Секція: Технічні науки

Сімоненко Валерій Павлович
доктор технічних наук, професор,
професор кафедри обчислювальної техніки
Національний технічний університет України
"Київський політехнічний інститут імені Ігоря Сікорського"
м. Київ, Україна

Вернер Анна Ігорівна
студент-магістр кафедри обчислювальної техніки
Національного технічного університету України
"Київський політехнічний інститут імені Ігоря Сікорського"
м. Київ, Україна

ОБ'ЄКТНО-ОРІЄНТОВАНИЙ СПОСІБ ПІДВИЩЕННЯ БЕЗПЕКИ ОБЧИСЛЮВАЛЬНИХ ВУЗЛІВ НА ОСНОВІ ПІДСИСТЕМИ АУДИТУ

З активним розвитком мережевих технологій та сервісів спостерігається невпинне експоненційне зростання загальної кількості шкідливого програмного забезпечення, дії якого спрямовані на виведення з ладу обчислювальних вузлів. Водночас, відповідно до відкритих статистичних даних [4], крива кількості нових видів зловмисних програмних застосунків має спадний характер. Розробники засобів інформаційної безпеки звітують [1], що більше семидесяти п'яти відсотків обчислювальних вузлів, що були інфіковані, мали встановлені найновіші оновлення систем безпеки. Це свідчить про недосконаліть існуючих засобів захисту, що спричинює неможливість вчасного виявлення та запобігання існуючим загрозам. У зв'язку з цим досить гостро постає питання необхідності посилення існуючих засобів безпеки

інформаційних систем, шляхом винайдення нових способів ідентифікації загроз.

В сучасних інформаційних системах для забезпечення цілісності даних найчастіше застосовують апаратні та програмні засоби захисту інформації (ПЗЗІ). Останні є найбільш поширеними завдяки гнучкості у використанні та можливості конфігурації під різні типи архітектур без необхідності виконання суттєвих змін у їх структурі.

Основні ПЗЗІ представлені у вигляді таких засобів: антивірусного програмного забезпечення, ідентифікації та автентифікації користувачів, протоколювання, керування доступом та засобів аудиту. Застосування таких спеціалізованих ПЗЗІ передбачає виявлення загроз шляхом використання ряду методів аналізу коду шкідливих застосунків. Серед них слід виокремити два основні підходи: статичний та динамічний аналіз.

За статичного аналізу не відбувається виконання шкідливого програмного забезпечення. Свідчення щодо небезпечності застосунків отримуються шляхом проведення аналізу метаданих файлів, сигнатур рядків, n-грам, викликів бібліотек, тощо. Даний підхід є досить ефективним завдяки своїй швидкості, проте він не дозволяє якісно ідентифікувати загрози, оскільки не є стійким до обфускації, що власне і спричинило необхідність появи динамічного аналізу.

Динамічний аналіз передбачає проведення аналізу програмного забезпечення безпосередньо під час його виконання в системі, шляхом спостереження за взаємодією зібраного зразка шкідливого застосунку з інформаційною системою. В порівнянні зі статичним аналізом такий підхід є більш ефективним та дозволяє виявляти значну частину шкідливого програмного забезпечення. Однак, він також містить недоліки, одним з яких є великі часові та ресурсні затрати, оскільки для спостереження за зразком

шкідливого застосунку створюється спеціальне ізольоване віртуальне середовище. З метою отримання кращих результатів розробниками ПЗЗІ виконується комбінування переваг приведених підходів – гібридний аналіз.

Відповідно до останніх досліджень в області розробки ПЗЗІ [5; 6; 2] все більше уваги при вдосконаленні методів аналізу приділяється застосуванню машинного навчання (англ. machine learning - ML). Застосування ML дозволяє значно знизити відсоток хибних висновків щодо безпечності програмного забезпечення, а також доводить [5] можливість запобігання атакам нульового дня.

Для підвищення безпеки обчислювальних вузлів пропонується вдосконалити методи аналізу у нижчезазначений спосіб, що базується на застосуванні підсистеми аудиту операційних систем.

