כישלון שיטות הגנת הסייבר הקלאסיות – מה הלאה?

אמיר אורבוך, גבי סיבוני

מבוא

שיטות ההגנה הקלאסיות הנהוגות בעולם בעשרות השנים האחרונות אינן מצליחות לעצור התקפות פוגעניות (malware) מודרניות העושות שימוש בפרצות אבטחה שאינן מוכרות (ולכן אין להן עדיין תיקון), שנקראות חולשות יום־אפס (zero-day vulnerabilities). דוגמאות להתקפות אלה על מחשבים ועל רשתות תקשורת של ארגונים עסקיים ושל ספקי תשתיות ושירותים חיוניים וקריטיים הן וירוסים, תולעים, דלת אחורית, סוסים טרויאניים – כלי ניהול/גישה מרחוק (RATs). שיטות ההגנה הקלאסיות, הכוללות אמצעי תוכנה וחומרה והמתבססות על חומות אש (FireWall), חתימות וחוקים (rules), תוכנות אנטי־וירוס, סינון תוכן, מערכות איתור חדירה (IDS) ודומיהם נכשלות לחלוטין בהגנה מפני איומים לא־מוכרים, דוגמת איומים המבוססים על חולשות יום־אפס ואיומים חדשים. איומים מתוחכמים וחמקניים אלה מתחזים להיות מידע ונתונים אמינים וחוקיים במערכת, ולכן מערכות ההגנה הקלאסיות אינן מספקות את המענה ההגנתי הדרוש. מערכות ההגנה המקובלות כיום מגֵנות מפני התקפות מוכרות על סמך חתימות ידועות וניתוח לאחור של התקפות, על מנת לייצר באופן היוריסטי,[1] אבל הן חסרות תועלת מול ההתקפות המתרבות והולכות שאינן מוכרות, וחסרות כל חתימה. לפתרון בעיה זו דרושים חשיבה ופתרונות אחרים. מאמר זה מציע גישת הגנה עדכנית, שבבסיסה ניתוח מידע רגיש שעליו יש להגן, למטרת זיהוי התנהגויות אנומליות.[2] המידע המנותח כולל את פעילות התקשורת הארגונית (datasilos) כמקור להבנת התנהגות לא־רגילה (אנומליות), המעידה ברוב המקרים על קיום פוגענים במערכת.

פרופ׳ אמיר אורבוך הנו חבר סגל בית הספר למדעי המחשב באוניברסיטת תל־אביב וחוקר במסגרת תוכנית ניובאוור ללוחמת סייבר במכון למחקרי בטחון לאומי.
ד״ר גבי סיבוני הוא ראש תוכנית צבא ואסטרטגיה וראש תוכנית לוחמת סייבר במכון למחקרי בטחון לאומי.
מאמר זה נכתב בסיועו של אביב רוטברט, תלמיד לתואר שלישי, מלגאי בתוכנית ניובאוור במכון למחקרי בטחון לאומי.

המאמר מציע להתבסס על הנתונים שעליהם יש להגן כמקור ידע לפיתוח מערכת ההגנה. ניתוח אנליטי של נתונים מסיביים (BIG-DATA Analytics) יאפשר זיהוי פוגענים כאלה תוך בניית מודל המאפשר אמינות גבוהה של זיהוי ומזעור התרעות השווא (false positive), המהוות אתגר לכל מערכת הגנה.

התפתחות האיומים והמגבלות של שיטות ההגנה המסורתיות

התקפות הסייבר הראשונות על מחשבים התבססו על וירוסים או תולעים המשכפלים עצמם ומתפשטים במהירות. אולם טכנולוגיית האנטי־וירוס נכשלה לחלוטין ונמצאה לא יעילה באיתור סוסים טרויאניים, שהתנהגותם שונה לחלוטין משל וירוסים. באופן מסורתי, מערכות ההגנה התפתחו כדי להגן מפני וירוסים מוכרים, מאחר שקיים קושי מהותי לזהות וירוסים אלה על פי התנהגותם ולא על פי מאפייני הזיהוי שלהם (חתימה). כך ניתן היה לייצר בסיסי מידע עם חתימות של וירוסים, ולהשוות קבצים ותקשורת המגיעים למחשבים מול חתימות אלה. גישה זו חייבה את יצרני תוכנות הגנה לעקוב באופן מתמשך אחר התפתחות הוירוסים כדי לייצר חתימה שלהם, ועל ידי כך להפיץ עדכונים ללקוחות כדי לאפשר להם לעדכן במהירות האפשרית את המערכות שבהן מותקנות תוכנות ההגנה שמתבססות על חתימות אלה. ההתפתחות הנרחבת בפיתוח וירוסים ופוגענים שונים וגידול עצום במספרם גרמה וגורמת לתהליך בלתי־אפשרי, המחייב השקעה של משאבים רבים בעדכון מתמשך של מאגרי נתוני החתימות של תוכנות אנטי־וירוס.

