В. Д. Бойко[1]
В. М. Слатвінська[1]

КОНТРОЛЬ ТА УПРАВЛІННЯ РОЗПОДІЛЕНИМИ СИСТЕМАМИ ВИЯВЛЕННЯ ВТОРГНЕНЬ З ВИКОРИСТАННЯМ ЛОГІВ У НЕСПРИЯТЛИВИХ УМОВАХ

[1] Національний університет «Одеська юридична академія»;

Розглянуто проблеми управління та контролю розподіленими системами виявлення вторгнень (SIEM) у несприятливих умовах (блекаути, кібератаки, бойові дії). Показано роль SIEM у зменшенні проміжку часу між вторгненням у систему та зараженням її malware та виявленням факту вторгнення. Перераховано проблеми у існуючих SIEM, наведено класифікацію методологій виявлення вторгнень, вказано на суттєву роль, яку відіграють системні журнали (логи) для виявлення та усунення кіберзагроз. В роботі описані різні можливості контролю та верифікації системних журналів: система, заснована на зв'язаних списках, системи, засновані на деревах хешів (Merkle Tree і Verkle Tree). Поставлено завдання створення системи, що маловразлива до повного або часткового тимчасового порушення зв'язності мережної інфраструктури, яке може відбутися в результаті несприятливих впливів. Запропоновано асинхронну систему контролю та управління системою верифікації та аудиту логів, яка базується на асинхронних процедурах відправлення "квитанцій" (push/pull) вузлами різного рівня мережевої інфраструктури та зберігає працездатність в умовах часткової втрати функціональності мережевої інфраструктури. Розгортання такої системи з одного боку дозволяє зберегти і підтримувати цілісність логів за умов несприятливих впливів, з другого боку завдяки асинхронності дозволяє рівномірно розподіляти навантаження всередині мережевий інфраструктури не перевантажуючи вузли аудиту інформації великим обсягом обчислень. Впровадження та розгортання системи на запропонованих принципах вимагає менше ресурсів у порівнянні з існуючими SIEM системами і може бути здійснено, як самостійно, так і з інтеграцією у вже розгорнуту SIEM якщо вона є. Запропоновані заходи дозволять значно збільшити безпеку, оперативність реагування на інциденти та можливість відновлення існуючих інформаційних мереж.
Ключові слова: SIEM, Merkle Tree, Verkle Tree, руткіти, виявлення зловмисного програмного забезпечення, зловмисне програмне забезпечення, кібератака, журнал, перевірка, хеш, IDS, IPS, мережева інфраструктура.

Вступ

Основним джерелом мережевих атак в даний час є ботнети - мережі заражених malware комп’ютерів, що мають доступ до інтернету. Ботнети є серйозною проблемою - досить сказати, що більшість комп’ютерних атак здійснюється за їх допомогою [1], [2]. Будь-яка сучасна інформаційно-комунікаційна мережа (information communication network - ICN) з доступом до глобального інфопростору практично безперервно піддається атакам різного роду і характеру [3]. Визначальним фактором успішного функціонування сучасних ботнетів є часовий інтервал між вторгненням та зараженням комп’ютера та виявленням того, що це сталося. Цей інтервал визначає ступінь шкідливості malware – чим він довший, тим більше атак встигне здійснити заражена машина, до того, як це виявиться і будуть застосовані заходи у вигляді налаштування фаєрволу, антивірусного сканування тощо. аж до переустановки операційної системи. Однак, виявлення вторгнень у систему в сучасних умовах все ще недостатньо швидке. У 2015 році середній час виявлення факту успішної кібератаки складав від 50 до 70 днів [4]. Згідно з звітами IBM за 2020 рік [5], на ідентифікацію витоків витрачалося в середньому 280 днів, Madiant Security Effectiveness Report 2020 [6] вказує, що 53% кібератак закінчується працювання систем безпеки.
Проблему намагаються вирішити системним підходом - використанням технології SIEM (Security information and event management), яка є синтезом існуючих раніше систем SEM (Security Event Management) і SIM (Security information management) [7], [8]. Однак, на практиці впровадження складних, витратних за ресурсами та потребують кваліфікації керуючого персоналу систем захисту, часто виявляється утрудненим для розробників та адміністраторів ICN середньої та нижчої ланки. При цьому робота з ключовою для виявлення та управління кібербезпекою інформацією донедавна будувалася виходячи з парадигми того, що вся інформація в ICN буде доступна в будь-який момент часу, що функціонування ICN мінімально схильне до збоїв і без урахування зростання складності