В термінології операційних систем під аудитом розуміють фіксацію спеціалізованими ПЗЗІ подій, що відбуваються у системі з метою подальшого їх аналізу. Збір даних аудиту відбувається на окремих хост-системах з журналів, що утворюються в процесі роботи. Отримані дані містять велику кількість корисної інформації, що дозволяє не тільки детально відслідковувати події, що відбуваються у системі, але і відновлювати хід подій у разі необхідності вивчення специфіки атак.

Безпосередньо, підсистема аудиту має лише діагностичні властивості, проте у зв'язку з інтеграцією підсистеми до модулів ядра та її роботи на найнижчому рівні з можливістю перехоплення системних викликів ядра пропонується її застосування як базової компоненти для проведення гібридного аналізу.

Підсистема аудиту контролює три основні типи подій: системні виклики, дозволяючи переглядати їх контекстну інформацію; події доступу до файлів у

якості альтернативного способу моніторингу активності доступу до файлів [3]; попередньо налаштовані для перехоплення події.

Отримуючи системні виклики з простору користувача компонента ядра підсистеми аудиту виконує фільтрацію. Після проходження через один з вхідних фільтрів виклик направляється для проходження через фільтр виключення, що на базі конфігурації правил аудиту направляє його для подальшої обробки безпосередньо демону аудиту.

Розвинена підсистема фільтрів системних викликів дозволяє суцільно контролювати поведінку будь-якого процесу в системі та, у разі співпадіння з шаблоном поведінки шкідливого програмного забезпечення, визначеного за допомогою алгоритмів машинного навчання на базі поведінкової моделі, запобігати подальшому виконанню процесу в системі.

Наявність надвеликої кількості шкідливого програмного забезпечення призводить до щоденних збитків не тільки великих корпорацій, а й усього суспільства в цілому. Через постійне вдосконалення ШПЗ виникає необхідність ускладнення алгоритмів та методів аналізу з метою захисту інформаційних систем не тільки від існуючих, а й від загроз нульового дня. Запропонований спосіб може бути застосований для запобігання виконанню і поширення зловмисних застосунків на базі будь-якої операційної системи, оскільки підсистема аудиту є нативним діагностичним засобом. Основною перевагою розробленого способу є не тільки можливість визначення загроз безпеці інформаційної системи, але і також можливість запобігання її виконанню та поширенню завдяки наявності найвищих привілеїв для роботи у системі. Для досягнення найкращих результатів роботи засобу захисту мають застосовуватися тільки ключові з точки зору детермінації загрози системні виклики, інакше це може призвести до значного зниження продуктивності роботи інформаційної системи загалом.

Література

1. Businesses Impacted by Repeated Ransomware Attacks and Failing to Close the Gap on Exploits, According to Sophos Global Survey // Sophos. URL: https://www.sophos.com/en-us/press-office/press-releases/2018/ 01/businesses-impacted-by-repeated-ransomware-attacks-according-to-sophos-global-survey.aspx (дата звернення: 20.03.2020).
2. Stokes J. W., Wang D., Marinescu M., Marino M. and Bussone B. Attack and Defense of Dynamic Analysis-Based, Adversarial Neural Malware Detection Models // MILCOM 2018 - 2018 IEEE Military Communications Conference (MILCOM), Los Angeles, CA. 2018. PP. 1-8.
3. Shortridge Kelly What is the Linux Auditing System // Capsule 8. January 7. 2020. URL: https://capsule8.com/blog/auditd-what-is-the-linux-auditing-system/ (дата звернення: 20.03.2020).
4. Lapowsky Malware last 10 years // AV-TEST. URL: https://www.av-test.org/en/statistics/malware/ (дата звернення: 20.03.2020).
5. Liu, Xinbo & Lin, Yaping & Li, He & Zhang, Jiliang. A Novel Method for Malware Detection on ML-based Visualization Technique. Computers & Security. 89. 101682. 10.1016/j.cose.2019.101682. (2019)
6. Agrawal R., Stokes J. W., Marinescu M. and Selvaraj K. Neural Sequential Malware Detection with Parameters // 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Calgary, AB, 2018. PP. 2656-2660.