
5	1.0000	1.0000	1.00	1.00	1.00	25,252
6	1.0000	1.0000	1.00	1.00	1.00	25,252
7	1.0000	1.0000	1.00	1.00	1.00	25,252
8	1.0000	1.0000	1.00	1.00	1.00	25,252
9	1.0000	1.0000	1.00	1.00	1.00	25,252
10	1.0000	1.0000	1.00	1.00	1.00	25,252

הכלים שבהם השתמשנו עבור ההתקפות:

- **Ettercap** - כלי לניתוח תעבורת הרשת וביצוע התקפות מסוג MITM - Man-in-the-Middle, מאפשר ליירט, לשנות או להזריק תעבורה בין שני צדדים ברשת, ומשמש לעיתים קרובות לבדיקות חדירות או לביצוע התקפות כמו ARP Spoofing.
- **Nmap** - סורק רשת עוצמתי המשמש למיפוי רשתות, גילוי מכשירים פעילים, פתיחת פורטים, וזיהוי שירותים ורמות אבטחה. כלי חיוני לבדיקות חדירות ולזיהוי חולשות במערכות. משמש לביצוע התקפות Port Scanning.
- **Hping3** - כלי לשליחת פאקטות מותאמות בפרוטוקולים שונים כמו TCP, UDP, ICMP. משמש לבדיקות אבטחת רשת, סריקת פורטים, ויצירת התקפות מבוססות תעבורה כמו DoS או flood attacks.
- **DoS Hulk** - סקריפט קוד פתוח שנועד לביצוע התקפת DoS באמצעות שליחת בקשות HTTP מרובות ללא הגבלה, במטרה להציף שרת ולהפסיק את פעולתו התקינה.
- **DoS Goldeneye** - כלי נוסף לביצוע התקפות DoS על שרתי אינטרנט. שולח מספר רב של בקשות HTTP במקביל, עם תמיכה ב-keep-alive, במטרה להעמיס על משאבי השרת.
- **DnsCat2** - כלי שמאפשר תקשורת דרך פרוטוקול DNS, לרוב במטרה לעקוף הגבלות רשת או לביצוע DNS Tunneling. שימושי בהתקפות בהן תוקף רוצה להעביר מידע דרך ערוצים נסתרים.

בכלים אלו נעשה שימוש במהלך המחקר של המתקפות השונות בסביבת מעבדה וגם לצורך איסוף נתונים הכוללים תעבורת תקשורת נגועה בהתקפות לצורך לימוד המודלים.

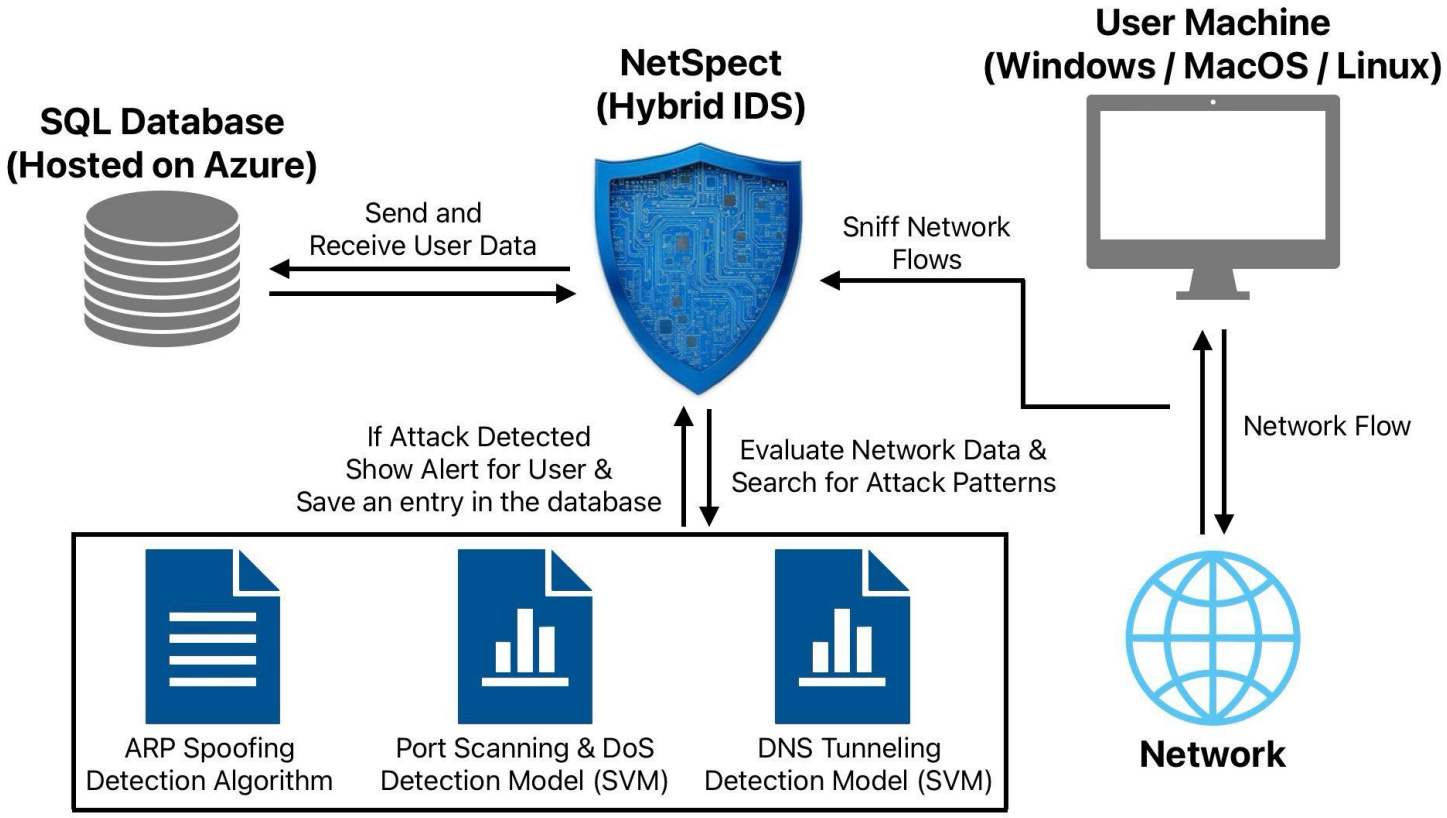
ארכיטקטורת המערכת

התוכנה שלנו מבוסס על ארכיטקטורה מרובת תהליכונים (multithreaded), והיא כוללת ממשק משתמש אינטואיטיבי וקל לשימוש הנרשם בשפת Python בעזרת PySide6. התוכנה בנוייה כך שכל פעולה עיקרית של המערכת מתבצעת על ידי תהליכון נפרד, מה שמאפשר לתוכנה לעבוד בזמן אמת ובצורה יעילה וללא תקיעות, גם במצבי קיצון בהם התוכנה צריכה לסרוק כמות מסיבית של תקשורת תחת מתקפה.

<u>לתוכנה שלנו מספר התליכונים:</u>

- תהליכון **GUI** - תהליכון שאחראי על כל הפעולות של הממשק משתמש. שימוש בתהליכון נפרד לממשק משתמש מאפשר לממשק משתמש לעבוד בצורה חלקה כאשר בקרע מתבצעות פעולות אחרות.
- תהליכון **Sniffer** - תהליכון שאחראי על סריקת תעבורת התקשורת, מיון ואיסוף פאקטות רלוונטיות ושליחת מקבצים של פאקטות לתהליכונים אחרים לצורך זיהוי התקפות.
- תהליכוני זיהוי התקפה - לכל אלגוריתם / מודל ישנו התליכון משלו, אשר התפקיד היחיד שלו הוא לחכות לקבלה של רשימת פאקטות, וזיהוי התקפה בפאקטות שקיבל על ידי שליחת הפאקטות לאלגוריתם או למודל הרלוונטי. במידה והוא מזהה התקפה, הוא מעדכן את תהליכון ה GUI אשר מעדכן את הממשק משתמש בהתאם ומתריע למשתמש.
- תהליכון **SQL** - תהליכון אשר עבודתו העיקרית הוא לתקשר עם שרת ה- SQL שלנו לצורך יבוא ושמירה של נתונים של המשתמשים במהלך השימוש בתוכנה, כאשר התוכנה יודעת לעבוד גם בלי תקשורת עם השרת.
- תהליכון שמירת דוחות - תהליכון שעבודתו היחידה הוא לשמור דוחות בקבצים ברקע על המחשב של המשתמש.
- תהליכון **Logger** - תהליכון שאחראי לשמור בקובץ log את כל הפעולות המתרחשות בתוכנה בכל רגע נתון, גם בפעולות של המשתמש וגם פעולות של תהליכונים אחרים.
- תהליכון **Data Collection** - תהליכון שפועל רק כאשר המשתמש בוחר בפיצ'ר הזה במפורש, התהליכון אחראי לאיסוף את תעבורת התקשורת ולשמור אותה בקובץ CSV חיצוני.