ניתן לחלק את סיכוני מתקפות סייבר באופן גס למשפחות הבאות: נוזקות, רוגלות, תולעים וסוסים טרויאניים (הפותחים 'דלתות אחוריות'[3]). חלוקה המתייחסת יותר למושא התקיפה כוללת: התקפות מתמשכות מתקדמות (Advanced Persistent Threats or APTs), שהחלו במתקפות סייבר של מדינות נגד רשתות צבאיות וארגוני ממשלה, ובשנים האחרונות התפתחו לתקיפה בעוצמה מדינתית של רשת ארגונית או תשתית קריטית אזרחית, ותקיפות של מערכות בקרה תעשייתיות המופעלות על ידי מחשבים (SCADA), כגון סטקסנט (Stuxnet). כך, מערכות של תשתיות חיוניות הנשלטות באמצעות מערכות בקרה תעשייתיות שבהן שולט פרוטוקול ה־SCADA חשופות לפגיעה העלולה להשבית את השירות החיוני, או אף לגרום נזק פיזי. נוסף לאלה – מתקפות על מערכות אלחוטיות ותחנות שידור ניידות, שימוש ברשתות חברתיות לצורכי הפצת רוגלות, נוזקות, ותקיפה של שירותי אחסון ומחשוב בענן.

מרחב התקיפה בסייבר ניתן לחלוקה הכוללת שני סוגי תקיפות שמנצלות חולשות רבות, כולל חולשות של יום־אפס:

תקיפות כלליות (Broadcast Attacks) – תקיפות המנסות לפגוע במחשבים ללא כל אבחנה. במסגרת תקיפות אלה ניתן למצוא גם הדבקה רחבה של סוכני תוכנה על מנת ליצור רשת שלמה של מחשבים שבויים (Botnet), וזאת כדי לגרום למחשבים אלה להפעיל מאוחר יותר פקודות עצמאיות, או למשוך פקודות מתוך שרת שליטה. כאמור לעיל, בדרך כלל, כשמידע על איומים חדשים מגיע לחברות האנטי־וירוס מזהים את חתימתם או חוקרים אותם באופן היוריסטי, וכך, באמצעות עדכונים שוטפים, ניתן להגן על המחשבים מפני תקיפות אלה. לאור קהל המטרה הנרחב, סביר להניח שהמידע על איומים כאלה יגיע במהרה לחברות הרלוונטיות ויוכנס לגרסאות עתידיות של מוצריהן. בחלק מהמקרים, מטרת תקיפה מסוג זה היא להגיע לכמות גדולה של מחשבים, למשל: עובדים (במקרה של התקפה על רשת ארגונית) או לקוחות (במקרה של התקפה על מוסד פיננסי, ניסיון לגנוב כרטיסי אשראי שבהם נעשה שימוש באינטרנט וכו'). לאחר הדבקת המחשב מותקן בו סוס טרויאני, המאפשר גניבת מידע או גישה מרחוק. תקיפות כאלה כוללות קוד זדוני מסוגים שונים, ואף קודים המשתַנים מהדבקה להדבקה, כדי להקשות את הגילוי באמצעות חתימה (וירוס רב־צורתי – Polymorphic Viruses). עדיין לא קיימת הגנה מלאה, משום שמפתחי סוסים טרויאניים בוחנים בצורה עקבית האם תוכנות האנטי־וירוס כבר זיהו את הקוד המפַגע וייצרו את החתימה או את קבוצת החוקים ההיוריסטיים המיירטת אותו. ברוב המקרים, אם מערכות הגילוי מצליחות לזהות את הקוד המפַגע, המפתחים מבצעים שינויים בדרך ההדבקה או ההפעלה שלו, כדי למנוע את הגילוי. לפיכך, קיימים סוסים טרויאניים רבים המצליחים להתחמק באופן עקבי מגילוי על ידי תוכנות ההגנה מובילות.

תקיפות ממוקדות (Targeted Attacks) – תקיפות אלה מתוכנתות במיוחד לצורך ספציפי ומנצלות חולשות שאינן מוכרות במערכות ההפעלה או בתוכנות מוכרות ונפוצות, תוך איתור עצמאי של חולשות חדשות. מטבע הדברים, הרוב המכריע של תוכנות האנטי־וירוס מבוסס על הגנת חתימה, הן אינן מסוגלות לזהות ולמנוע תקיפות מסוג זה, וקהל המטרה המצומצם מאפשר לתקיפות כאלה לחמוק "מתחת למכ"ם" של יצרני האנטי־וירוס. ראוי לציין שעולם האיומים מתפתח בצורה מהירה לכיוון של מתקפות ממוקדות על יעדים איכותיים.

תעבורת הנתונים ברשת תקשורת מודרנית היא רבה מאוד, בשל הצורך לספק שירותים רבים לתחנות קצה מסוגים שונים, ביניהן: מחשבים אישיים, תחנות עבודה, שרתים, מתגים וציוד תקשורת, ועוד יחידות רבות ומגוונות. באלה עושים שימוש משתמשים רבים, שברובם הגדול אינם בעלי גישת אבטחה כלשהי. כתוצאה מתופעה זו, התקפות APT מתמקדות לא רק במכונות אלא גם באנשים, לדוגמה: דרך השימוש ברשתות חברתיות. כך למשל, ההתקפה על חברת RSA שכוונה