ICN. Ці питання докладно розглядалися у роботі [9] що вийшла ще до початку повномасштабного вторгнення, блекаутів та кількох глобальних відключень важливих структурних складових ICN (“Київстар”, facebook, microsoft тощо). Наступні події (блекаути, втрата функціональності систем через бойові дії, нещодавнє відключення мобільного зв’язку “Київстар”) підтвердили актуальність викладених у статті положень та були додатково розглянуті у [10]. Сучасна ICN існує в умовах все зростаючої щільності несприятливих впливів і факторів, що вражають [3]. При цьому структура самої ICN забезпечує досить великий ступінь автономності, що з одного боку дозволяє швидше відновлювати систему після збоїв або вимушеного простою (наприклад, в результаті блекауту), а з іншого ускладнює управління інформацією та ключовими подіями для кібербезпеки ICN. Наприклад, якщо блекаут викликав втрату зв’язності окремих частин ICN, це може призвести до збоїв у роботі SIEM, яка не розрахована на несинхронну роботу з розподіленою системою.
У цій роботі пропонується структура та схема взаємодії, контролю та управління розподіленими системами виявлення вторгнень з використанням логів, що використовує асинхронний принцип доступу до даних.
Метою роботи є розробка структури та схеми взаємодії, контролю та управління розподіленими системами виявлення вторгнень з використанням логів, що використовує асинхронний принцип доступу до даних.

Системи виявлення вторгнень з використанням логів

Існуючі технології виявлення вторгнень у сучасних інформаційно-комунікаційних мережах у рамках технології SIEM можна розділити на кілька основних категорій, які використовують різні методи виявлення. У найзагальнішому вигляді до них можна віднести:

- аналіз мережевого трафіку,
- аналіз вмісту пам’яті та жорстких накопичувачів,
- аналіз системних журналів.

Зазначимо, що системи виявлення вторгнень найчастіше будуються за гібридною моделлю, використовуючи кілька різних технологій.
Якщо розглядати ці схеми з точки зору виявлення джерела malware, то перша з цих технологій потребує серйозної підготовки та доступу до всіх ланок ієрархії ICN, через які проходить атака. При цьому аналіз мережного трафіку серйозно утруднений через фрагментацію ICN, яка у свою чергу пояснюється відставанням у впровадженні технологій IPv6, через що сучасний інфопростір є ієрархією підмереж з “сірими” IP-адресами, в якій існує багато рівнів вкладеності і часто - підключення, що динамічно змінюються (найочевидніший приклад - смартфон, що перемикається між WiFi-мережами). Все це сильно ускладнює детектування та виявлення джерела атаки шляхом аналізу мережевого трафіку. Додатковими факторами, що погіршують, є все зростаючий обсяг переданих даних, низька грамотність як технічного персоналу, так і кінцевих користувачів, затримки з оновленням ОС і так далі.
Друга категорія технологій ґрунтується на аналізі вмісту пам’яті та жорстких накопичувачів. Найчастіше цю роль виконує антивірусний software різної спрямованості та методології (пошук по сигнатурах, пошук по евристикам). Цей метод також є досить повільним, особливо з урахуванням зростання обсягів інформації, що зберігається і передається, і обсягів самого software.
Аналіз системних журналів (логів) є одним з найбільш оперативних і доступних методів виявлення вторгнення malware. Логи є цінним матеріалом для аналізу характеру атак, поведінки атакуючої сторони та інформування у разі порушення захисту [2]. При цьому аналіз логів як засіб боротьби з malware поступово перетворюється на мету для проектувальників malware, які чудово обізнані про ці технології і намагаються їх обійти. Відповідно до класифікації MITRE [11], атаки, пов’язані з модифікацією логів, відносяться до метакласу з ідентифікатором T1070. Зокрема, туди входять техніки T1070.001 (Очищення журналів подій Windows) і техніка T1070.002 (Очищення системних журналів Linux або Mac). Чотири роки тому в доповіді [12] ми припустили, поява malware, яка маскуватиме факт вторгнення в систему шляхом заміни записів у системних журналах і запропонували технологію захисту, засновану на зв’язкових списках). На той момент у базі даних MITRE було зареєстровано 2 різновиди malware, що використовують техніку T1070.001 (Очищення журналів подій Windows) та 1 різновид, що використовують техніку T1070.002 (Очищення системних журналів Linux або Mac). У 2022-му році ця цифра склала 17 типів для T1070.001 і 3 типу для T1070.002. На даний момент зареєстровано 28 типів, що використовують техніку T1070.001 та 4 різновиди, що використовують техніку T1070.002. І це тільки початок - зараз

