DOI 10.18372/2310-5461.62.18705
УДК 004.056(045)

***М.Ю. Опанович*,** аспірант
Національний університет "Львівська Політехніка"
orcid.org/0000-0002-2748-2965
e-mail: maksym.y.opanovych@lpnu.ua

АНАЛІЗ СПОСОБІВ ОБХОДУ ІНСТРУМЕНТІВ ДЛЯ ЗАПОБІГАННЯМ ЗАГРОЗ ТА ДОСЛІДЖЕННЯ МЕТОДІВ ПРОТИДІЇ ЦЬОМУ

Вступ

Динамічний ландшафт кібербезпеки постійно зазнає викликів еволюцією кіберзагроз, ускладненням методів атак та збільшенням кількості векторів для атак. Оскільки технології все глибше інтегруються в усі сфери життя, складність захисту інформаційних систем зростає. Серед найбільш небезпечних елементів у цій сфері є групи Advanced Persistent Threat (APT), чиї кампанії ретельно сплановані для непомітного проникнення та перебування в цільових середовищах протягом тривалого часу. Ці групи характеризуються високим рівнем організації та значними ресурсами, якими вони користуються, часто за підтримки держав або подібних потужних утворень. Їхні операції демонструють тривожну тенденцію до ескалації кіберзагроз як з точки зору складності, так і потенційної шкоди.

Одним із критичних аспектів сучасних кіберзагроз є передові методи, які використовують зловмисники для обходу традиційних механізмів захисту, включаючи системи виявлення вторгнень, антивірусне програмне забезпечення та рішення для виявлення та реагування на кінцеві точки (EDR). Матриця Mitre ATT&CK[1], відкрита база знань про тактики, техніки та процедури, заснована на реальних спостереженнях, класифікує більше 40 стратегій, які застосовують суб'єкти загроз, щоб уникнути виявлення та аналізу. Такі методи, як обфускація, шифрування та використання вразливостей системи та мережі, дозволяють зловмисникам непомітно здійснювати свою діяльність, маскуючись під законні операції, таким чином ускладнюючи зусилля засобів кіберзахисту для ефективного виявлення та протидії таким загрозам.

Використання "Living Off The Land Binaries and Scripts" (LOLBAS)[2] є прикладом винахідливості зловмисників, які використовують вбудовані інструменти та сценарії, доступні в цільовій системі, для виконання шкідливих дій. Такий підхід зменшує ймовірність виявлення, оскільки дії не вимагають зовнішнього коду, завдяки чому вони змішуються зі звичайними системними операціями.

Вразливості в EDR та антивірусному програмному забезпеченні самі по собі можуть стати шлюзами для зловмисників. У разі використання, ці вразливості можуть вимкнути або обійти ці інструменти безпеки, залишаючи системи незахищеними та даючи зловмисникам свободу дій у мережі.

Крім того, розробка генераторів шкідливих програм є ще одним кроком у вдосконаленні кіберзагроз. Ці інструменти використовують методи машинного навчання для створення зловмисного програмного забезпечення, яке здатне адаптуватися до найновіших засобів захисту та обходити їх. Постійно розвиваючись, ці варіанти зловмисного програмного забезпечення можуть уникати виявлення на основі сигнатур і поводитися непередбачувано, роблячи традиційні механізми захисту менш ефективними проти них.

Взаємодія цих передових технологій свідчить про гонку кіберозброєнь, де розвиток оборонних технологій постійно зустрічається з новими інноваційними методами нападу. У міру того, як кіберзагрози стають складнішими, а потенційна шкода зростає, необхідність у надійних, адаптивних стратегіях безпеки стає першочерговою. Це передбачає не лише вдосконалення існуючих технологій, але й сприяння глибшому розумінню поведінки суб'єктів загрози та постійну адаптацію систем безпеки для протидії новим методам атак.

Постановка проблеми

Дослідження технік і методів обходу систем захисту має вирішальне значення з кількох вагомих причин. По-перше, він посилює рівень безпеки організацій, надаючи інформацію про потенційні вразливості в їхніх системах і мережах. Оскільки кіберзлочинці постійно вдосконалюють свій арсенал, щоб обійти традиційні заходи безпеки, розуміння методів, які вони використовують, дозволяє командам безпеки передбачати такі атаки та готуватися до них ефективніше.

Крім того, вивчення методів ухилення від виявлення допомагає в розробці більш складних і ефективних інструментів безпеки. Розбираючи, як зловмисники використовують уразливості системи, дослідники та розробники в сфері кібербезпеки можуть розробити та впровадити більш стійкі механізми захисту, які можуть виявляти та протидіяти навіть найбільш прихованим та інноваційним загрозам. Цей безперервний цикл

навчання та адаптації має важливе значення для того, щоб бути на крок попереду зловмисників, які завжди шукають нові способи зламати захист.

Також, дослідження цих методів сприяє ширшому розумінню середовища загроз, що є корисним для всієї спільноти кібербезпеки. Обмін знаннями про нові методи та успішні контрзаходи сприяє співпраці та зміцнює можливості колективної оборони. Цей спільний підхід є життєво важливим, оскільки кіберзагрози не поважають географічних чи організаційних кордонів, що робить загальну вразливість загальною проблемою.

Вивчення тактики обходу захисту є невід'ємною частиною дотримання нормативних вимог і захисту конфіденційної інформації. Багато галузей, зокрема ті, що обробляють важливі дані, як-от фінанси, охорона здоров'я та уряд, регулюються суворими правилами щодо захисту даних. Перебуваючи в курсі останніх методів ухилення, організації можуть забезпечити дотримання цих правил, уникаючи значних штрафів і репутаційної шкоди, які можуть виникнути внаслідок витоку даних.

Аналіз останніх досліджень та публікацій

У статті [3] обговорюється інноваційний підхід до виявлення загроз безпеці, які використовують легітимні інструменти, уже наявні в системі користувача – атаки Living-Off-The-Land (LOL). Ці атаки особливо складно виявити, оскільки вони використовують інструменти, яким зазвичай довіряють, в обхід багатьох традиційних заходів безпеки. Щоб вирішити цю проблему, автори представляють структуру активного навчання під назвою LOLAL, яка ітеративно покращує свої можливості виявлення шляхом аналізу невизначених і аномальних зразків командного рядка.