לאנשים בארגון, והצליחה לחדור למערכות מאובטחות ביותר.[4] בשנים האחרונות אנו עדים לעלייה דרמטית בהיקף של התקפות חדשות מתוחכמות ולא מתועדות, בעלות אופי חמקני. הדבר מתבטא הן בקבוצת התקיפות הכלליות והן באלה הממוקדות. התקפות אלה מתגברות על כל ההגנות הקלאסיות והתקניות של החברות המובילות את תחום ההגנה כיום. מאחורי פיתוח אמצעי התקיפה עומדות השקעות בקנה־מידה גדול של מדינות ושל ארגוני פשיעה, כשהיקף הנזקים הוא רחב מאוד.[5] למעשה, קיימת עלייה עצומה בכמות הפוגענים המצליחים לחדור את כל מערכות ההגנה הקיימות, והמתגברים על כל ההגנות הקלאסיות המתבססות על חתימות וחוקים. העלייה הינה במאות אחוזים משנת 2011 עד ימים אלה.[6]

מערכות ההגנה הקיימות כיום מתבססות בעיקר על מניעה וסיכול של איומים מוּכרים, תוך שימוש בחתימות ובחוקים ידועים מראש. מערכות אלה אינן יכולות לגלות מתקפות יום־אפס שאין להן חתימה ידועה ברגע נתון. כך גם מתקשות המערכות הללו לזהות סוסים טרויאניים ודלתות אחוריות, כשלהתקפות מתוחכמות וחמקניות רבות אין חתימות ידועות. הם חודרים כמעט לכל מערכת מחשוב משום שהם נראים כנתונים וקודים חוקיים ואינם נראים פוגעניים. תקיפות מצליחות לחדור לרשתות הארגוניות ולמחשבי הקצה למרות כל מערכות ההגנה, בשל העובדה כי ההופעה וההתנהגות הראשונית של הפוגענים נראית חוקית ותקינה. נוסף לכך, רוב המערכות המבצעיות כיום בנויות לטיפול בסוג מסוים של התקפה, ואין להן יכולת לטפל במגוון גדול של התקפות שונות בעלות מוטציות וגרורות.

אחת הדרכים לאתר תקיפות שאינן מוּכרות ושאינן חתומות באמצעות תוכנות ההגנה המקובלות היא על ידי זיהוי התנהגות א־נורמלית של קודים שוהים במערכות הארגוניות, השונה מההתנהגות נורמלית של מרבית הנתונים. התנהגות שונה זו תסגיר את הקודים הזדוניים. בגישה זו, התנהגות לא־רגילה של רכיב תוכנה המנסה לבצע פעילות שאינה מורשית יכולה להוות בסיס אפשרי לזיהוי ולמניעת מתקפות. יצרני תוכנות ההגנה בעולם מבינים את האתגר ופועלים כדי לספק יכולות זיהוי כאלה. אולם, כאן טמון האתגר המשמעותי ביותר – הקושי לספק כלי אמין שלא יפיק התרעות שווא ושלא יפגע בצורה משמעותית בחוויית המשתמש. התרעות השווא הנן אחד האתגרים המשמעותיים ביותר של מערכות הגנה. התרעת שווא נוצרת כאשר המערכת מתריעה על קוד חוקי שהתנהגותו נורמלית, ומגדירה אותו כקוד זדוני או כחשוד ככזה. עומס רב מדי של התרעות שווא כאלה פוגע בצורה מהותית ביכולת העבודה במערכות המחשוב, ועלול לגרום למשתמשים לאבד את האמון במערכת ההגנה. אתגר שני הוא המענה לקוד זדוני החומק ממערכת ההגנה. תופעה זו קרויה false negative – כאשר מתקבלת תוצאה הנראית שלילית, אך למעשה היא חיובית. (בדומה לנשא נגיף של מחלה

חמורה, המקבל תוצאות בדיקת מעבדה שליליות באשר לנוכחות הנגיף בגופו). שני אתגרים אלה הנם ליבת העיסוק בתחום מערכות ההגנה בכלל, ובתחום השימוש בניתוח התנהגות אנומלית של קוד זדוני במערכות מידע בפרט.

זיהוי אנומליות כגישה למענה אופרטיבי

מאמר זה יתרכז בדיון על הגנה המבוססת על איתור אנומליות ברשתות תקשורת, ברבדיה השונים. הבעיה רחבה יותר וכוללת את הצורך בזיהוי אנומליות של קודים זדוניים שהוחדרו בנקודות תורפה בתוכנות וביישומים (applications). גישת הגנה זו אינה נדונה במאמר זה, אלא אם כן, הקוד הזדוני נחשף בתקשורת הארגונית. למרות האמור לעיל, ניתן להניח שחלק מהרעיונות המוזכרים מתאימים גם למציאת אנומליות בתוכנות וביישומים.