основною технологією маскування malware є просте видалення записів із системного журналу. Проте, з розвитком систем виявлення очікується появи malware, яке навчиться видаляти чи підробляти записи у системному журналі вибірково те щоб не видавати факт вторгнення відсутністю записів взагалі.

Деякий час захист системних журналів не був пріоритетом при розробці систем безпеки - особливо серед системних адміністраторів середньої та нижчої ланки ICN, які часто не володіють ресурсами та знаннями для організації розвиненої інфраструктури безпеки, а іноді й поєднують роботу із забезпечення кібербезпеки з будь-якими іншими службовими обов'язками. . Додатковою перешкодою була відсутність єдиних стандартів, тактики та політики захисту системних журналів. Впровадження комплексної інтегрованої системи, як правило, представляє складність і часто не є пріоритетом при розробці та експлуатації ICN. Відповідно, найчастіше експлуатація ICN поза великими корпораціями зводиться до набору hardening практик і "корисних порад", які далеко не завжди виконуються системними адміністраторами. Це вимагає організації інфраструктури, яка з одного боку забезпечувала б збирання, аналіз та зберігання системних журналів, з іншого боку дозволяла запобігти фальсифікації показань системних журналів і - з третьої сторони не була б ресурсозатратною як у розгортанні, так і в експлуатації. Останнє важливо, оскільки на будь-яких системах рівня нижче корпоративного, складність та витратність в експлуатації часто призводить до того, що рішення не запроваджується взагалі.

У роботах [13] було запропоновано таке рішення, що ґрунтувалося на технології зв'язкових списків. У цьому рішенні додатковий контроль за цілісністю системних журналів зводився до того, що в системний журнал через деякі проміжки часу додавали спеціальне повідомлення, що представляє собою хеш-суму з повідомлень, що потрапили в системний журнал, включаючи передостаннє контрольне повідомлення. Таким чином, журнал з одного боку верифікувався наявністю ланцюжка контрольних сум, з іншого така схема не вимагала великих обчислювальних витрат, інфраструктури та гнучко налаштовувалась, оскільки інтервал можна було задавати як тимчасовим, так і за кількістю повідомлень у системному журналі. Така система добре показала себе на практиці на серверних системах середнього та низького рівня, проте, у зв'язку з широким поширенням хмарних систем, все гострішою стає необхідність в аналогічних заходах захисту, але вже дозволяють контролювати розподілені системи - де реалізація простої схеми зв'язкових списків не є можливою технічно.