Стаття [4] пропонує новий алгоритм виявлення для виявлення зловмисного використання легітимних інструментів Living-Off-The-Land. Дослідження представляє метод, що використовує методи обробки природної мови, як-от регулярні вирази та одноразове кодування, для перетворення командних рядків у вектори числових токенів. Контрольовані моделі навчання потім навчаються на цих векторах, щоб ідентифікувати шаблони, які вказують на зловмисну діяльність. У роботі наголошується на труднощах у розрізненні легітимних і шкідливих команд через тонку природу атак LOL, що може призвести до великої кількості помилкових спрацьовувань у реальному середовищі.

Стаття [5] представляє новий метод підвищення шансів уникнення виявлення зловмисного програмного забезпечення за допомогою комбінування різних генераторів. У роботі досліджується, як змішування результатів різних генераторів може створити більш складне зловмисне програмне забезпечення, яке більш імовірно не буде виявлено інструментами захисту.

У статті [6] подано вичерпний огляд різноманітних тактик і прийомів, які використовує зловмисне програмне забезпечення для ухилення від динамічного аналізу. Основна увага приділяється стратегіям, які використовують зловмисне програмне забезпечення для виявлення та виходу з аналітичних середовищ, зокрема автоматизованих систем, таких як пісочниці, які використовуються для ідентифікації зловмисних дій без виконання в основній системі.

У статті [7] досліджується ефективність різних антивірусів проти антивірусу Bitdefender. Перевірені інструменти для обходу захисту включають Veil Framework, TheFatRat, Shellter, Unicorn, Venom, Phantom-Evasion, Onelinepy і MsfMania. У роботі робиться висновок, що успіх антивірусних інструментів значною мірою залежить від їх популярності та частоти оновлень, а також від розміру бази користувачів.

У статті [8] досліджує, чи застосовуються у шкідливих програмах, що належать до Advanced Persistent Threats (APT), більш складні методи анти-аналізу порівняно з більш поширеними типовими шкідливими програмами. У дослідженні було проаналізовано 1037 зразків цільового шкідливого програмного забезпечення та 16246 зразків загального шкідливого програмного забезпечення на предмет наявності методів протидії налагодженню та захисту віртуальних машин.

Постановка задачі дослідження

Основною метою цього дослідження аналіз способів обходу інструментів для виявлення загроз. На основі аналізу розробити рекомендації та способи практичної імплементації рішень для виявлення зловмисної активності націленої на обхід інструментів для виявлення загроз.

Основний матеріал

Інструменти запобігання загроз та їх недоліки

Антивірусні рішення були основою комп'ютерної безпеки протягом десятиліть. Антивірусне програмне забезпечення в першу чергу зосереджене на запобіганні, виявленні та видаленні шкідливого програмного забезпечення або шкідливих програм. Традиційні антивірусні рішення працюють шляхом сканування файлів за базою даних відомих сигнатур шкідливих програм для виявлення загроз. Сучасні антивірусні рішення розвинулися до евристичного аналізу, що дозволяє їм виявляти раніше невідомі віруси, вивчаючи шаблони поведінки, які відхиляються від норми.