אנומליות שהוצעו לראשונה בשנת 1987[7] הן סטיות מההתנהגות המצופה, שהיא ההתנהגות הנורמלית. ההנחה הבסיסית עבור כל מערכת למציאת אנומליות היא שלנתונים זדוניים (malicious) יש מאפיינים שאינם קיימים בהתנהגות הנורמלית שאופיינה בזמן הלמידה. מאז פותחו תיאוריות ומתודולוגיות נוספות המתבססות על גישות של למידת מכונה (Machine Learning) ועל תורת האינפורמציה[8] כגון רשתות עצביות,[9] מכונת וקטורים תומכים (support vector machine),[10] אלגוריתמים גנטיים[11] ועוד רבים אחרים. כמו כן קיימות גישות רבות העושות שימוש בכריית נתונים לשם מציאת קוד זדוני.[12] סקירה כללית על איתור אנומליות מובאת במאמרם של Chandola & Banerjee[13], וכן מובא מחקר על שיטות לאיתור קוד זדוני.[14]

אחת הגישות לאיתור התקפות על נתונים מרשתות תקשורת נעשית באמצעות ניטור (monitoring) האנומליות של הפעילות הרשתית, על ידי מציאת הסטייה מפרופיל נורמלי שנלמד מנתונים שפירים (תקינים, לא־פוגעניים). מתודולוגיה זו מתבססת על כלים שנלקחו ממחקרים על למידת מכונה,[15] אנליזה מתמטית וסטוכַסטית[16], סטטיסטיקה, כריית נתונים, תורת הגרפים, תורת האינפורמציה, גיאומטריה, תורת ההסתברות ותהליכים אקראיים, ועוד. כלים של למידת מכונה וכריית נתונים בשילוב המתודולוגיות שהוזכרו משמשים בהצלחה בתחומים רבים אחרים כמו מערכות להמלצת מוצרים של Amazon,[17] Netflix,[18] זיהוי תווים אופטי,[19,20] תרגום של שפה טבעית,[21] זיהוי דוא"ל זבל (spam).[22] למידת מכונה עוסקת בפיתוח אלגוריתמים שיאפשרו למחשב ללמוד על סמך דוגמאות. קיימת למידה מונחית של נתונים ידועים מראש (supervised), שבה יודעים מראש את המשמעויות הנכונות של הפרמטרים, כלומר, לנתונים יש תוויות סיווג (labeled); בלמידה בלתי־מונחית (unsupervised) מטרת האלגוריתמים היא למצוא ייצוג

פשוט של הנתונים, ללא תוויות סיווג. למידה מונחית מוגבלת יותר מבחינת תכולת הנתונים הנלמדים, אבל מצד שני, התוצאות אמינות יותר ולכן היא עדיפה.

הלמידה הראשונית נעשית על קבוצת נתונים "בריאה", שניתן להניח שאין בה פוגענים כלשהם. קבוצה זו נקראת קבוצת הלימוד (training set). רצוי בדרך כלל ששיטת הלמידה תדע לאבחן האם חלק מקבוצת הלימוד כולל פוגענים עד אחוז מסוים מסך־כל הנתונים. ברור שאם רוב קבוצת הלימוד מכיל פוגענים, אזי הם יילָמדו ויזוהו כנתונים נורמליים. כחלק מתהליך הסינון מופעל לעיתים תהליך הנקרא 'הסרת חריג החשוד כחריג' (outlier removal), שמוציא מקבוצת הלימוד נתונים הנראים כרעשים או זיהום.

קבוצת הלימוד מנותחת על ידי מגוון שיטות מתמטיות קיימות, לצד שיטות חדשניות. באמצעות תהליך זה ניתן לאתר את המאפיינים הנורמליים של הנתונים הנבחנים. למידה מסוג זה נקראת One Class. לעומתה קיימת שיטה שבה המאפיינים נלמדים על ידי השוואת קבוצת לימוד המכילה נתונים נקיים ולא־נקיים (לדוגמה: דוא"ל עם או בלי ספאם), הנקראת Binary Class. קבוצת הלימוד נגזרת מתוך מסת הנתונים הנצברת והנשמרת בארגון יחד עם נתונים חדשים הנשמרים באופן שוטף. למטרה זו פותחו שיטות ללמידת הנתונים המאפיינים את ההתנהגות הנורמלית. הבנת הגיאומטריה[21] של הנתונים הנלמדים היא אחת משיטות הניתוח, אולם קיימות שיטות נוספות. לדוגמה: בתהליך המתואר להלן מתואר מבנה כללי אפשרי של אלגוריתמים המופעלים והמעבדים את קבוצת הלימוד, במטרה למצוא את המאפיינים של ההתנהגות הנורמלית (התקינה):

1. פירוק כל יחידת נתונים בסיסית של תקשורת או אירוע למאפיינים (features, parameters).
2. כימות היחסים בין המאפיינים. קיימות מספר שיטות לאפיון היחסים הללו. שיטת גרעין (kernel Method)[23] היא אחת מהמתודולוגיות הנפוצות שבעזרתה מגדירים יחסים בין המאפיינים. בדרך כלל נעשה שימוש בפונקציות מרחק מתמטיות להגדרת היחסים הללו. אלה הם יחסי קִרבה וריחוק שמגוון תכונות מקיימות ביניהן. לאחר שלב זה נשמרים היחסים שבין נתוני התקשורת או האירועים.
3. הורדת ממד (dimension) הנתונים. בדרך כלל ממד הנתונים גבוה והוא נקבע לפי כמות המאפיינים שמרכיבים יחידת תקשורת בסיסית או יחידת אירוע בסיסית. לכן מורידים את ממד הנתונים[22] (לדוגמה, מעשרה מימדים לשניים) תוך שימור היחסים והקוהרנטיות בין המאפיינים שאותרו בשלב הקודם. הדבר דומה לפעולת דגימה שבה בוחרים בצורה מושכלת רק חלק קטן מהנתונים המקוריים, שמייצגים אותם בצורה נאמנה. נדרשת חדשנות מתמטית, אלגוריתמית וחישובית כדי לעבד נתונים מממד גבוה שיתאימו למחשב ושייצגו בצורה טובה