דיאגרמת ארכיטקטורה:



מנגנון סריקת תעבורת התקשורת:

על מנת לזהות את התקפות במהירות וביעילות, גם תחת עומסים בתקשורת, יישמנו אלגוריתם דמוי Round-Robin לצורך מיון וסגמנטציה של ה- flows בזמן שליחתם לתהליכונים המפעילים את מודלי / אלגוריתמי זיהוי התקפות.

תחילה, הפאקטות נאספות לשלושה מאגרים נפרדים, כאשר כל מאגר מיועד לסוג מסוים של פאקטות, כלומר, מאגר אחד לפאקטות ARP, מאגר שני לפאקטות UDP ו-TCP, ומאגר שלישי לפאקטות DNS. לכל סוג פאקטות הוגדר threshold ייחודי, אשר קובע את מספר הפאקטות שיש לאסוף לפני שליחתם לבדיקה על ידי מודל או אלגוריתם לזיהוי התקפות. בנוסף, לכל סוג פאקטות נקבע גם timeout limit, המייצג את משך הזמן המרבי לאיסוף הפאקטות, גם אם טרם הושג הסף. כאשר אחד מהתנאים מתקיים, או

שהסף הושג או שהזמן תם, הפאקטות מועברות לעיבוד לצורך זיהוי פוטנציאלי של התקפות.

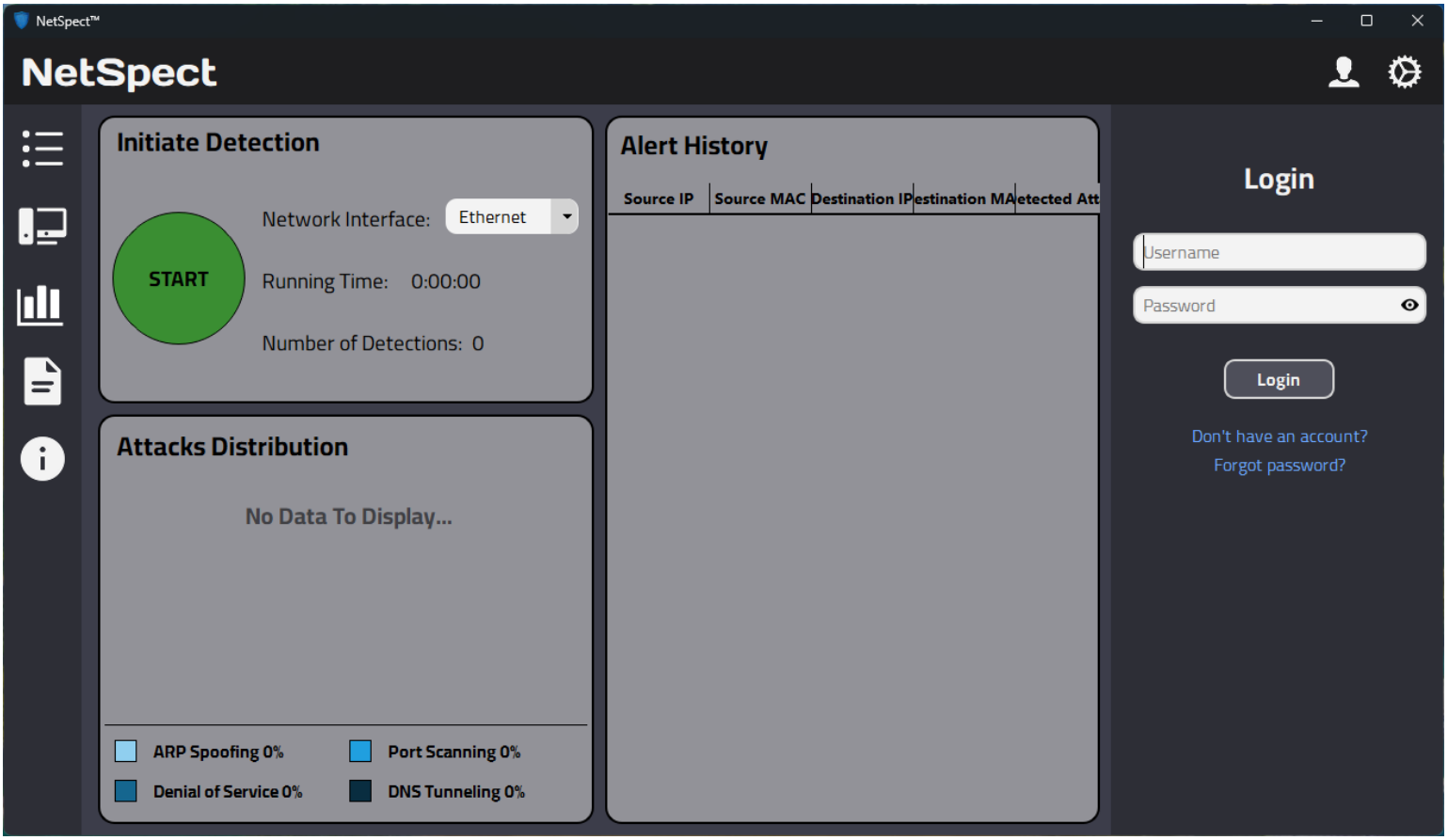
המטרה בהגדרת ערכי ה-threshold וה-timeout limit היא לאפשר זיהוי יעיל בזמן אמת, תוך שמירה על אחידות בתהליך האיסוף לאורך זמן. מנגנון זה מוודא שנאספת כמות מינימלית אך מספקת של פאקטות לצורך ניתוח, ובמקביל מסייע בשמירה על משאבי המערכת שעליה פועלת התוכנה.

לאחר שהפאקטות נאספות לרשימות לפי הסוגים שהוגדרו, התוכנה מפעילה את מנגנון העיבוד ברגע שמתקיים תנאי לשליחת הפאקטות למודלים או לאלגוריתמים. בשלב זה, המערכת סורקת את הרשימות לפי סדר הגעת הפאקטות, ובוחרת באופן מדורג כמות קטנה של פאקטות מכל flow. התהליך נמשך עד להגעה למכסה המקסימלית שהוגדרה מראש. בהתאם לעקרונות של אלגוריתם Round-Robin, אם לאחר סבב אחד עדיין לא נאספה כמות הפאקטות הדרושה, התהליך חוזר על עצמו שוב לפי סדר ההגעה, תוך בחירה נוספת של פאקטות מכל flow, עד להשלמת המכסה.

כתוצאה מהתהליך הזה, אנו מבטיחים שהתוכנה תאסוף ותעביר את הנתונים למודלים או לאלגוריתמים בזמן אמת, תוך שמירה על סדר הגעת הפאקטות. שמירה זו מאפשרת זיהוי מדויק של התקפות ברגע התרחשותן, בהתאם לרצף הנתונים כפי שנקלטו במערכת.