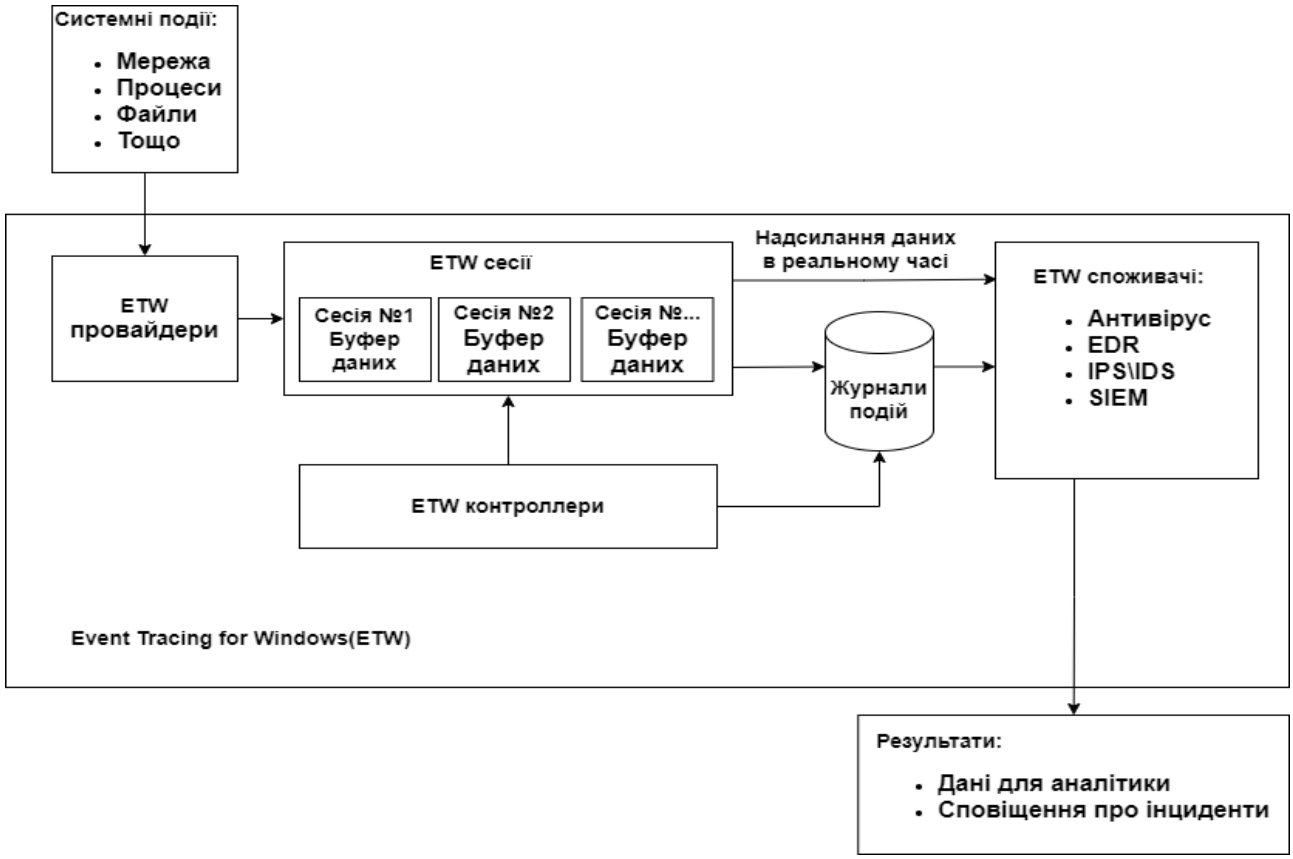

Рис. 1. Архітектура використання ETW в інструментах виявлення та протидії загроз

Рішення EDR представляють більш просунутий і динамічний підхід до кібербезпеки. Вони зосереджені не лише на виявленні зловмисних дій, але й забезпечують комплексний аналіз даних, пошук загроз та можливості реагування. Вони відстежують і збирають дані з кінцевих в режимі реального часу, використовуючи поведінкову аналітику для виявлення підозрілих дій, які можуть свідчити про злом або атаку в процесі.

Ці та інші рішення на сьогоднішній день використовують Event Tracing for Windows (ETW) – вбудований механізм ведення журналів Windows, призначений для спостереження та аналізу поведінки додатків. ETW був представлений досить давно (Windows XP) як фреймворк, впроваджений в ядро для діагностики поведінки компонентів ОС і проблем з продуктивністю. З тих пір він був значно розширений і вдосконалений. В Windows 11 ETW може генерувати більше 50 000 типів подій від близько 1000 постачальників.

Використання ETW для збору телеметрії кінцевої точки має наступні переваги:

- доступний у всіх останніх версіях ОС Windows без необхідності встановлення;
- підтримує стандартизований фреймворк для створення та використання журналів подій;
- високошвидкісний моніторинг, що дозволяє програмам надсилати події в реальному часі або з LOG-файлів.

Один з перших прикладів використання інструментів на основі ETW для аналізу та виявлення поведінки шкідливого програмного забезпечення був представлений Марком Руссиновичем в його доповіді “Malware Hunting with the Sysinternals Tools” [9]. З тих пір розробники сучасних ПЗ використовують ETW для моніторингу подій, пов’язаних з безпекою, а також для успішного виявлення та реагування на найсучасніше шкідливе програмне забезпечення.

Незважаючи на дослідницькі зусилля та передові механізми безпеки, розгорнуті за допомогою цих інструментів, нещодавні дослідження та тести [10, 11] демонструють, що навіть найдосконаліші рішення не можуть самотужки забезпечити надійний захист. Дані праці, через моделювання атак APT груп, продемонстрували 2 ключові проблеми:

- більшість найсучасніших систем EDR не змогли запобігти або зареєструвати значну частину сценаріїв атак, представлених у дослідженні. Це свідчить про значну прогалину в можливостях сучасних EDR рішень протистояти складним атакам;
- роботи показали методи за допомогою яких зловмисники можуть маніпулювати провайдерами телеметрії EDR, що дозволяє їм виконувати атаки непомітно.

Окрім цього слід зауважити, що будь який інструмент для захисту – це перш за все програмне

забезпечення і як будь яке програмне забезпечення, за своєю природою має свої вразливості та слабкості. Можна виділити 3 основні типи вразливостей, а саме вразливість самого програмного забезпечення, логічні вразливості та вразливості в проектуванні.

Як приклади вразливостей можна навести:

- шкідлива програма Aikido [12] маніпулювала вказівником файлової системи NTFS для того, щоб EDR видаляв легітимні файли, замість шкідливих;
- вразливість в Crowdstrike Falcon [13] дозволяла обходити захист від несанкціонованого доступу, та видаляти сенсор з кінцевої точки;
- вразливість в продуктах McAfee [14] дозволяла редагувати реєстри свої інструментів, внаслідок чого, було можливо виконували довільні DLL-файли.

В свою чергу, логічні вразливості полягають у експлуатації нормальної роботи інструментів безпеки на користь зловмисників. Наприклад, EDR та антивірусні рішення є досить агресивними по відношенню до EXE-файлів. Як тільки вони помітять шкідливі патерни в файлі він найімовірніше буде видалений або поміщений у карантин. Але при цьому логіка виявлення загроз даних інструментів більш імовірно дозволить виконання такого ж файлу у форматі DLL, категоризуючи його як підозрілий, а не зловмисний. Хоч така дія і створить сповіщення, про підозрілу активність, та зловмисний файл все одно буди виконаний.

Вразливості пов'язані з проектуванням інструментів експлуатують архітектуру та процес роботи рішень безпеки. Дослідження Binary [15] показало, що ETW має вразливості, які можуть бути використані для зупинки роботи цього компоненту, що, у свою чергу, засліплює інструменти безпеки, що його використовують. Вимкнення ETW впливає не лише на EDR чи антивіруси, але й на звичайні інструменти діагностики, такі як Process Monitor, які покладаються на ETW для збору системних подій. Це дозволяє шкідливому програмному забезпеченню працювати непоміченим в уражених системах.

Підсумовуючи можна зазначити, шо будь який EDR чи антивірус, так само як і інші схожі інструменти для виявлення і запобігання загроз, є програмним забезпеченням, що мають свої вразливості та недоліки. Зловмисники мають можливість досліджувати механізми роботи інструментів, їхню логіку, тим самим можуть знаходити вразливості, які можуть використовуватись в подальших атаках. Враховуючи те, що атаки APT груп починаються з розвідки, зловмисники можуть дізнатись про засоби безпеки і підготуватись до атаки відповідно.

Техніки обходу захисту

Матриця технік Mitre містить 35 описаних технік для обходу систем виявлення і запобігання загрозам в операційній системі Windows. Їх можна розділити на 2 категорії: обхід вбудованих мір захисту та контролю доступу та обхід інструментів захисту. В даній роботі будуть розглядатись саме методи обходу інструментів захисту.