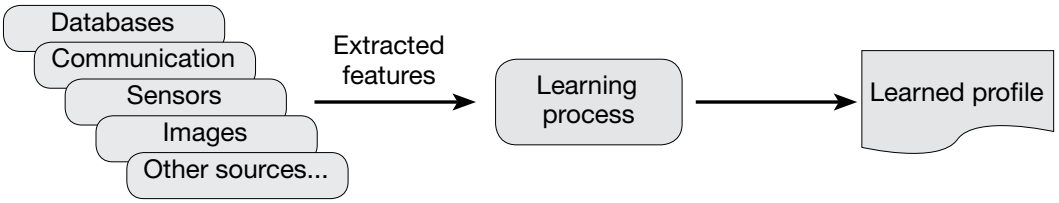
ומהימנה את נתוני המקור. הדגימה שמטרתה לצמצם את נפח הנתונים יכולה להיות אקראית, וניתן להוכיח שהיא משמרת את הקוהרנטיות של הנתונים. לשם כך יש שיטות מתמטיות רבות. אחת השיטות לייעול החישובים כדי לבנות מייצג קומפקטי של נתונים רבי־ממדים היא בניית מילונים (dictionaries),[24] שגורמים להאצה בחישובים תוך שימור היחסים והתכונות שאותרו לפני הורדת הממד. שיטות אחרות להאצת החישובים מאפשרות דילול (sparsification) של הנתונים[24,25] מטרת הגישות הללו היא אפיון הפרופיל הנורמלי של הנתונים מתוך קבוצת הלימוד, תוך התגברות על הבעיות החישוביות הכבדות בעיבוד קבוצת הלימוד. פעולת הלמידה היא בדרך כלל כבדה מבחינה חישובית. פעולה זו נעשית ברקע (offline) ואינה נדרשת לפעול בזמן אמת. שיטות נפוצות הנן: PCE,[25] LLE,[26] ISOMAP,[27] ועוד.

השיטות שתוארו לעיל מאפשרות לעבד ביעילות את קבוצת הלימוד שהיא "כבדה" ואף עלולה להיות בלתי־אפשרית מבחינה חישובית. מטרת עיבוד קבוצת הלימוד היא לאפיין את ההתנהגות הרגילה (הנורמלית) של נתוני הלימוד על סמך הבחינה של קבוצת הלימוד ועל בסיס היחסים שהוגדרו בין המאפיינים של הנתונים והאירועים של קבוצת הלימוד, בהנחה שהלמידה והמסקנות ממנה ישקפו את ההתנהגות הנורמלית של כל הנתונים החדשים העתידיים, שלא היו חלק מקבוצת הלימוד. ככל שנפח הנתונים של קבוצת הלימוד גדול יותר והמאפיינים רבים ומגוונים יותר, מאפייני ההתנהגות הנורמלית שהוסקו מקבוצת הלימוד יהיו אמינים יותר. אבל אז הסיבוכיות החישובית עולה, ולכן צריך להשקיע מאמץ רב בייצור אלגוריתמים שהם יעילים מבחינה חישובית ויכולים לטפל בנפחי נתונים גדולים.

התהליך המתואר מפרט מודל למידה אפשרי המייצר אפיון של ההתנהגות הנורמלית של הנתונים העתידיים בעזרת הפרופיל הנורמלי של קבוצת הלימוד. מכאן ואילך בודקים את המאפיינים של כל נתון חדש שמגיע או של אירוע חדש. מעבדים את המאפיינים הללו כדי לראות האם הם סוטים מהפרופיל הנורמלי שנלמד ונקבע בזמן הלימוד (אנומליה). הסטיות מהפרופיל הנורמלי צריכות לאתר את המתקפות המאופיינות כמתקפות יום־אפס. בשיטה שתוארה עד כה לא משתמשים בחתימות אלא במציאת סטיות התנהגותיות מהפרופיל הנורמלי שנוצר מעיבוד קבוצת הלימוד.

תרשים 1 הוא תיאור של תהליך הלמידה שתואר לעיל. התרשים מראה גם את מגוון המקורות שמהם נשאב המידע לצורכי הלימוד הראשוני.