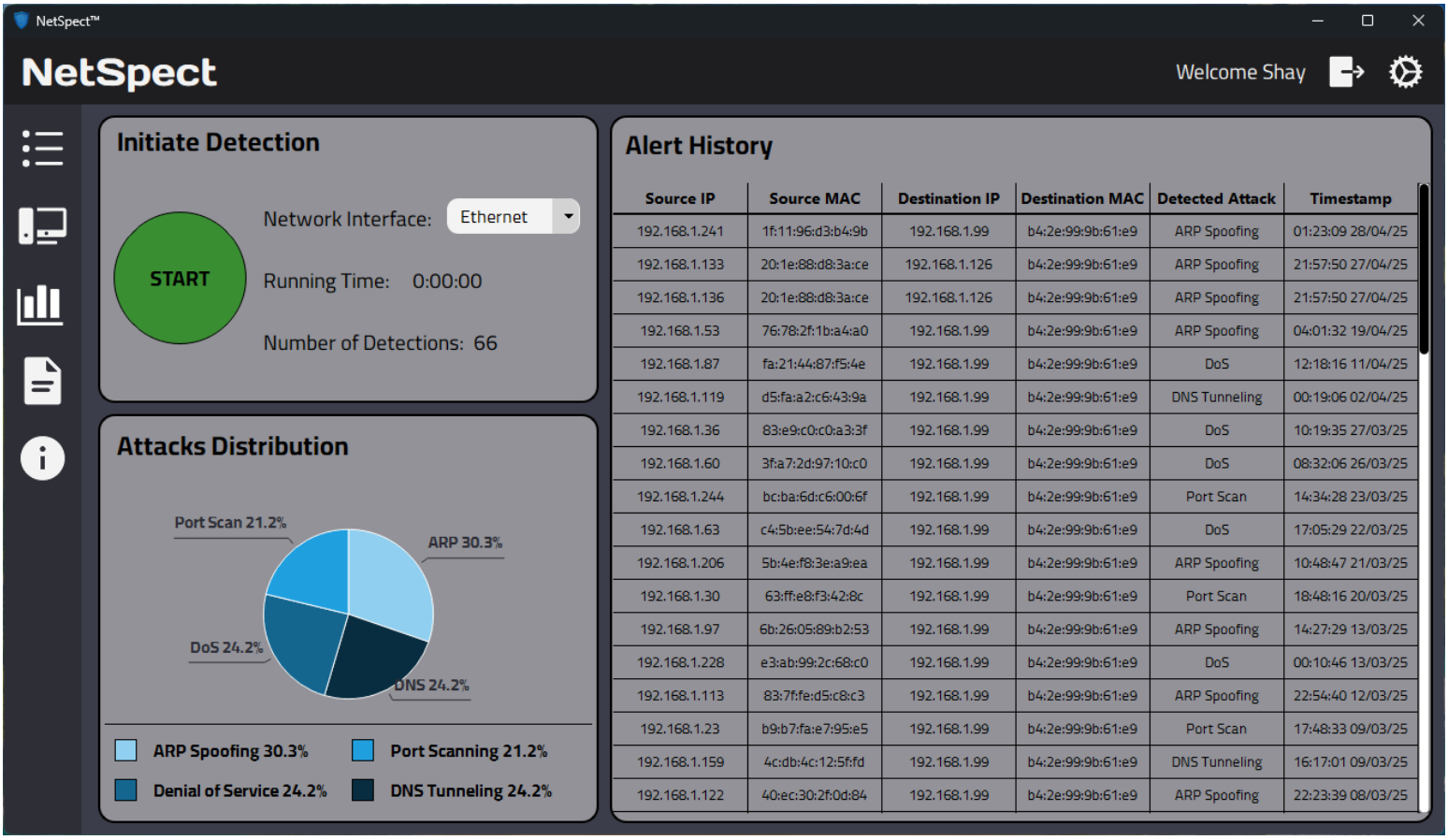
תיאור המערכת

להלן צילומי מסך מהאפליקציה הסופית:



<u>מסך בית - משתמש לא מחובר (משתמש דיפולטיבי) - ערכת נושא למצב כהה:</u>

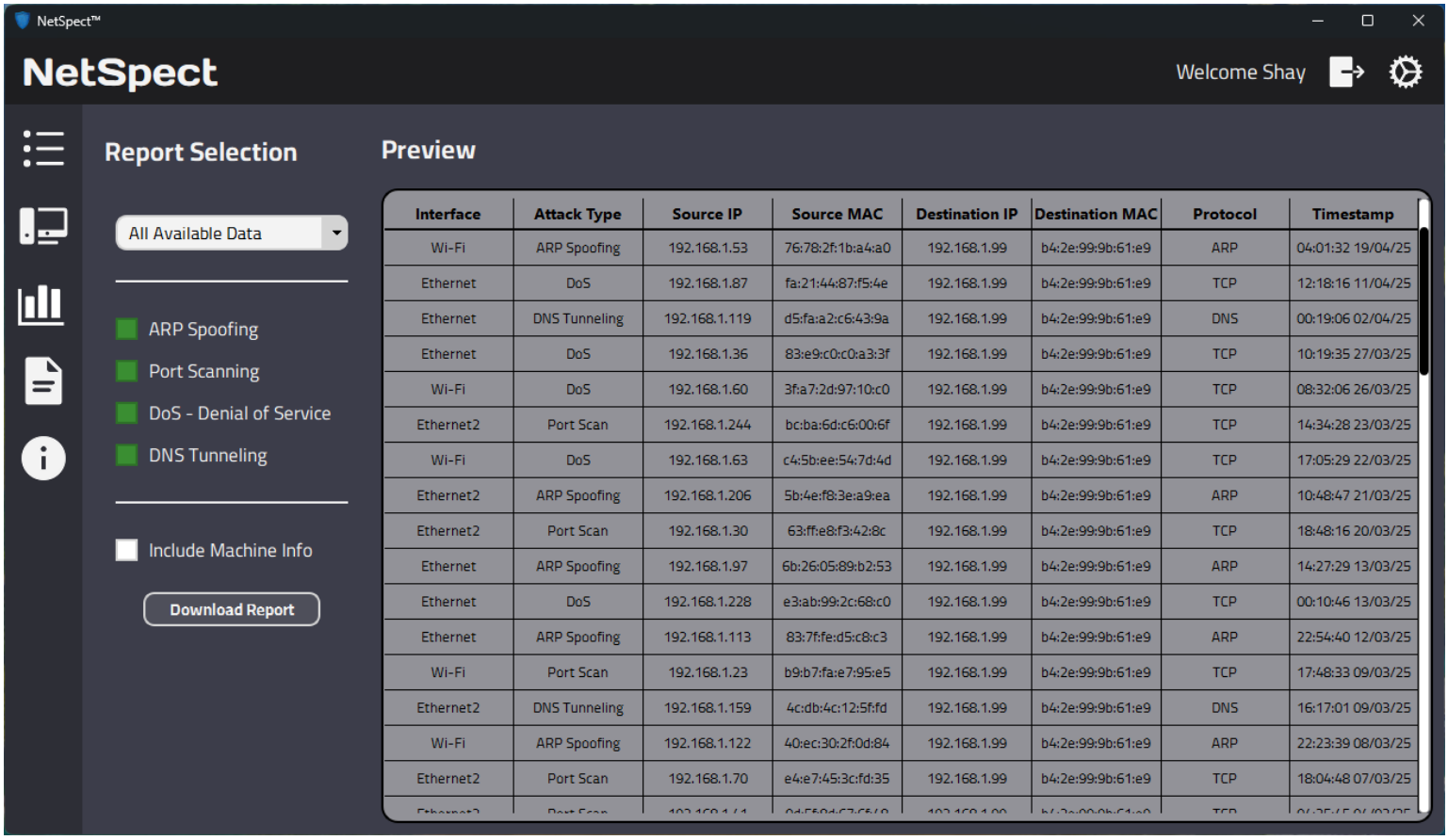
- הפעלה ועצירה של סריקת תעבורת התקשורת לצורך זיהוי התקפות סייבר.
- אפשרות לשינוי ממשק הרשת בהתאם לסוג התקשורת, חיבור קווי או חיבור אלחוטי ועוד.
- צפייה בהיסטוריית התראות עם כמות מידע מינימלית.
- צפייה בהתפלגות המתקפות בדיאגרמת פאי לפי היסטוריית ההתראות.
- הצגת מספר ההתראות שזוהו לאורך הזמנים.
- הצגת משך זמן הסריקה הנוכחית.
- מעבר לכל עמוד אחר בתוכנה.
- מעבר לעמוד הגדרות.
- התחברות למשתמש קיים / פתיחת משתמש קיים / שינוי סיסמה.



Source IP	Source MAC	Destination IP	Destination MAC	Detected Attack	Timestamp
192.168.1.241	1f:11:96:d3:b4:9b	192.168.1.99	b4:2e:99:9b:61:e9	ARP Spoofing	01:23:09 28/04/25
192.168.1.133	20:1e:88:d8:3a:ce	192.168.1.126	b4:2e:99:9b:61:e9	ARP Spoofing	21:57:50 27/04/25
192.168.1.136	20:1e:88:d8:3a:ce	192.168.1.126	b4:2e:99:9b:61:e9	ARP Spoofing	21:57:50 27/04/25
192.168.1.53	76:78:2f:1b:a4:a0	192.168.1.99	b4:2e:99:9b:61:e9	ARP Spoofing	04:01:32 19/04/25
192.168.1.87	fa:21:44:87:f5:4e	192.168.1.99	b4:2e:99:9b:61:e9	DoS	12:18:16 11/04/25
192.168.1.119	d5:fa:a2:c6:43:9a	192.168.1.99	b4:2e:99:9b:61:e9	DNS Tunneling	00:19:06 02/04/25
192.168.1.36	83:e9:c0:c0:a3:3f	192.168.1.99	b4:2e:99:9b:61:e9	DoS	10:19:35 27/03/25
192.168.1.60	3f:a7:2d:97:10:c0	192.168.1.99	b4:2e:99:9b:61:e9	DoS	08:32:06 26/03/25
192.168.1.244	bc:ba:6d:c6:00:6f	192.168.1.99	b4:2e:99:9b:61:e9	Port Scan	14:34:28 23/03/25
192.168.1.63	c4:5b:ee:54:7d:4d	192.168.1.99	b4:2e:99:9b:61:e9	DoS	17:05:29 22/03/25
192.168.1.206	5b:4e:f8:3e:a9:ea	192.168.1.99	b4:2e:99:9b:61:e9	ARP Spoofing	10:48:47 21/03/25
192.168.1.30	63:ff:e8:f3:42:8c	192.168.1.99	b4:2e:99:9b:61:e9	Port Scan	18:48:16 20/03/25
192.168.1.97	6b:26:05:89:b2:53	192.168.1.99	b4:2e:99:9b:61:e9	ARP Spoofing	14:27:29 13/03/25
192.168.1.228	e3:ab:99:2c:68:c0	192.168.1.99	b4:2e:99:9b:61:e9	DoS	00:10:46 13/03/25
192.168.1.113	83:7f:fe:d5:c8:c3	192.168.1.99	b4:2e:99:9b:61:e9	ARP Spoofing	22:54:40 12/03/25
192.168.1.23	b9:b7:fa:e7:95:e5	192.168.1.99	b4:2e:99:9b:61:e9	Port Scan	17:48:33 09/03/25
192.168.1.159	4c:db:4c:12:5f:fd	192.168.1.99	b4:2e:99:9b:61:e9	DNS Tunneling	16:17:01 09/03/25
192.168.1.122	40:ec:30:2f:0d:84	192.168.1.99	b4:2e:99:9b:61:e9	ARP Spoofing	22:23:39 08/03/25