Аналізуючи матрицю технік можна розділити техніки на наступні категорії:

- **Обфускація.** Зловмисники можуть спробувати ускладнити виявлення або аналіз виконуваного файлу або файлу шляхом шифрування, кодування або іншим чином замаскувати його вміст у системі або під час передачі. Корисне навантаження може бути стиснене, архівоване або зашифроване, щоб уникнути виявлення. Це може бути використано під час початкового доступу або пізніше, щоб зменшити ймовірність виявлення. Частини файлів також можуть бути зашифровані, щоб приховати рядки відкритого тексту, які в іншому випадку допомогли б захисникам у виявленні. Корисне навантаження також може бути розділене на окремі, на перший погляд, безпечні файли, які виявляють шкідливу функціональність лише при повторній збірці;
- **Експлуатація вразливостей.** Зловмисники можуть використати вразливість системи або програми, щоб обійти засоби захисту. Використання вразливості відбувається, коли зловмисник використовує програмну помилку в програмі, службі або в самому програмному забезпеченні чи ядрі операційної системи для виконання коду, контрольованого зловмисником. Вразливості можуть існувати в захисному програмному забезпеченні, яке може бути використане для його вимкнення або обходу. Зловмисники можуть мати попередню інформацію через розвідку, що в середовищі існує захисне програмне забезпечення, або вони можуть виконати перевірку під час або незабаром після того, як система була скомпрометована, для виявлення захисного програмного забезпечення. Програмне забезпечення безпеки, швидше за все, буде безпосередньо націлене на експлуатацію;
- **Виявлення інструментів налагодження та віртуалізації.** Зловмисники можуть використовувати різні засоби для виявлення та уникнення відладчиків. Вони зазвичай використовуються захисниками для відстеження та/або аналізу виконання потенційного корисного навантаження шкідливого програмного забезпечення. Також зловмисники можуть використовувати різні засоби для виявлення та уникнення середовищ віртуалізації та аналізу. Це може включати зміну поведінки на

основі результатів перевірок на наявність артефактів, що вказують на середовище віртуальної машини або пісочницю. Якщо противник виявляє віртуальне середовище, він може змінити своє шкідливе програмне забезпечення, щоб від'єднатися від жертви або приховати основні функції імплантату. Противник може використовувати інформацію, отриману в результаті обходу віртуалізації/пісочниці під час автоматизованого виявлення, для формування подальшої поведінки;

- **Приховування артефактів.** Зловмисники можуть намагатися приховати артефакти, пов'язані з їхньою поведінкою, щоб уникнути виявлення. Так вони можуть приховувати вікна програм, файли, аргументи командного рядка процесу шляхом перезапису пам'яті процесу або можуть обійти захисні механізми, виконуючи команди, які ховаються від сигналів переривання процесу, а також використовувати поштові правила для приховування вхідних листів у поштовій скриньці скомпрометованого користувача;
- **Ін'єкції.** Зловмисники можуть впроваджувати код у процеси, щоб обійти захист на основі процесів, а також, можливо, підвищити привілеї. Ін'єкція в процес - це метод виконання довільного коду в адресному просторі окремого поточного процесу. Запуск коду в контексті іншого процесу може дозволити доступ до пам'яті процесу, системних/мережевих ресурсів і, можливо, підвищити привілеї. Виконання через ін'єкцію процесу може також уникнути виявлення продуктами безпеки, оскільки виконання маскується під легітимний процес. Більш складні реалізації можуть виконувати кілька ін'єкцій у процес для сегментації модулів і подальшого уникнення виявлення, використовуючи іменовані канали або інші механізми міжпроцесної взаємодії (IPC) як канал зв'язку;
- **Непряме виконання команд.** Зловмисники можуть зловживати утилітами, які дозволяють виконувати команди в обхід обмежень безпеки, що обмежують використання інтерпретаторів командного рядка. Для виконання команд можуть використовуватися різні утиліти Windows, можливо, без виклику cmd. Наприклад, Forfiles, помічник сумісності програм (pcalua.exe), компоненти підсистеми Windows для Linux (WSL), а також інші утиліти можуть викликати виконання програм і команд з інтерпретатора команд і сценаріїв, вікна виконання. Зловмисники можуть зловживати цими можливостями для обходу захисту, зокрема, для довільного виконання, обходячи засоби виявлення та/або запобігання (наприклад, групові політики), які обмежують/запобігають використанню команд або розширень файлів, що найчастіше асоціюються зі шкідливим програмним навантаженням;
- **Маскування.** Зловмисники можуть намагатися маніпулювати властивостями своїх артефактів, щоб вони виглядали легітимними або безпечними для користувачів і/або засобів захисту. Маскування відбувається, коли ім'я або місцезнаходження об'єкта, легітимного чи зловмисного, змінюють з метою уникнення засобів захисту та спостереження. Це може включати маніпуляції з метаданими файлів, введення користувачів в оману щодо неправильної ідентифікації типу файлу та надання легітимних назв завдань або служб.

Усі ці техніки широко використовуються APT групами, вірусами-вимагачами і іншими типами шкідливого програмного забезпечення. Оскільки, навіть без сторонніх інструментів запобіганням загроз та їх виявлення, у Windows містяться механізми захисту інформації, контролю доступів та вбудований антивірус, зловмисники змушені використовувати механізми обходу захисту або впроваджувати їх у свої інструменти, для успішного виконання.

Інструменти для обходу захисту

Разом з розвитком технологій та інструментів для захисту і виявлення загроз, розвивались і створювались спеціальні інструменти для обходу захисту. Хоча мануальні методи обходу захисту та їх імплементація в інструменти зловмисників, безумовно, можливі, використання спеціалізованих інструментів для уникання захисних механізмів дає зловмисникам кілька переваг:

- **Продуктивність**: Спеціалізовані інструменти часто призначені для автоматизації процесу ухилення від виявлення, заощаджуючи зловмисникам час і зусилля. Ручні методи обходу вимагають значного досвіду для ефективної реалізації. Використовуючи інструменти, спеціально розроблені для обходу захисту, зловмисники можуть оптимізувати свої операції і зосередитися на інших аспектах своїх атак;
- **Ефективність**: Спеціалізовані інструменти для обходу часто ефективніше обходять заходи безпеки, ніж ручні методи. Ці інструменти зазвичай розробляються і підтримуються досвідченими зловмисниками, які розуміють останні тенденції в кібербезпеці і можуть відповідно адаптувати свою тактику обходу. Як наслідок, вони можуть бути краще підготовлені до того, щоб уникнути виявлення сучасними технологіями безпеки;
- **Адаптивність**: Спеціалізовані інструменти ухилення часто регулярно оновлюються, щоб включати нові методи ухилення та контрзаходи проти нових технологій безпеки. Зловмисники можуть використовувати ці інструменти, щоб випереджати захисників і продовжувати уникати виявлення, навіть якщо технології безпеки розвиваються.