תרשים1: תהליך הלמידה



השיטות הללו ונגזרותיהן למציאת פוגענים על ידי ניטור ההתנהגות של הנתונים ניתנות להפעלה בשני אופנים שונים, המשלימים זה את זה באופן השימוש בהם, כשהמשותף הוא למידה ברקע (offline) של נתוני התקשורת מהפרוטוקול שדרכו מגיעים הנתונים לארגון (כמו למשל: port 443 (HTTPS), UDP port 53 (DNS), TCP port 80 (HTTP), TCP שהם גם פרוטוקלי web), ובניית הפרופיל שמתאר את ההתנהגות הנורמלית של נתוני הפרוטוקול המסוים שאותו יש לבדוק, על פי קבוצת הלימוד.[28]

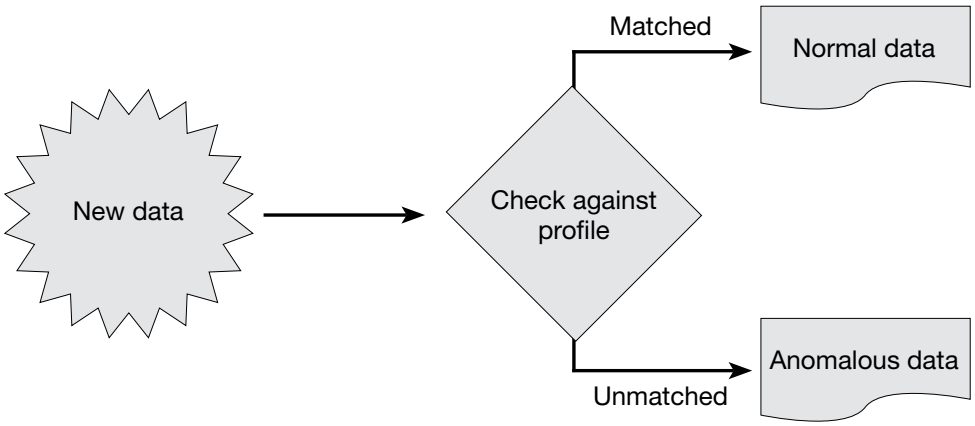
1. **הפעלה בזמן אמת** – האלגוריתם למציאת אנומליות בנתוני תקשורת (ממומש בתוכנה או בחומרה) ממוקם בכניסה לארגון, אחרי שעבר את כלי ההגנה הרגילים של IDS, IPS FireWalls (חתימות וחוקים אִפשרו לו להיכנס), ואז הוא בודק כל יחידת תקשורת – האם התנהגותה מתאימה לפרופיל הנורמלי שנלמד מקבוצת הלימוד. אם הוא התגלה כאנומליה, דרכו לתוך הארגון נחסמת. היות שלא משתמשים בחתימות, הניתוח של מהות האנומליה ייעשה אוטומטית או ידנית.
2. **הפעלה ברקע** – מציאת פוגענים ברקע. נתוני התקשורת שנכנסו לארגון דרך כל מערכות ההגנה נראים כנתונים חוקיים ומתחילים לפעול לאחר מכן, כמו רשת מרגלים שנטמעים בסביבה ומתחילים לפעול בזמנים עתידיים. לצורך כך יש לעבד לוגים ואירועים שהתרחשו בעבר ומתרחשים כעת. כדי לעבד מידע של לוגים שנשמרו וחדשים שמגיעים משתמשים כיום בטכנולוגיית Security Information and Event Management (SIEM). SIEM היא מערכת ניטור אבטחת מידע נפוצה ברשתות ארגוניות, ומשמשת כמקום מרכזי לשמירה ולפענוח של לוגים ואירועים של נתוני התקשורת. SIEM היא הארכיון של כל נתוני התקשורת והאירועים, ובעזרתה אפשר לבצע ניתוחים "משפטיים" (forensic) למציאת אנומליות.

ניתן להפעיל את השיטות למציאת אנומליות שתוארו במאמר על הנתונים ש־SIEM אספה. אפשר להפעיל גם כלי כריית נתונים אחרים על נתוני ה־SIEM. ל־SIEM יש שתי פונקציות (מרכיבים) עבור ניהול הביטחון: Security information management (SIM) ו־Security event management (SEM). בשיטה שעושה

שימוש בנתונים של SIEM צריך להפעיל בהתמדה את המתודולוגיה למציאת אנומליות כדי לאתר את פעולתם של הפוגענים, כאשר יפעלו במועד עתידי כלשהו.

תרשים 2 מתאר תהליכים לבדיקת המידע לאור תוצאות ניתוח הלמידה.

תרשים 2: תהליך הזיהוי



שימוש ב BIG-DATA למציאת אנומליות – הנתונים והאירועים מכתיבים את אופן הזיהוי

הרעיון המרכזי בבסיס מציאת האנומליות כפי שתואר לעיל הוא אפיון ההתנהגות של הנתונים בקבוצת הלימוד, והסקה מהם על התנהגות הנתונים שלא השתתפו בקבוצת הלימוד, כלומר, אפיון הנתונים החדשים שיגיעו. במילים אחרות, הנתונים מכתיבים את העיבוד, והדבר מתבטא באלגוריתמים שמכוונים ללמוד את הנתונים כפי שהם ולהסתגל אליהם. זאת בניגוד לכל ההגנות מפני פוגענים הקיימות כיום, שאין להן שום קשר להתנהגות הנתונים, אלא הן מחפשות דפוסים (patterns) של פוגענים ידועים זה כבר. במקרה של נתוני תקשורת, מנתחים את הנתונים מכל יחידת אינפורמציה של הפרוטוקול שאותו מנטרים. מוצאים את היחסים ביניהם על סמך שיטות גרעין ומשכנים אותם באופן לא־לינארי במרחבים עם ממד נמוך יותר. כך מורידים את ממד הנתונים שבדרך כלל הוא גבוה, והדבר מאפשר למצוא אנומליות באופן יעיל.