<u>מסך בית - משתמש מחובר - ערכת נושא למצב כהה:</u>

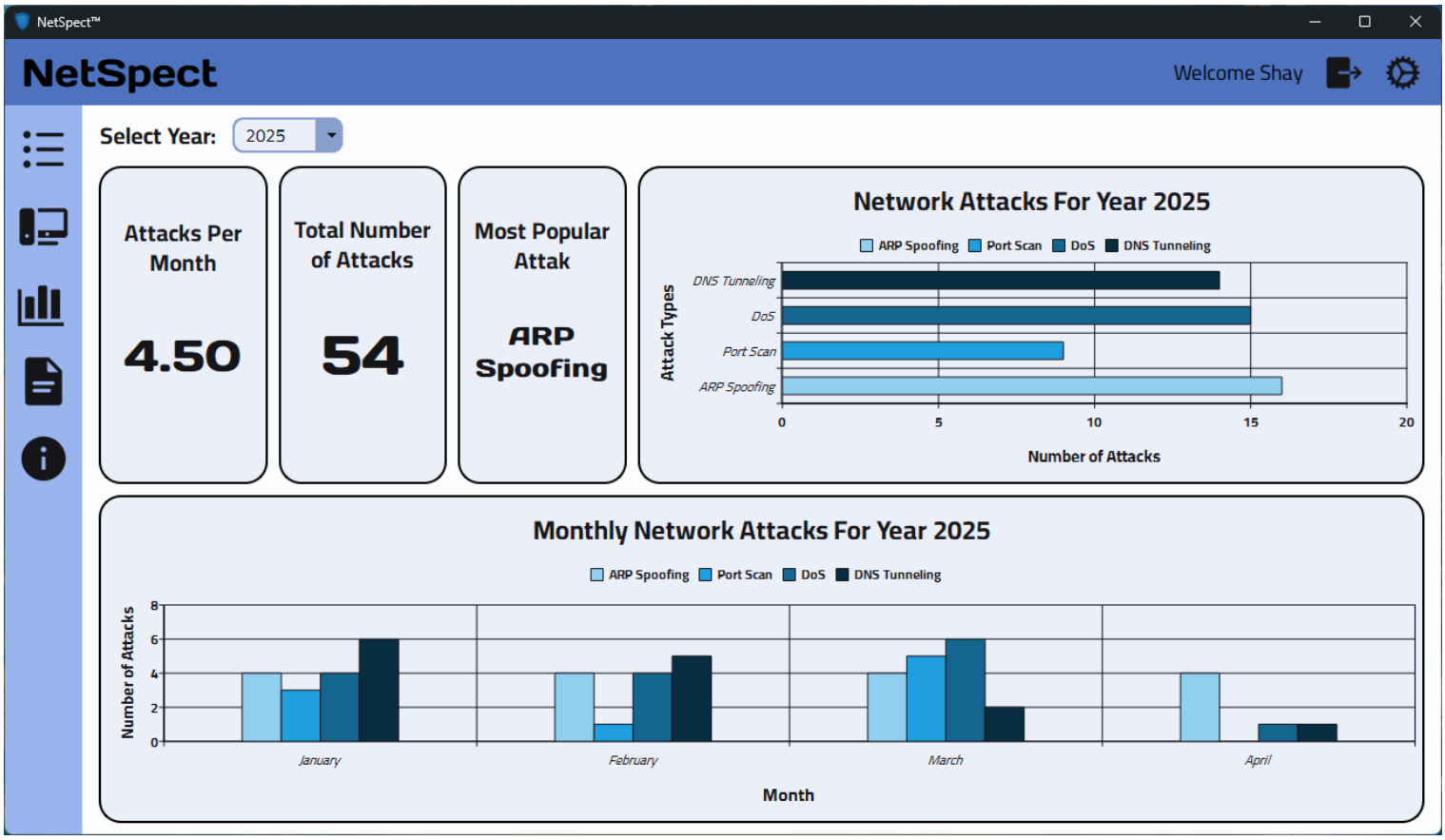
- הפעלה ועצירה של סריקת תעבורת התקשורת לצורך זיהוי התקפות סייבר.
- אפשרות לשינוי ממשק הרשת בהתאם לסוג התקשורת, חיבור קווי או חיבור אלחוטי ועוד.
- צפייה בהיסטוריית התראות עם כמות מידע מינימלית.
- צפייה בהתפלגות המתקפות בדיאגרמת פאי לפי היסטוריית ההתראות.
- הצגת מספר ההתראות שזוהו לאורך הזמנים.
- הצגת משך זמן הסריקה הנוכחית.
- מעבר לכל עמוד אחר בתוכנה.
- מעבר לעמוד הגדרות.
- התנתקות מהמשתמש.



Interface	Attack Type	Source IP	Source MAC	Destination IP	Destination MAC	Protocol	Timestamp
Wi-Fi	ARP Spoofing	192.168.1.53	76:78:2f:1b:a4:a0	192.168.1.99	b4:2e:99:9b:61:e9	ARP	04:01:32 19/04/25
Ethernet	DoS	192.168.1.87	fa:21:44:87:f5:4e	192.168.1.99	b4:2e:99:9b:61:e9	TCP	12:18:16 11/04/25
Ethernet	DNS Tunneling	192.168.1.119	d5:fa:a2:c6:43:9a	192.168.1.99	b4:2e:99:9b:61:e9	DNS	00:19:06 02/04/25
Ethernet	DoS	192.168.1.36	83:e9:c0:c0:a3:3f	192.168.1.99	b4:2e:99:9b:61:e9	TCP	10:19:35 27/03/25
Wi-Fi	DoS	192.168.1.60	3f:a7:2d:97:10:c0	192.168.1.99	b4:2e:99:9b:61:e9	TCP	08:32:06 26/03/25
Ethernet2	Port Scan	192.168.1.244	bc:ba:6d:c6:00:6f	192.168.1.99	b4:2e:99:9b:61:e9	TCP	14:34:28 23/03/25
Wi-Fi	DoS	192.168.1.63	c4:5b:ee:54:7d:4d	192.168.1.99	b4:2e:99:9b:61:e9	TCP	17:05:29 22/03/25
Ethernet2	ARP Spoofing	192.168.1.206	5b:4e:f8:3e:a9:ea	192.168.1.99	b4:2e:99:9b:61:e9	ARP	10:48:47 21/03/25
Ethernet2	Port Scan	192.168.1.30	63:ff:e8:f3:42:8c	192.168.1.99	b4:2e:99:9b:61:e9	TCP	18:48:16 20/03/25
Ethernet	ARP Spoofing	192.168.1.97	6b:26:05:89:b2:53	192.168.1.99	b4:2e:99:9b:61:e9	ARP	14:27:29 13/03/25
Ethernet	DoS	192.168.1.228	e3:ab:99:2c:68:c0	192.168.1.99	b4:2e:99:9b:61:e9	TCP	00:10:46 13/03/25
Ethernet	ARP Spoofing	192.168.1.113	83:7f:fe:d5:c8:c3	192.168.1.99	b4:2e:99:9b:61:e9	ARP	22:54:40 12/03/25
Wi-Fi	Port Scan	192.168.1.23	b9:b7:fa:e7:95:e5	192.168.1.99	b4:2e:99:9b:61:e9	TCP	17:48:33 09/03/25
Ethernet2	DNS Tunneling	192.168.1.159	4c:db:4c:12:5f:fd	192.168.1.99	b4:2e:99:9b:61:e9	DNS	16:17:01 09/03/25
Wi-Fi	ARP Spoofing	192.168.1.122	40:ec:30:2f:0d:84	192.168.1.99	b4:2e:99:9b:61:e9	ARP	22:23:39 08/03/25
Ethernet2	Port Scan	192.168.1.70	e4:e7:45:3c:fd:35	192.168.1.99	b4:2e:99:9b:61:e9	TCP	18:04:48 07/03/25

<u>מסך דוחות - משתמש מחובר - ערכת נושא למצב כהה:</u>

- צפייה בהיסטוריית ההתראות עם כל המידע הרלוונטי.
- אופציה למיון וסינון היסטוריית ההתראות לפי סוג התקפה.
- אופציה למיון וסינון היסטוריית ההתראות לפי תקופת זמן.
- אופציה לשמור מידע על המערכת באת שמירת הדוח אודות ההתראות.
- שמירת דוח של ההתראות בהתאם לקונפיגורציית המיון והסינון שנבחרו.
- מעבר לכל עמוד אחר בתוכנה.
- מעבר לעמוד הגדרות.
- התנתקות מהמשתמש.