У роботі [14] було запропоновано розвиток системи шляхом організації перехресних перевірок цілісності на базі технології дерев хешів, найвідомішою реалізацією яких є Merkle Tree. Використання Merkle tree виглядає наступним чином - на основі хешів вихідних записів вибудовується дерево хешів, в якому хеш верифікуються попарно, утворюючи собою ієрархію - дерево хеш, яке зі зростанням бази даних росте пропорційно O(log N). Використання даної схеми роботи дозволить з одного боку зберегти всі переваги системи, заснованої на хеш-сумах (невимогливість до ресурсів, відсутність необхідності у побудові та обслуговуванні складної інфраструктури), з іншого, шляхом незначного збільшення надмірності дозволило отримати такі переваги, як можливість роботи з розподіленими ресурсами, що покращить загальну безпеку системи.

Подальшим розвитком ідеї стало використання Verkle Trees[15], в якому хеш-дерево скорочено до кількох рівнів, що серйозно впливає на швидкість та простоту логів інфраструктури перевірки. Крім інших переваг, Verkle Tree дозволяє гнучко використовувати алгоритми різної ефективності та роботи — від звичайного хешування до схеми поліноміального зобов'язання KZG[16]. В [14] використовувалася криптографічна хеш-функція SHA-256, щоб спростити реалізацію Verkle Tree у різних контекстах використання завдяки наявності інструментальної бази у вигляді бібліотек, що добре зарекомендували себе.

Синхронність у несприятливих умовах функціонування ICN

Як було сказано вище, централізація збору системних журналів в даний час додатково утруднюється через все збоїв у мережі (блекаути, кібератаки, бойові дії і т.д.), що частішають, при яких мережа може частково або повністю втрачати зв'язність і розпадатися на незалежні сегменти з подальшим відновленням. Існуючі системи SIEM передбачають можливість практично миттєвого доступу до ключової інформації обсягом всього підконтрольного сегменту ICN. Короткочасна або тривала втрата взаємодії з підсистемами ICN може призвести до розладу та плутанини у існуючих системах контролю цілісності балок.

Отже, існує необхідність в інфраструктурі збору та верифікації системних журналів у рамках розподіленої ICN, яка не вимагала б складної інфраструктури та витрат на впровадження та дозволяла б функціонувати ICN в умовах часткової доступності її окремих "острівів" - напівавтономна або цілком автономна на деякі періоди часу. Існуючі рішення зазвичай зводяться до резервування та дублікації в тому чи іншому вигляді системних журналів по всій системі централізованим або централізованим чином [17], [18], що вимагає великих вкладень (наприклад, фактично подвоюється або потроюється необхідна ємність сховищ інформації) не забезпечує оперативного виявлення порушень цілісності системних журналів, оскільки у межах такої системи її можна виявити лише шляхом послідовної звірки записів оригіналу та копії. При цьому слід враховувати, що кібератаці або несприятливому впливу може бути схильний до будь-якого з рівнів системи.

У роботі пропонується схема асинхронної взаємодії систем верифікації логів. В рамках цієї системи низові ланки ICN працюють згідно зі схемою, наведеною в [13], де кожен журнал верифікується шляхом технології зв'язкових списків. Система розширюється за рахунок додавання до кожної контрольної суми метаданих: дати та внутрішнього часу обчислення дайджесту, кількості завірених повідомлень, а також ідентифікатора вузла ICN. Це є природним розширенням системи ведення логів і часто не вимагає перегляду схеми, оскільки такі метадані додаються до запису автоматично самим програмним забезпеченням для ведення логів. Однак, замість регулярного збору інформації по системі, шляхом запитів від вищих вузлів ієрархії ICN, після кожного обчислення контрольної суми, низовий елемент виконує в термінології запропонованої системи дію PUSH, самостійно відправляючи запит "квитанцію" на рівень вище. Це може бути реалізовано як з підтвердженням отримання квитанції вищим вузлом, так і без неї. У свою чергу, кожен вищий вузол збирає отримані "квитанції". Після досягнення певної кількості квитанцій, він аналогічним чином обчислює свій дайджест, додає до нього метадані та надсилає отриману квитанцію ще на один рівень вище. Таким чином вибудовується ієрархічна система, подібна до Merkle Tree, яка дозволяє працювати асинхронно і без безперервного доступу до всіх мережевих сегментів ICN. У разі збою зв'язку повідомлення з конкретного вузла будуть просто надіслані пізніше.