הנתונים שבהם נחפש אנומליות הם נתונים שמכונים כיום BIG-DATA. אלה נתונים בהיקף עצום הנאספים מכלל מקורות המידע הזמינים ברשת הארגונית. בארגונים רבים הם נשמרים על ידי מתודולוגיית SIEM. לפי אריק שמידט, מנהלה־לשעבר של גוגל, כמות נתונים של חמישה אקסה־בייט (Exabyte) [29] נוצרו משחר הציוויליזציה ועד שנת 2003. לטענתו של שמידט, כמות זו נוצרת עתה כל יומיים.

להלן מספר דוגמאות ליצירה של BIG-DATA: הבורסה של ניו יורק (NYSE) מייצרת מדי יום 1TB של נתונים, Facebook מייצר כל יום 20TB של נתונים דחוסים והמאיץ ב־CERN שבשווייץ מייצר מדי יום 40TB של נתונים. לפי דוח שפורסם[30], נפח הנתונים גדֵל פי שניים מדֵי שנה, ולפחות מחצית מן העסקים שומרים את הנתונים במשך שלוש שנים לפחות, לצרכים אנליטיים. חלקם מחויבים לפי חוק לשמור נתונים אלה במשך מספר שנים. מקורות חדשים בכמויות עצומות צצים כל הזמן בעסקים שונים כמו שירותים (utilities). חלק ניכר (80%) מנתונים אלה אינם מובְנים (unstructured), ולכן אינם ניתנים לשימוש יעיל בארגון. BIG-DATA הפך מקור לכריית מידע המאפשר לאתר פוגענים. לחברות רבות וידועות יש BIG-DATA יומיומי, כמו Facebook, Google, Amazon, LiveJournal, Wikipedia וזו רשימה חלקית מאוד. BIG-DATA נשמר כיום גם בענן. כמות הנתונים שנאגרת בכל ארגון היא עצומה ואף גדֵלה עם הזמן. כדי לטפל בנפחי נתונים גדולים (data silos) פותחו כלים לעיבוד BIG DATA שאינם קשורים לכריית נתונים או למציאת אנומליות כגון Hadoop[31], MapReduce[32] ו־Memcached[33,34] – מסדי נתונים מקביליים[35] עצומים שמאפשרים לבצע שאילתות מהירות עליהם. בנוסף מפתחים "צינורות" תקשורת רבים (חברת Mellanox) להעברה מהירה של כמויות הנתונים הללו. מאמץ רב מושקע בפיתוח כלים מתקדמים לעיבוד יעיל של BIG-DATA. לכן, הוא יכול להיות מקור למציאת קשת נרחבת של אנומליות התנהגותיות מתוחכמות של פוגענים שונים.

סיכום

כדי לעבד BIG-DATA ולאתר פוגענים "איכותיים" ביעילות, יש לשלב בין כל השיטות שהוזכרו לעיל. הוזכרו כלים שרובם לא־לינאריים, המצמצמים את הנפח של BIG-DATA שהוא רב־ממדי בלי לפגוע בקוהרנטיות של הנתונים, תוך הקפדה על יעילותם של האלגוריתמים, כדי לטפל בנפחי נתונים עצומים. צירוף השיטות שהזכרנו במאמר זה הוא: ביצוע למידה מתוך קבוצה קטנה של נתונים, הפעלת שיטת גרעין על הנתונים שקובעת את היחסים (המרחקים) בין נקודת הדגימה, הורדת ממד הנתונים על ידי דגימה בדידה או אקראית של הנתונים. הדבר מדלל את הנתונים וכך מקבלים "שיכון" יעיל של BIG-DATA רב־ממדי במרחב עם ממד נמוך יותר משמעותית, ובו מבצעים את זיהוי האנומליות. בניית מילונים והפעלה של אלגוריתמים חכמים ויעילים, יחד עם כלים לעיבוד BIG-DATA – כל אלה פותחים אפשרויות רבות למציאת פוגענים בכל ארגון, על ידי אפיון ההתנהגות הנורמלית ואיתור סטיות ממנה.

הגישה המוצעת היא שילוב בין אנליזה של BIG-DATA בעלת יעילות חישובית גבוהה וכלים מתקדמים למציאת אנומליות שהן הפוגענים של מתקפות יום־אפס

שאין להם עדיין חתימות ודפוסי התנהגות ידועים. המתודולוגיה שנדונה כאן חייבת למצוא "סיכה" באוקיינוס של נתונים.[36] נקודת המוצא היא שהאלגוריתמים המוצעים מתאימים את עצמם ומסתגלים לנתונים עצמם. הנתונים הם שמכתיבים את אופן פעולת האלגוריתמים. המתודולוגיה המוצעת במאמר משלבת הבנת מבנה הנתונים על ידי למידה מתוך קבוצה קטנה, והסקת מסקנות לגבי ההתנהגות העתידית של הנתונים שלא השתתפו בקבוצת הלימוד. מתודולוגיה זו מסוגלת לאתר פוגענים שפעילותם מיידית, וכאלה שנכנסו לארגון ופועלים מאוחר יותר כמו סוסים טרויאניים.

הערות

1 באופן היוריסטי הנו באמצעות חוקים המסייעים לגילוי הקוד המפַגע.