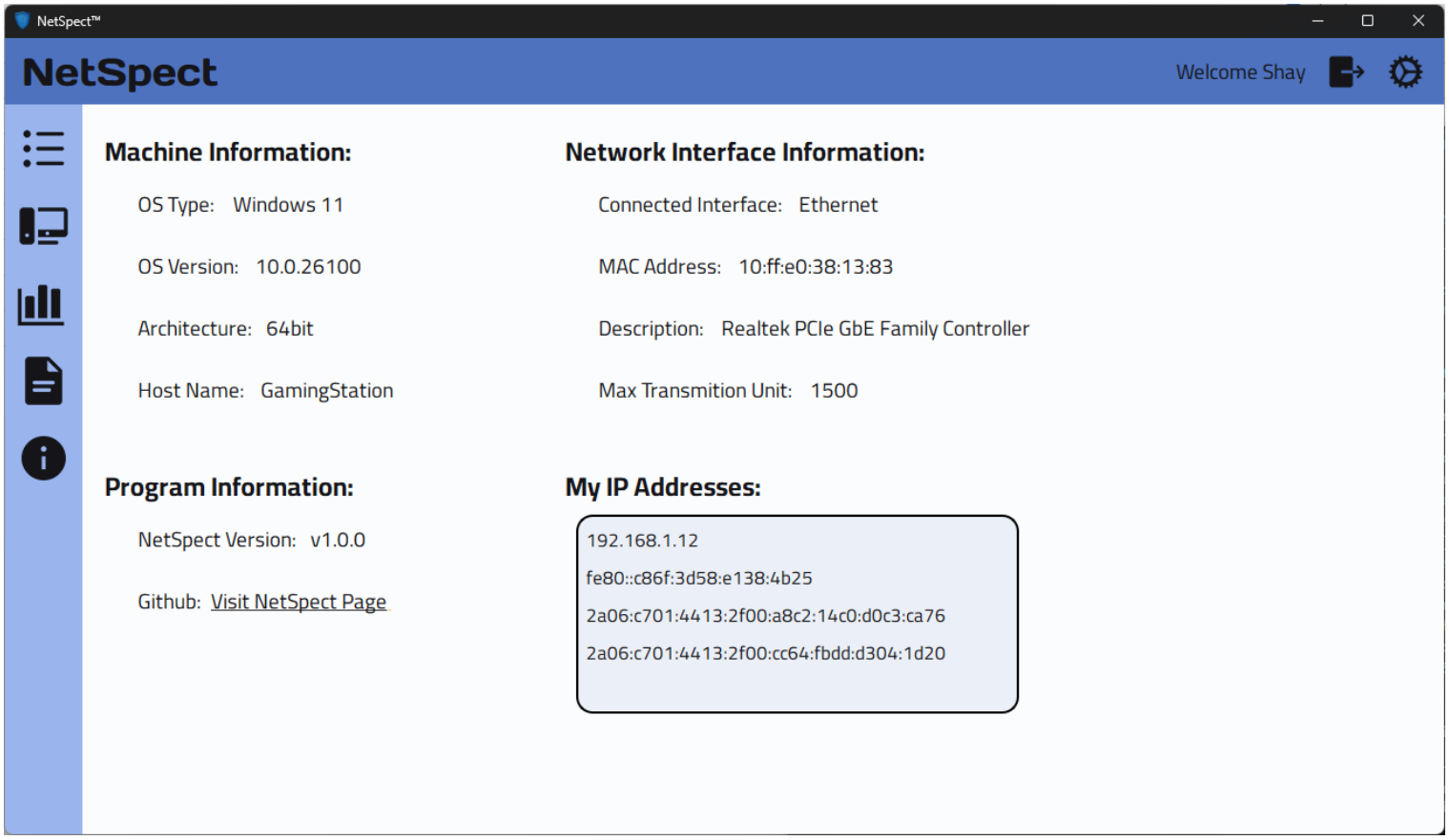


<u>מסך אנליטיקה - משתמש מחובר - ערכת נושא למצב בהיר:</u>

- צפייה בגרף המציג את מספר ההתראות לכל סוג התקפה לאורך השנה שנבחרה.
- צפייה בכמות ההתקפות הממוצעת בכל חודש בשנה שנבחרה.
- צפייה במספר הכולל של ההתקפות שזוהו בשנה שנבחרה.
- צפייה השם ההתקפה הכי פופולארית בשנה שנבחרה.
- צפייה בגרף המציג את מספר ההתראות לכל סוג התקפה בכל חודשי השנה שנבחרה.
- אופציה לשנות את השנה שעבורה מוצגים הנתונים בעמוד.
- מעבר לכל עמוד אחר בתוכנה.
- מעבר לעמוד הגדרות.
- התנתקות מהמשתמש.

<u>מסך הגדרות - משתמש מחובר - ערכת נושא למצב בהיר:</u>

- אופציה לשינוי כתובת מייל המשויכת למשתמש הנוכחי.
- אופציה לשינוי שם המשתמש של המשתמש הנוכחי.
- אופציה לשינוי הסיסמה של המשתמש הנוכחי.
- אופציה לשינוי ערכת הנושא, ניתן לבחור בין מצב בהיר או מצב כהה.
- אופציה לשנות את מנגנון הפעולה של התוכנה ולעבור ממצב זיהוי התקפות למצב של איסוף נתונים מתעבורת רשת מסוג *UDP & TCP* או *DNS*.
- אופציה להוסיף, למחוק ולהעתיק כתובת *MAC* לרשימה שחורה של כתובות, דבר המאפשר לחסום זיהוי התקפות מאותן כתובות *MAC*.
- מעבר לכל עמוד אחר בתוכנה.
- מעבר לעמוד הגדרות.
- התנתקות מהמשתמש.



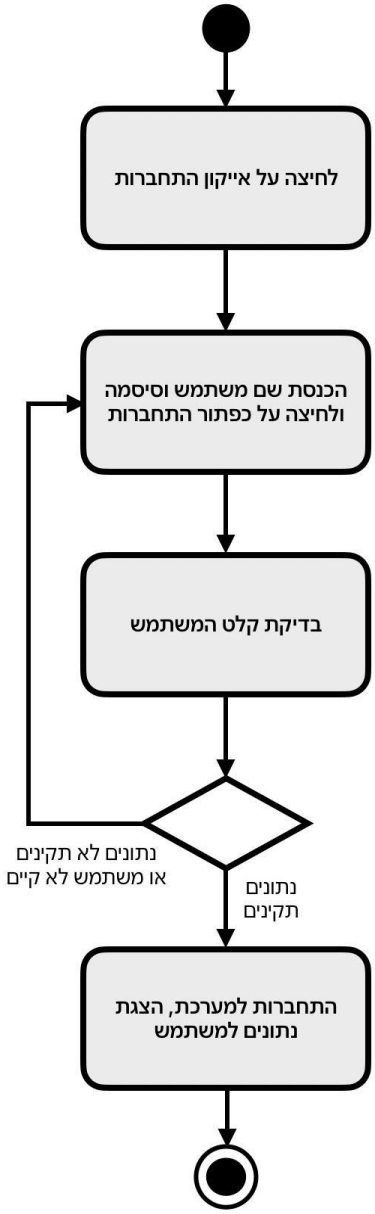
מסך אינפורמציה - משתמש מחובר - ערכת נושא למצב בהיר:

- צפייה באינפורמציה אודות המחשב הנוכחי אשר מריץ את התוכנה.
- צפייה במידע אודות ממשק הרשת שנבחר במסך הבית לצורך סריקת תעבורת הרשת.
- מידע אודות מספר הגרסה של התוכנה וקישור לעמוד ה- *GitHub* שלנו.
- מעבר לכל עמוד אחר בתוכנה.
- מעבר לעמוד הגדרות.
- התנתקות מהמשתמש.

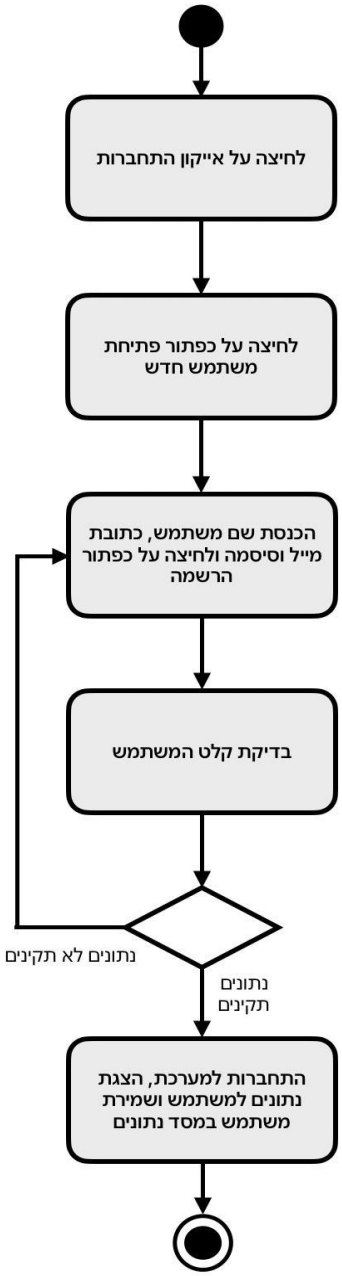
דיאגרמת פעילות

להלן מספר דיאגרמות פעילות (Use Case Diagram) על הפעולות המרכזיות והחשובות ביותר במערכת שלנו, פעולות אלו מתחילות בפעולה של המשתמש ומסתיימות בהצגת תוצאות או ביצוע פעולות בתוכנה שלנו:

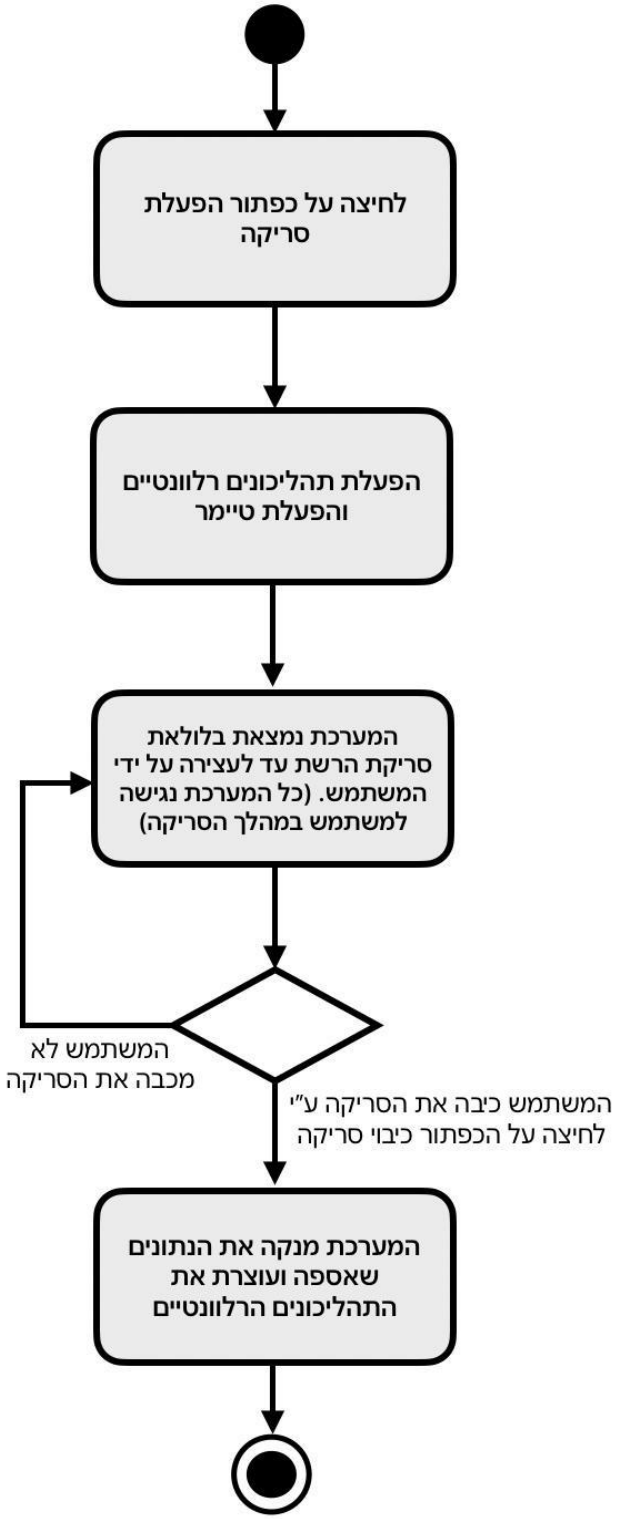
<u>כניסה למשתמש קיים:</u>



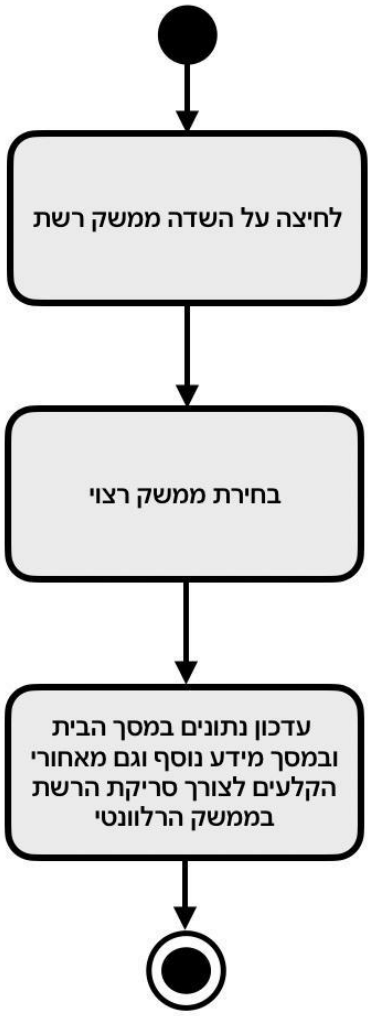
<u>פתיחת משתמש חדש:</u>



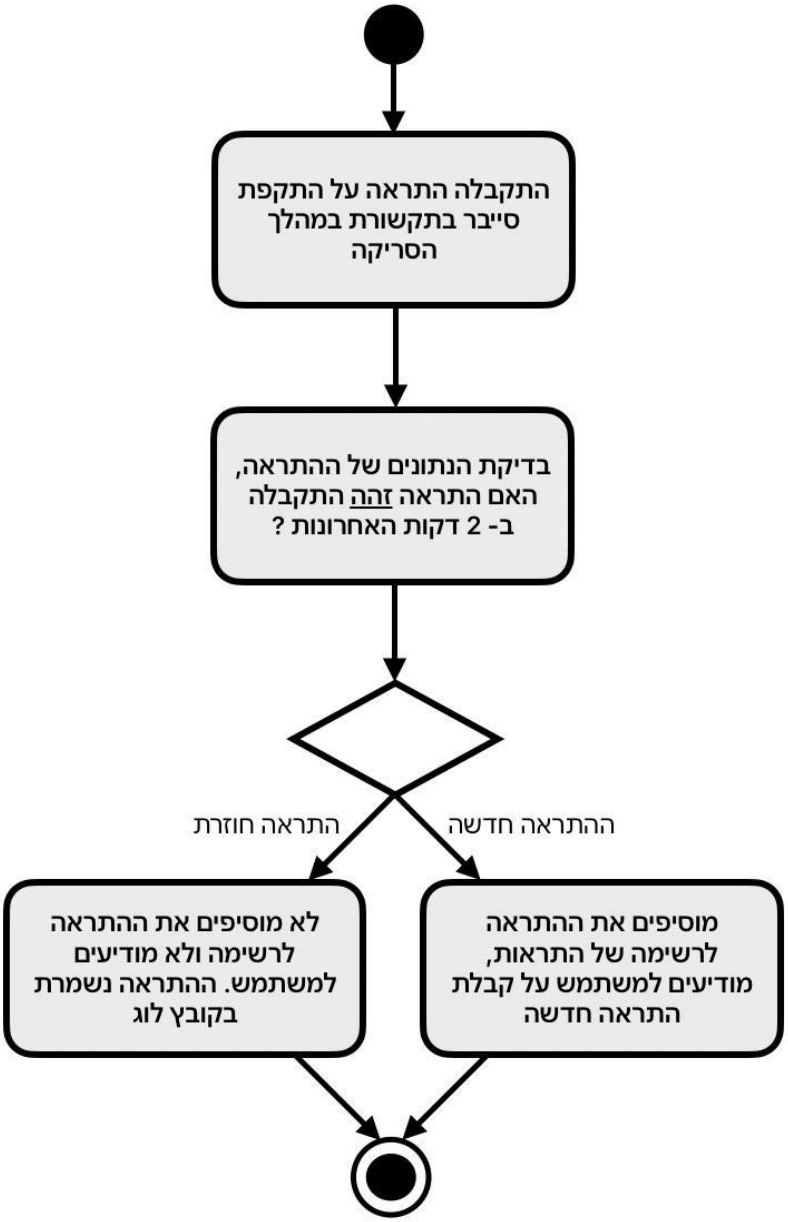
הפעלה וכיבוי מנגנון סריקת הרשת:



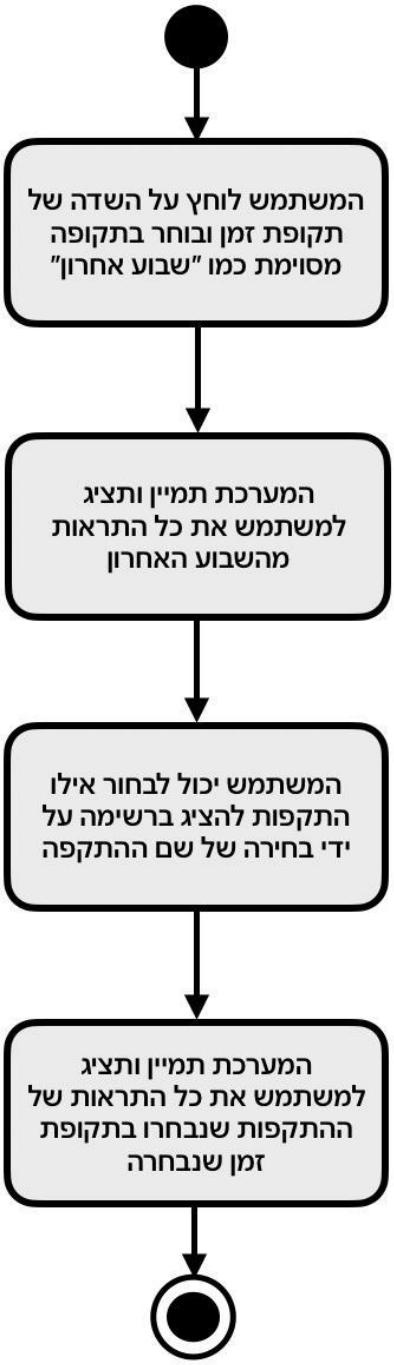
שינוי הממשק הרשת:



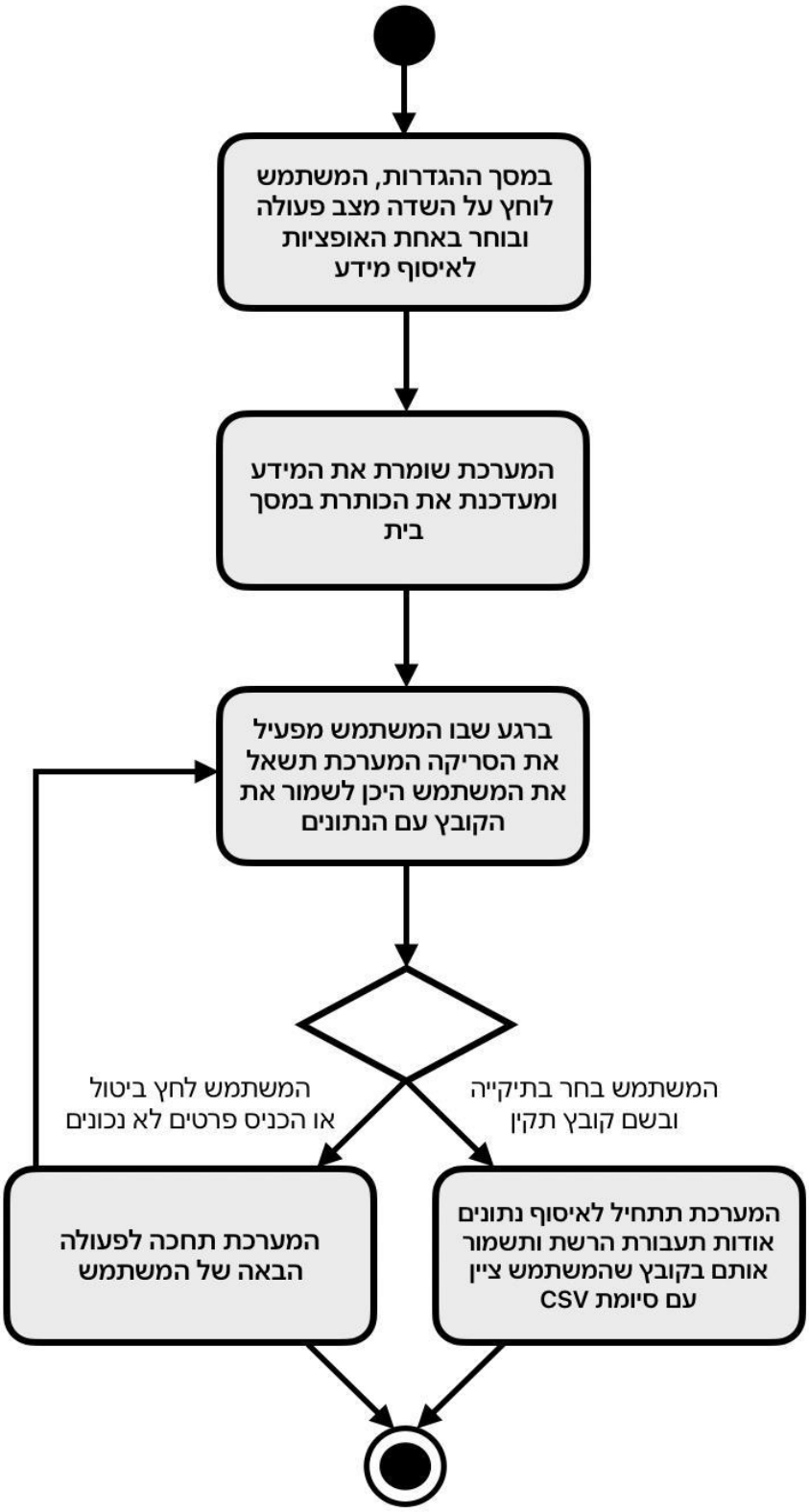
קבלת התראה על זיהוי התקפה:



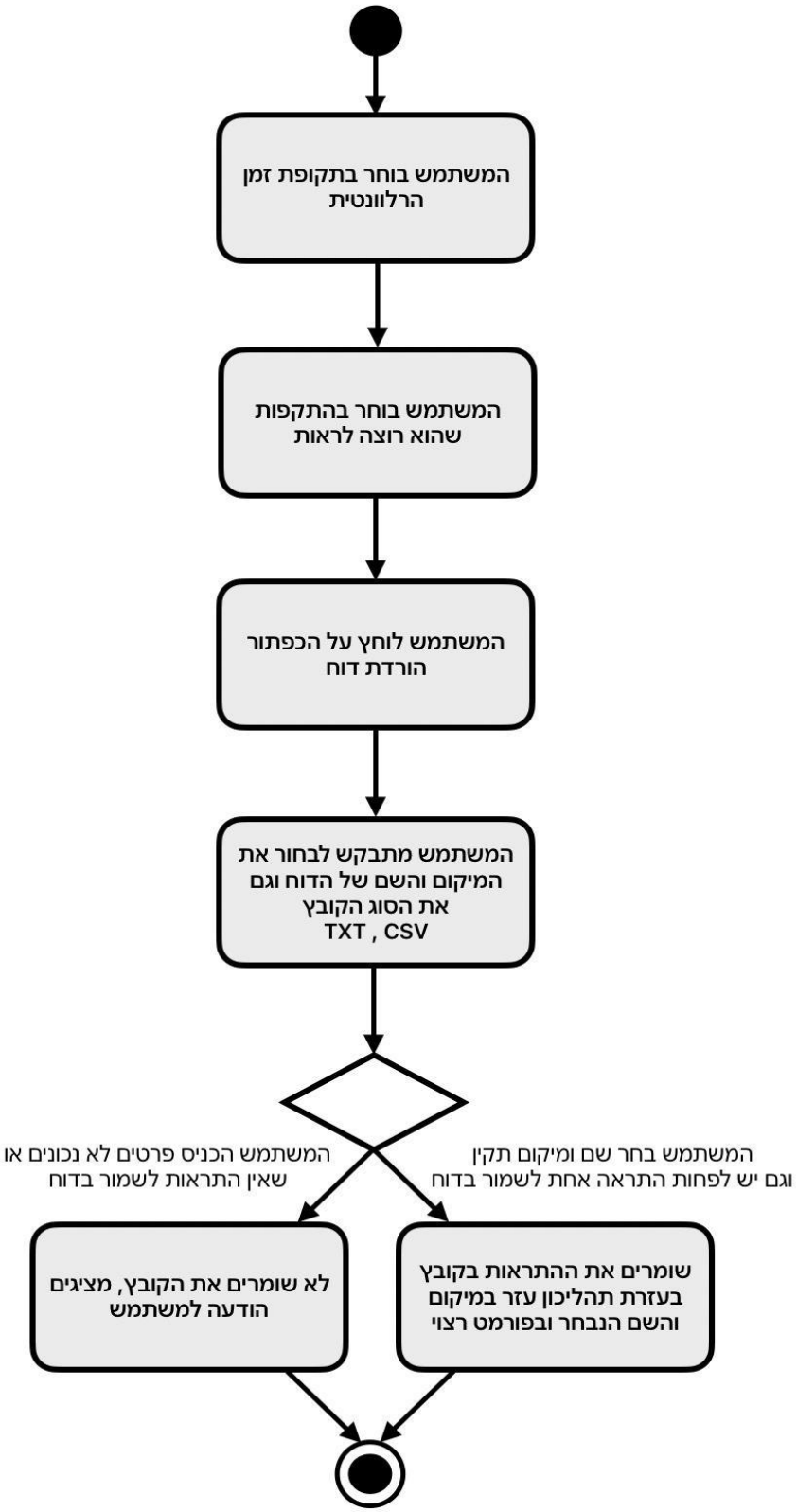
מיון וסיווג היסטוריית ההתראות:



שינוי בין זיהוי בזמן אמת לאיסוף מידע:



הורדת דוח אודות היסטוריית ההתראות:



בדיקות

בפרויקט שלנו ישנם 3 סוגי בדיקות שונות:

- ❖ בדיקות Unit Test - בדיקות פשוטות לחלקים קטנים וספציפיים בקוד.
- ❖ בדיקות UI - בדיקות ממשק משתמש, כולל גם בדיקות פשוטות וגם מורכבות.
- ❖ בדיקות End To End - בדיקות מורכבות שבודקות את המערכת מקצה לקצה, כוללות בדיקת ממשק משתמש וגם יכול זיהוי התקפות ושמירת נתונים לשרת.