В даній роботі розглядались і аналізувались наступні інструменти: ScareCrow [16], Ivy [17], Shellter [18], VENOM [19]. Дані інструменти мають різні реалізації методів уникнення механізмів виявлення та запобігання загрозам.

Venom – це генератор/компілятор шелл-коду Metasploit. Venom використовує MSFvenom з фреймворку Metasploit для генерації шелл-коду в різних форматах: C, Python, Ruby, DLL, MSI, HTA-PSH і вставляє згенерований код в одну функцію. Шелл-код виконується в пам'яті функцією однієї з мов і використовує компілятор, для створення виконуваного файлу і одночасно запускає мультиобробник для обробки віддаленого з'єднання.

Shellter – це інструмент, який дозволяє впроваджувати корисне навантаження у легітимний виконуваний файл Windows (EXE). Таке вбудовування дозволяє замаскувати корисне навантаження під справжній виконуваний файл, що може значно підвищити шанси на обхід антивірусу. Shellter перекодовує будь-який власний 32-розрядний автономний додаток Windows для вбудовування користувацького командного коду або будь-якого корисного навантаження, створеного за допомогою msfvenom, щоб обійти антивірус.

Ivy – це фреймворк для створення корисного навантаження для виконання довільного вихідного коду VBA (макросів) у пам'яті. Завантажувач Ivy робить це, зловживаючи програмним доступом в об'єктному середовищі VBA для завантаження, розшифровки та виконання командного коду. Ця методика максимально наближена до справді без файлової, оскільки більшість без файлових атак сьогодні вимагають скидання певних файлів на диск, що дозволяє обійти стандартні правила виявлення коду VBA, засновані на сигнатурах.

ScareCrow – це фреймворк для створення корисного навантаження, яке завантажується в легітимний процес Windows в обхід елементів керування білим списком додатків. Після завантаження в пам'ять завантажувач DLL використовує техніку для видалення гачка EDR з системних DLL-файлів, запущених в пам'яті процесу. ScareCrow може вибирати ці DLL і маніпулювати ними в пам'яті, використовуючи функцію API VirtualProtect, яка змінює секцію дозволів пам'яті процесу на інше значення, зокрема, з "Execute–Read" на "Read-Write-Execute".

Дані інструменти значно спрощують і пришвидшують підготовку до атаки, так як, час створення та налаштування зловмисної програми скорочується до декількох хвилин.

LOLBAS техніки та інструменти

Як вже було зазначено раніше LOLBAS техніки являють собою використання легітимних інструментів і служб та сервісів, які є частиною операційної системи Windows або легітимних програм сторонніх виробників. Оскільки існуючі антивіруси та засоби захисту кінцевих точок продовжують вдосконалюватися у виявленні файлових шкідливих програм, досвідчені зловмисники шукають інші шляхи, щоб залишатися невикритими. Одним з таких методів є використання існуючих інструментів у цільовій системі, встановлених як частина операційної системи з легітимною метою.

Ці інструменти та програми не є шкідливими за своєю суттю, але можуть бути використані зловмисниками у своїх цілях. Вони пропонують потужну функціональність для системного адміністрування, усунення несправностей та інших адміністративних дій. Зловмисники можуть зловживати інструментами LOLBAS для виконання шкідливих команд, обходу засобів контролю безпеки, підвищення привілеїв і підтримки стійкості в скомпрометованих системах. Наприклад:

- Зловмисники можуть використовувати PowerShell для завантаження і виконання шкідливого навантаження або regsvr32 для завантаження і виконання DLL-файлів з віддалених місць;
- certutil.exe – це програма командного рядка Windows, яка використовується для управління сертифікатами, таких як налаштування служб сертифікатів, перевірка сертифікатів і пар ключів. Вона також має функціонал для завантаження файлів з Інтернету та кодування чи декодування файлів сертифікатів, що дозволяє зловмисникам використовувати цей інструмент для заван-таження шкідливих файлів або приховування існуючих файлів.

Методи LOLBAS часто включають в себе ухилення від виявлення програмним забезпеченням безпеки та обхід засобів контролю безпеки. Використовуючи легальні інструменти та методи, зловмисники можуть змішатися зі звичайною активністю системи і не викликати підозр. Традиційні антивірусні рішення можуть бути не в змозі виявити таку поведінку. Сучасні рішення для виявлення загроз, які намагаються виявити такі атаки, як правило, базуються на евристиці та співставленні регулярних виразів. Оскільки ці LOLBAS інструменти також можуть використовуватися звичайними користувачами, такими як системні адміністратори або розробники, ці методи зазвичай призводять до великої кількості хибних спрацьовувань. Окрім того, виявлення активності LOLBAS може бути складним завданням для традиційних рішень безпеки, оскільки цим інструментам вже довіряє система і вони можуть не викликати тривоги. На сьогоднішній день існує близько 200 компонентів операційної системи та сторонніх програм які можна віднести до LOLBAS.

Генератори шкідливих програм

В останні роки популярність алгоритмів машинного навчання значно зросла, у тому числі і в сфері кібербезпеки. Машинне навчання все частіше стає ключовим інструментом для посилення заходів кібербезпеки, особливо у сфері виявлення шкідливого програмного забезпечення. Така інтеграція зумовлена необхідністю боротися з постійно зростаючими та витонченими загрозами, які створюють сучасні шкідливі програми. Традиційні підходи до виявлення шкідливих програм, які часто покладаються на ідентифікацію відомих сигнатур, не встигають за швидким розповсюдженням унікального та модифікованого шкідливого програмного забезпечення. Машинне навчання пропонує динамічний і проактивний підхід до цієї проблеми, використовуючи свою здатність вчитися на даних для виявлення шаблонів, які вказують на зловмисну активність, навіть якщо конкретні сигнатури шкідливого програмного забезпечення раніше не були відомі.