2 התנהגות אנומלית של קוד תוכנה או מידע הנה התנהגות לא־רגילה (לא־אופיינית), המעלה חשד לקיומו של פוגען במערכת.

3 פרצת אבטחה המאפשרת גישה למחשב ללא צורך באימות זהות. יכולה לנבוע משגיאת תכנות, מפרצה מכוונת בקוד מקור או כתוצאה מהתקנת תוכנה ייעודית (כגון סוס טרויאני).

4 Gabi Siboni and Y. R., "What Lies behind Chinese Cyber Warfare", *Military and Strategic Affairs*, Volume 4, No. 2, (September 2012), pp. 43-56.

5 Symantec, "Internet Security Threat Report 2011" *Trends, Vol. 17*, April 2012.

6 *FireEye Advanced Threat Report - 1H*, 2012. http://www2.fireeye.com/advanced-threat-report-1h2012.html

7 D.E., Denning, "An Intrusion-Detection Model, IEEE Trans*", Software Eng.,* Vol. SE-13 (2), 1987, pp. 222-232.

8 W. Lee and D. Xiang, "Information-Theoretic Measures for Anomaly Detection", in *Proc. IEEE Symposium on Security and Privacy*, 2001.

9 Z. Zhang, J. Li, C. Manikopoulos, J. Jorgenson, and J. Ucles," HIDE: a Hierarchical Network Intrusion Detection System Using Statistical Preprocessing and Neural Network Classification", in *Proc. IEEE Workshop on Information Assurance and Security*, 2001.

10 W. Hu, Y. Liao, and V. R. Vemuri, "Robust Anomaly Detection Using Support Vector Machines", in *Proc. International Conference on Machine Learning*, 2003.

11 C. Sinclair, L. Pierce, and S. Matzner, "An Application of Machine Learning to Network Intrusion Detection", in *Proc. Computer Security Applications Conference*, 1999.

12 M. A. Siddiqui, *Data mining methods for malware detection*, PhD Dissertation, University of Central Florida, 2008.

13 V. Chandola, A. Banerjee, V. Kumar, "Anomaly detection: A survey", *ACM Computing Surveys (CSUR)*, 41(3), article 15, 2009.

14 N. Idika, A.P. Mathur, *A survey of malware detection techniques*, Dept. of Computer Science, Purdue University, 2007.

15 R. Sommer and V. Paxson, "Outside the Closed World: On Using Machine Learning For Network Intrusion Detection", *Proc. IEEE Symposium on Security and Privacy*, May 2010.

16 תהליכים סטוכסטיים הם תהליכים שבהתפתחות שלהם במשך הזמן מעורבת מידה מסוימת של אקראיות בכל רגע נתון.

17 G. Linden, B. Smith, and J. York, "Amazon.com Recommendations: Item-to-Item Collaborative Filtering", *IEEE Internet Computing*, vol. 7, no. 1, pp. 76–80, 2003.

18 J. Bennett, S. Lanning, and N. Netflix, "The Netflix Prize", in *Proc. KDD Cup and Workshop*, 2007.

19 L. Vincent, Google Book Search: "Document Understanding on a Massive Scale", 2007.

20 R. Smith, "An Overview of the Tesseract OCR Engine", in *Proc. International Conference on Document Analysis and Recognition*, 2007.

21 F.J. Och and H. Ney, "The Alignment Template Approach to Statistical Machine Translation", *Comput. Linguist.*, vol. 30, no. 4, pp. 417–449, 2004.

22 P. Graham, "A Plan for Spam", in *Hackers & Painters*, O'Reilly, 2004.

23 B. Scholkopf and A. J. Smola, *Learning with Kernels: Support Vector Machines, regularization, Optimization, and Beyond*, The MIT Press, 2002.

24 M. Elad, "Sparse Redundant Representations", From *Theory to Applications*, Springer, 2010.

25 I.T. Jolliffe, *Principal Component Analysis*, Springer, New York, 1986.

26 S.T. Rowels, L.K. Saul, "Nonlinear dimensionality reduction by locally linear embedding", *Science*, Vol. 290 no. 5500 pp. 2323-2326, (2000).

27 J.B. Tenenbaum, V. de Silva and J.C. Langford, "A global geometric framework for nonlinear dimensionality reduction", *Science*, Vol. 290 no. 5500 pp. 2319-2323, (2000).

28 גישה זו מאפשרת גם בקרת ביצועים, ניתוח התנהגות משתמשים, ניתוח יחסי אדם־מכונה ובקרת תהליכים.

29 Exabyte = billion billion bytes

30 http://techcrunch.com/2010/08/04/schmidt-data/

31 http://hadoop.apache.org

32 J. Dean and S. Ghemawat. *MapReduce: Simplified Data Processing on Large Clusters*, *OSDI*, 2004.

33 L. Gavish, *New Caching policies for MEMCACHED*, M.Sc Thesis, Tel Aviv University, 2012.

34 B. Fitzpatrick. "Distributed caching with memcached", *Linux Journal*, 2004, http://hadapt.com/

35 M. Baker, D. Turnbull, G. Kaszuba, "Finding Needles in Haystacks (the Size of Countries)" *Blackhat*, Amsterdam, Netherlands, March 14-16, 2012.