בדיקות Unit Test:

בפרויקט שלנו רשמנו 27 בדיקות Unit Test, עם רמת כיסוי בינונית מכיוון שכמחצית מהקוד שלנו מורכב וכולל הרבה רכיבים שתלויים אחד בשני ומונעים מאיתנו לבנות עבורם בדיקות Unit Test נפרדות.

```
tests/UnitTest.py::testGuidToStrPositive PASSED                                                    [ 77%]
tests/UnitTest.py::testGuidToStrNegative[07444BF8-F269-473F-B278-891AA8D81C6E] PASSED              [ 81%]
tests/UnitTest.py::testGuidToStrNegative[{473F-B278-891AA8D81C6E}] PASSED                          [ 85%]
tests/UnitTest.py::testGuidToStrNegative[not a guid] PASSED                                        [ 88%]
tests/UnitTest.py::testGetSystemInformation PASSED                                                 [ 92%]
tests/UnitTest.py::testSaveFlowsInFile PASSED                                                      [ 96%]
tests/UnitTest.py::testSaveCollectedData PASSED                                                    [100%]

================================================ 27 passed in 0.15s ================================================
```

בדיקות UI:

בפרויקט שלנו רשמנו 33 בדיקות UI, עם רמת כיסוי של 80% מתוך הפונקציונליות של המערכת. בדיקות אלו כוללים בדיקות התחברות והתנתקות, שינוי פרטי משתמש, בדיקות קלטים לפי Regex, בדיקות הרשמה למערכת, בדיקות סריקת הרשת התקשורת ועוד. לכל אחד מהבדיקות ישנם תרחישים חיוביים ושליליים.

```
PASSED                                                    [ 93%]
tests/GUITest.py::testChangePasswordNegative[passwords1] 01:20:41 12/06/25 [INFO] Main_Thread: Initialized PortScanDo
S and DNSTunneling models and scalers successfully.
01:20:44 12/06/25 [INFO] SQL_Thread: Connected to database successfully.
PASSED                                                    [ 96%]
tests/GUITest.py::testChangePasswordNegative[passwords2] 01:20:44 12/06/25 [INFO] Main_Thread: Initialized PortScanDo
S and DNSTunneling models and scalers successfully.
01:20:46 12/06/25 [INFO] SQL_Thread: Connected to database successfully.
PASSED                                                    [100%]

========================================== 33 passed in 82.71s (0:01:22) ==========================================
```

בדיקות End To End:

הסוג האחרות של הבדיקות שעשינו הם בדיקות מקצה לקצה, בדיקות אלו נעשו באופן ידני לפי תסריטי בדיקה מוגדרים מראש. מטרתן העיקרית של בדיקות אלה היא לוודא את יכולת הזיהוי של המודלים והאלגוריתמים, לבדוק את תקינות תהליך שמירת הנתונים ושליפתם ממסד הנתונים, ולוודא כי ממשק המשתמש פועל כמצופה במצבים השונים שנבדקו.

תסריטי הבדיקה:

שם הבדיקה	צעדים	תוצאה צפוייה
זיהוי התקפת **ARP Spoofing**	התחברות למשתמש קיים	ההתחברות הצליחה
	התחלת סריקת התקשורת במכשיר A	הסריקה התחילה, הכפתור אדום, ה- timer עולה כל שניה
	הרצת התקפת ARP Spoofing לזיוף ה- router ברשת הלוקאלית ממכשיר B בעזרת Ettercap	ההתקפה רצה בהצלחה
	צפייה בהיסטוריית ההתראות במכשיר A - זיהוי ההתקפה התקבל ונרשם	ההתקפה זוהתה בהצלחה במכשיר A, כתובת ה- IP ו- MAC תואמים לשני המכשירים. ההתקפה נרשמה בהצלחה במסד הנתונים עבור המשתמש הנוכחי.
זיהוי התקפת **Port Scanning**	התחברות למשתמש קיים	ההתחברות הצליחה
	התחלת סריקת התקשורת במכשיר A	הסריקה התחילה, הכפתור אדום, ה- timer עולה כל שניה
	הרצת התקפת Port Scanning לסריקת 10,000 פורטים פתוחים על מכשיר A ממכשיר B בעזרת הכלי Nmap עם סריקה שקטה	ההתקפה רצה בהצלחה והתחילה להתבצע
	צפייה בהיסטוריית ההתראות במכשיר A - זיהוי ההתקפה התקבל ונרשם	ההתקפה זוהתה בהצלחה במכשיר A, כתובת ה- IP ו- MAC תואמים לשני המכשירים. ההתקפה נרשמה בהצלחה במסד הנתונים עבור המשתמש הנוכחי.
זיהוי התקפת **DoS TCP SYN Flood**	התחברות למשתמש קיים	ההתחברות הצליחה
	התחלת סריקת התקשורת במכשיר A	הסריקה התחילה, הכפתור אדום, ה- timer עולה כל שניה

	הרצת התקפת DoS עם כלי 3Hping ממכשיר B אשר תתקוף את מכשיר A	ההתקפה רצה בהצלחה והתחילה להתבצע
	צפייה בהיסטוריית ההתראות במכשיר A - זיהוי ההתקפה התקבל ונרשם	ההתקפה זוהתה בהצלחה במכשיר A, כתובת ה- IP ו- MAC תואמים לשני המכשירים. ההתקפה נרשמה בהצלחה במסד הנתונים עבור המשתמש הנוכחי.
זיהוי התקפת **DoS HTTP GET Flood**	התחברות למשתמש קיים	ההתחברות הצליחה
	הפעלת שרת Flask על מכשיר A בפורט 8090	השרת רץ ב localhost:8090 וניתן להשתמש בו
	התחלת סריקת התקשורת במכשיר A	הסריקה התחילה, הכפתור אדום, ה- timer עולה כל שניה
	הרצת התקפת DoS ממכשיר B על שרת ה- Flask בפורט 8090 אשר רץ במכשיר A	ההתקפה רצה בהצלחה והתחילה להתבצע
	צפייה בהיסטוריית ההתראות במכשיר A - זיהוי ההתקפה התקבל ונרשם	ההתקפה זוהתה בהצלחה במכשיר A, כתובת ה- IP ו- MAC תואמים לשני המכשירים. ההתקפה נרשמה בהצלחה במסד הנתונים עבור המשתמש הנוכחי.
זיהוי התקפת **DNS Tunneling**	התחברות למשתמש קיים	ההתחברות הצליחה
	התחלת סריקת התקשורת במכשיר A	הסריקה התחילה, הכפתור אדום, ה- timer עולה כל שניה
	להפעיל את שרת ההתקפה במכשיר B בעזרת 2dnscat	השרת עלה ומחכה לחיבור של מכשיר קורבן
	הרצת קובץ זדוני ממכשיר A (חייב להיות Windows) כדי להתחבר לשרת ההתקפה במכשיר B	הקובץ רץ בהצלחה, ההתחברות הצליחה
	הרצת התקפת DNS Tunneling ממכשיר B, תחילה יוצרים ערוץ תקשורת shell עם הקורבן ולאחר מכן מזריקים פקודות כמו ipconfig	ההתקפה רצה בהצלחה, הפקודות רצות ברקע והתוצאות שלהם מופיעות במכשיר B
	צפייה בהיסטוריית ההתראות במכשיר A - זיהוי ההתקפה התקבל ונרשם	ההתקפה זוהתה בהצלחה במכשיר A, כתובת ה- IP ו- MAC תואמים לשני המכשירים. ההתקפה נרשמה בהצלחה במסד הנתונים עבור המשתמש הנוכחי.

סיכום

לסיכום, הפרויקט שלנו סיפק בהצלחה מערכת לזיהוי חדירות בזמן אמת (HIDS) המשלב אלגוריתמים לזיהוי חתימות וגם מודלים לזיהוי אנומליות, מה שמקנה לו את המסוגלות לזהות ארבעה סוגים קריטיים של מתקפות סייבר בתקשורת: Port Scanning, DoS, ARP Spoofing, and DNS Tunneling כאשר עבור כל סוג מתקפה קיימת יכולת זיהוי של מספר תתי-סוגים. המערכת מבוססת על פתרון מותאם ואופטימלי הפועל בתצורת ריבוי תהליכים (multi-threaded) ומשלבת באופן חכם בין אלגוריתמים לזיהוי התקפות לבין מודלים של למידת מכונה. בנוסף, התמודדות עם החסרונות של מאגרי נתונים קיימים באמצעות איסוף ידני של נתוני רשת אפשרה השגת רמת דיוק גבוהה במיוחד, תוך הפחתה משמעותית בכמות ההתראות השגויות (false positives). התוצאה היא פתרון אמין, יעיל וידידותי למשתמש, אשר מספק מענה מתקדם לאתגרי הסייבר והאבטחה של רשתות מודרניות.

קישור ל- GitHub Repository של הפרויקט
קישור לסרטון Demo של הפרויקט

www.sce.ac.il

קמפוס באר שבע

ביאליק 56, באר שבע 84100

קמפוס אשדוד

ז'בוטינסקי 84, אשדוד 77245