Разом з розвитком машинного навчання у сфері захисту, з'явились дослідження того, як зловмисники можуть використовувати його у свої цілях. Для цих цілей були створені генератори шкідливих програм. Наприклад:

- **MAB-Malware**: Цей генератор використовує алгоритм навчання з підкріпленням, відомий як багаторукий бандит (MAB). Він працює без урахування стану, тобто не враховує порядок маніпуляцій з файлами. Він вносить зміни до файлів, доки цільовий детектор не класифікує зразок як доброякісний або не досягне ліміту змін, з другою фазою мінімізації дій для видалення непотрібних модифікацій;
- **AMG** (Adversarial Malware Generator): Цей генератор доступний у двох варіантах. Перший варіант це навчений, що використовує алгоритм проксимальної оптимізації політики (PPO) для визначення найкращих дій на основі вивчених політик. І другий варіант є випадковий, що вибирає дії випадковим чином із заздалегідь визначеного набору маніпуляцій з файлами PE. Обидві версії працюють з файлами до тих пір, поки вони не уникнуть цільового класифікатора або не досягнуть максимальної кількості змін;
- **FGSM**(Fast Gradient Sign Method): Спочатку розроблений для обробки зображень, цей метод був адаптований для шкідливих програм. Він спотворює невелику частину байт файлу шкідливого програмного забезпечення, а потім намагається вставити ці зміни у вихідний файл. Процес змін повторюється доти, доки шкідливе програмне забезпечення не уникне виявлення або доки не буде досягнуто максимальної кількості ітерацій.

Такі рішення значно спрощують і пришвидшують обфускацію зловмисних файлів, що дозволяє обійти сигнатурний аналіз та класифікатори зловмисних програм які побудовані на алгоритмах машинного навчання.

Способи протидії та виявлення

Кількість векторів, методів та інструментів для атак постійно зростає. Тому сфера кібербезпеки завжди потребувала і потребує комплексного підходу. Як показує аналіз методів та засобів обходу систем та механізмів захисту, команди захисту не можуть сподіватись на окремі інструменти. Вони потребують комплексних підходів, таких як, багатошаровий та глибокий(Defense-in-Depth) захист, а також дотримання концепції нульової довіри(Zero Trust) [20] та постійного зміцнення оборони (Security Hardening).

Жоден окремий інструмент не може забезпечити повноцінного захисту і, окрім того, кожен інструмент – це програмне забезпечення, яке по своїй природі може мати недосконалості та вразливості. Як показало дослідження, такі інструменти як EDR чи антивіруси мають досить багато вразливостей, які дозволяють обійти захист, або знешкодити ці інструменти. Для нівелювання даної проблеми потрібно впроваджувати багатошаровий захист.

Принцип багатошарового захисту полягає у впровадження інструментів та механізмів захисту на кожному шарі інфраструктури яку захищають, навіть якщо вони частково чи повністю дублікують функціонал один одного. Такий підхід гарантує, що якщо якийсь з рівнів захисту вийде з ладу чи пропустить атаку, наступні шари будуть функціонувати та зможуть виявити та зупинити чи ускладнити атаку.

Концепція нульової довіри передбачає відсутність довіри до будь-якого користувача чи пристрою. Вона спрямована на захист чутливих ресурсів шляхом впровадження суворого контролю доступу та постійного моніторингу і перевірки поведінки пристроїв та користувачів. В контексті дослідження обходу інструментів захисту, керуючись концепцією нульової довіри, необхідно звести до мінімуму кількість пристроїв та користувачів який довіряють системи захисту. Для зменшення кількості хибних спрацювань, що зросте через збільшення кількості об'єктів під моніторингом, необхідно робити максимально деталізовані та таргетовані виключення з правил виявлення. Також в даному контексті є необхідним дотримуватись принципу найнижчих привілеїв, надаючи користувачам, пристроям та програмам найнижчих доступів та привілеїв необхідних для їхньої роботи.

Принцип зміцнення оборони – це підхід, який націлений на посилення захисту та мінімізацію поверхні потенційних атак, а також на зниження кількості векторів можливих загроз. Це досягається за допомогою ряду мір, таких як, використання усіх наявних механізмів та інструментів безпеки, вимкнення усіх непотрібних сервісів та аккаунтів, шифрування пристроїв, сегментації мережі, використання захищений протоколів передачі даних.

Також усі ці концепції передбачають постійний активний моніторинг інфраструктури. Для його ефективної імплементації потрібно, перш за все, налаштувати логування подій системи і використовувати SIEM рішення. Роль SIEM (Security information and event management) полягаю у агрегації усіх журналів подій, їх нормалізації та обробці, щоб на основі цих даних будувати правила виявлення загроз та проводити розслідування інцидентів безпеки. В операційній системі Windows та середовищі Active Directory найкращим способом забезпечити ефективний моніторинг є комбінування стандартних журналів подій та журналів з утиліти Sysmon [21]. Ця комбінація не тільки зможе забезпечити виявлення великого спектру атак, але і продукує значно меншу кількість журналів подій, що у свою чергу, зменшує навантаження на кінцеві точки та SIEM. Також журнали з усіх інструментів безпеки можуть та повинні надсилатись до SIEM.

За допомогою агрегованих даних в SIEM можна побудувати правила виявлення загроз, що можуть бути пропущеними такими інструментами як EDR чи антивірус. Так виявлення LOLBAS атак є ефективнішим за допомогою правил виявлення в SIEM. На основі даних про процеси з утиліти Sysmon можна створити ефективні правила виявлення зловмисної експлуатації легітимних програм та компонентів операційної системи. Тому навіть, якщо зловмисники зможуть обійти інструменти протидії загрозам, атака все одно буде виявлена. Також правила виявлення в SIEM будуть ефективними проти загроз від зловмисних файлів, створених за допомогою спеціальних інструментів чи генераторів, оскільки ці правила побудовані на виявленні зловмисної поведінки, а не на виявленні сигнатур чи патернів у коді шкідливої програми.

Також необхідно створити правили виявлення для обходу захисту від несанкціонованого доступу до інструментів захисту та джерел журналів подій, враховуючи їхні вразливості. Завдяки подіям в утиліті Sysmon можна побудувати правила виявлення для маніпуляцій з процесами, таких як, завантаження бібліотек в процес чи створення каналів в процесах, завдяки яким зловмисники найчастіше намагаються знешкодити інструменти та механізми захисту. Окрім цього, необхідно створити для виявлення зміни стану інструментів та механізмів безпеки, для отримань сповіщень, що певний інструмент чи джерело журналів подій не функціонує протягом певного часу.

Висновки

Аналіз способів обходу інструментів та механізмів виявлення та протидії загрозам показав, що зловмисники активно створюють та покращують свої інструменти та техніки для уникнення виявлення своєї діяльності. Це було підтверджено численними дослідженнями та прикладами реальних атак. Зловмисники постійно досліджують інструменти захисту та на основі цього навчились використовувати вразливості цих інструментів для уникнення виявлення. Окрім цього постійний розвиток інструментів для обходу захисту, в тому числі і генераторів шкідливих програм, за допомогою яких, зловмисники обфусковують корисне навантаження шкідливих файлів, що дозволяють доставляти та виконувати шкідливі програми непомічено, створюють нові виклики для сфери кіберзахисту.

Все більше використання легітимних інструментів в зловмисних цілях поступово стає трендом в атаках. Завдяки цьому зловмисники можуть змішати зловмисну активність з легітимною. Враховуючи, що зазвичай більшість легітимних програм та компонентів операційної системи вважаються довіреними, такі інструменти як EDR чи антивірус дозволяють їх виконання. Разом це створює значні перешкоди у виявленні загроз.

Як показало дослідження жоден окремий інструмент не може забезпечити повноцінний захист. Для забезпечення надійного захисту інфраструктури необхідно дотримуватись принципу постійного зміцнення захисту та впроваджувати комплексні підходи, до яких входять стратегічні, організаційні та тактичні міри. До них входять впровадження концепцій глибокого захисту та нульової довіри, створення політик аудиту інфраструктури та правил виявлення зловмисної активності, що можу бути не виявлена такими засобами як EDR чи антивірус.

Впровадження цих мір знижує поверхню та кількість векторів атак, що значно ускладнює їх виконання. Навіть якщо зловмисника вдасться обійти одну з мір захисту, наступні зможуть зупинити чи виявити атаку.

З огляду на результати дослідження можна виділити наступні напрямки для подальших досліджень. Перш за все, важливо зосередитися на розробці більш стійких та адаптивних механізмів захисту, які можуть ефективно протидіяти складним

атакам, що використовують обфускацію, шифрування та експлуатацію вразливостей. Дослідження в цій галузі може включати розробку нових алгоритмів для виявлення та нейтралізації загроз на основі поведінкових моделей та машинного навчання.

Крім того, важливим напрямком є подальше вивчення і вдосконалення систем SIEM (Security Information and Event Management), які можуть покращити можливості виявлення загроз шляхом аналізу та кореляції великих обсягів даних. Це також включає розвиток методів для виявлення атак, що використовують легітимні інструменти та скрипти (LOLBAS), і впровадження комп-лексних стратегій безпеки, таких як глибокий захист і нульова довіра. Вивчення цих напрямків допоможе створити більш надійні системи кібербезпеки, здатні ефективно реагувати на нові та еволюціонуючі загрози.

ЛІТЕРАТУРА

[1] LOLBAS..lolbas-project.github.io..URL: (access data 14.05.2024).

[2] Mitre Enterprise Matrix. attack.mitre.org. URL: https://attack.mitre.org/matrices/enterprise/ (access data 14.05.2024).

[3] Talha Ongun, Jack W. Stokes, Jonathan Bar Or, Ke Tian, Farid Tajaddodianfar, Joshua Neil, Christian Seifert, Alina Oprea, and John C. Platt (2021). Living-Off-The-Land Command Detection Using Active Learning. In Proceedings of the 24th International Symposium on Research in Attacks, Intrusions and Defenses (RAID '21). Association for Computing Machinery, New York, NY, USA, 442–455. https://doi.org/10.1145/3471621. 3471858.

[4] Stamp, Ryan. (2022). Living-off-the-Land Abuse Detection Using Natural Language Processing and Supervised.Learning. https://doi.org/10.48550/arXiv. 2208.12836.

[5] Kozák, Matouš & Jureček, Martin. (2023). Combining Generators of Adversarial Malware Examples to Increase Evasion Rate. https://doi.org/10.48550/ arXiv.2304.07360.

[6] Afianian, Amir & Niksefat, Salman & Sadeghiyan, Babak & Baptiste, David. (2018). Malware Dynamic Analysis Evasion Techniques: A Survey. https://doi.org/10.48550/arXiv.1811.01190.

[7] Aminu S. A., Sufyanu Z., Sani T., & IdrisA. (2020). Evaluating the effectiveness of antivirus evasion tools against windows platform. *Fudma journal of sciences*, 4(1), 112–119. URL: https://fjs.fudutsinma.edu.ng/index.php/fjs/article/view/27.

[8] Ping Chen, Christophe Huygens, Lieven Desmet, Wouter Joosen (2016). Advanced or Not? A Comparative Study of the Use of Anti-debugging and Anti-VM Techniques in Generic and Targeted Malware. 31st IFIP International Information Security and Privacy Conference (SEC), May 2016, Ghent, Belgium. pp.323-336, ff10.1007/978-3-319-33630-5 22ff. ffhal-01369566f.

[9] Russinovich M. (2015). Malware Hunting with the Sysinternals Tools. RSAConference-2015: Presentation, San Francisco, 20–24 April 2015.

[10] Karantzas G, Patsakis C. (2021). An Empirical Assessment of Endpoint Detection and Response Systems against Advanced Persistent Threats Attack Vectors. *Journal of Cybersecurity and Privacy*. 1(3):387–421. https://doi.org/10.3390/ jcp1030021.

[11] Advanced Threat Protection Test 2023 – Enterprise. av-comparatives.org. URL: https://www.av-comparatives.org/tests/advanced-threat-protection- test-2023-enterprise/. (access data 14.05.2024).

[12] Vijayan J. For Cyberattackers, Popular EDR Tools Can Turn into Destructive Data Wipers. darkreading.com.URL: https://www.darkreading.com/ vulnerabilities-threats/cyberattackers-popular-edr-tools-destructive-data-wipers. (access data 14.05.2024).

[13] CVE-2022-44721 Crowdstrike Falcon Uninstaller. github.com.URL: https://github.com/gmh5225/ CVE-2022-44721-CsFalconUninstaller. (access data 14.05.2024).

[14] Discovering Zero-Day Vulnerabilities in McAfee Products | mr.d0x. Security Research | mr.d0x. URL: https://mrd0x.com/discovering-mcafee-products-zero-day-vulnerabilities/ (access data 14.05.2024)

[15] Modern EDR Design Issues: Bypassing ETW-Based Solutions. Firmware Security | Supply Chain Risk Management | BINARLY. URL: https://www.binarly.io/blog/design-issues-of-modern-edrs-bypassing-etw-based-solutions. (access data 14.05.2024)

[16] GitHub - Tylous/ScareCrow: ScareCrow - Payload creation framework designed around EDR bypass. GitHub.URL:https://github.com/Tylous/ScareCrow (access data 15.05.2024)

[17] GitHub-Tylous/Ivy:GitHub. URL: https://github.com/ Tylous/Ivy. (access data 15.05.2024).

[18] Shellter | AV Evasion Artware. Shellter | AV EvasionArtware.URL:https://www.shellterproject.com/. (access data 16.05.2024)

[19] GitHub – r00t-3xp10it/venom: venom - C2 shellcode generator/compiler/handler. GitHub. URL: https://github.com/r00t-3xp10it/venom. (access data 15.05.2024)

[20] Журавчак, Д., Глущенко , П., Опанович , М., Дудикевич , В., & Піскозуб , А. (2023). Концепція нульової довіри для захисту Active Directory для виявлення програм-вимагачів. Кібербезпека: освіта, наука, техніка, 2(22), 179–190. https://doi.org/10.28925/2663-4023.2023.22. 179190

[21] Opanovych, M. (2024). Enhancing Active Directory Security Monitoring with Sysmon. The Science of Tomorrow: Innovative Approaches and Forecasts. (pp. 60–64). Futurity Research Publishing.

Опанович М. Ю.

АНАЛІЗ СПОСОБІВ ОБХОДУ ІНСТРУМЕНТІВ ДЛЯ ЗАПОБІГАННЯМ ЗАГРОЗ ТА ДОСЛІДЖЕННЯ МЕТОДІВ ПРОТИДІЇ ЦЬОМУ

У цьому дослідженні розглядаються методи ухилення, що застосовуються групами Advanced Persistent Threat (APT), які обходять традиційні засоби захисту, такі як антивірусне програмне забезпечення та системи виявлення і реагування на кінцевих точках (EDR). Основна увага приділяється таким методам, як обфускація, шифрування та використання вразливостей інструментів захисту, які визначені як суттєві виклики для ефективного виявлення загроз. Досліджується, як зловмисники використовують " Living Off The Land Binaries and Scripts" (LOLBAS) для маніпулювання вбудованими системними інструментами для прихованих атак, органічно поєднуючи зловмисну діяльність з легітимними процесами, щоб уникнути їх виявлення. Крім того, обговорюється роль систем безпеки інформації та керування подіями (SIEM) у покращенні можливостей виявлення. Системи SIEM виділяються своєю здатністю збирати й аналізувати дані безпеки в мережі, забезпечуючи цілісне уявлення, яке допомагає ранньому виявленні потенційних загроз і порушень, таким чином зміцнюючи рівень безпеки організації. Проаналізовані вразливості в самих EDR та антивірусних рішеннях, показуючи, як вони можуть бути використані для вимкнення або обходу цих критично важливих засобів захисту, залишаючи системи вразливими. Крім того, розглядається розробка генераторів шкідливого програмного забезпечення з використанням методів машинного навчання. Ці інструменти, здатні створювати адаптивне шкідливе програмне забезпечення, призначене для обходу поточних заходів безпеки, визнані як такі, що знаменують собою складну еволюцію в наступальних можливостях. Для боротьби з цими загрозами пропонується комплексна стратегія безпеки, що охоплює концепції нульової довіри, глибинної оборони і принципу посилення безпеки. Відповідно до моделі нульової довіри, необхідна сувора перевірка всіх запитів на доступ, незалежно від їх походження. Глибокий захист характеризується впровадженням декількох рівнів безпеки в IT-середовищі, в той час як принцип посилення безпеки зосереджується на зміцненні систем і додатків для зменшення вразливостей і кількості векторів атак.

Ключові слова: APT, EDR, антивірус, виявлення загроз, кібербезпека, обхід захисту, LOLBAS, Sysmon.

Opanovych M. Y.

ANALYSIS OF THE WAYS OF BYPASSING THE TOOLS FOR THE THREATS PREVENTION AND RESEARCHING COUNTERING METHODS

This study examines the evasion techniques employed by Advanced Persistent Threat (APT) groups that bypass traditional defenses such as antivirus software and endpoint detection and response (EDR) systems. The focus is on techniques such as obfuscation, encryption, and exploitation of security tool vulnerabilities, which have been identified as significant challenges to effective threat detection. It examines how attackers use Living Off The Land Binaries and Scripts (LOLBAS) to manipulate built-in system tools for stealth attacks, seamlessly combining malicious activity with legitimate processes to avoid detection. The role of security information and event management (SIEM) systems in improving detection capabilities is also discussed. SIEM systems stand out for their ability to collect and analyze security data across a network, providing a holistic view that helps to identify potential threats and breaches early, thereby strengthening an organization's security posture. Vulnerabilities in EDRs and antivirus solutions themselves are analyzed, showing how they can be exploited to disable or bypass these critical protections, leaving systems vulnerable. In addition, the development of malware generators using machine learning techniques is discussed. These tools, which are capable of creating adaptive malware designed to bypass current security measures, are recognized as marking a complex evolution in offensive capabilities. To combat these threats, a comprehensive security strategy is proposed that incorporates the concepts of zero trust, defense in depth, and the principle of security enhancement. According to the zero-trust model, all access requests, regardless of their origin, must be strictly verified. Defense-in-depth is characterized by the implementation of multiple layers of security in the IT environment, while the principle of hardening focuses on strengthening systems and applications to reduce vulnerabilities and the number of attack vectors.

Keywords: APT, EDR, antivirus, threat detection, cybersecurity, defense evasion, LOLBAS, Sysmon.

Стаття надійшла до редакції 17.05.2024 р.
Прийнято до друку 12.06.2024 р.