При необхідності аудиту системи, вищестоящий вузол відправляється запит PULL, який нижчестоящі вузли дублюють ієрархії вниз. Після чого вузли, до яких прийшов PULL-запит знову виконують процедуру PUSH за всіма запитаними даними, а вузол, що надіслав запит PULL звіряє отримані повторно дайджести з централізовано. Така схема дозволяє з одного боку працювати в умовах тимчасових втрат зв'язку та зв'язності, з іншого боку за рахунок асинхронності дозволяє рівномірно розподіляти навантаження в ICN не перевантажуючи вузли-збирачі інформації великим обсягом обчислень.

Висновки

Запропонована система контролю та управління розподіленими системами виявлення вторгнень з використанням логів у несприятливих умовах використовує асинхронний режим роботи та дозволяє з одного боку зберігати та підтримувати цілісність логів в умовах несприятливих впливів та вражаючих факторів, з іншого боку за рахунок асинхронності дозволяє рівномірно розподіляти навантаження у ICN не перевантажуючи вузли-збирачі інформації великим обсягом обчислень. Впровадження та розгортання системи на запропонованих принципах вимагає менше ресурсів у порівнянні з існуючими SIEM системами і може бути здійснено, як самостійно, так і з інтеграцією у вже розгорнуту SIEM якщо вона є. Ці заходи дозволять значно збільшити безпеку, оперативність реагування на інциденти та можливість відновлення існуючих ICN.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

[1] Silva, S. S. C., Silva, R. M. P., Pinto, R. C. G., and Salles, R. M. (2013). Botnets: A survey. *Computer Networks*, *57*(2), 378–403. https://doi.org/10.1016/j.comnet.2012.07.021

[2] Barford, P., and Yegneswaran, V. (2007). An inside look at botnets. In *Advances in information security* (pp. 171–191). Springer US. https://doi.org/10.1007/978-0-387-44599-1_8

[3] Bi, W., MacAskill, K., and Schooling, J. (2023). Old wine in new bottles? Understanding infrastructure resilience: Foundations, assessment, and limitations. *Transportation Research Part D: Transport and Environment*, *120*, 103–793. https://doi.org/10.1016/j.trd.2023.103793

[4] Johnson, J. (2015). *Average number of days to resolve a cyber attack on companies in the united states as of august 2015, by attack type*. https://www.statista.com/statistics/193463/average-days-to-resolve-a-cyber-attack-in-us-companies-by-attack/

[5] IBM. (2021, March). *Cost of a data breach report 2020*. https://www.ibm.com/security/digital-assets/cost-data-breach-report/#/ru

[6] Mandiant. (2020). *Mandiant security effectiveness report* (pp. 1–22). FireEye. https://www.fireeye.com/current-threats/annual-threat-report/security-effectiveness-report.html
[7] Cinque, M., Cotroneo, D., and Pecchia, A. (2018). Challenges and directions in security information and event management (SIEM). *2018 IEEE International Symposium on Software Reliability Engineering Workshops (ISSREW)*, 95–99. https://doi.org/10.1109/issrew.2018.00-24
[8] González-Granadillo, G., González-Zarzosa, S., and Diaz, R. (2021). Security information and event management (SIEM): Analysis, trends, and usage in critical infrastructures. *Sensors*, *21*(14), 1–28. https://doi.org/10.3390/s21144759
[9] Boyko, V., Vasilenko, M., and Slatvinska, V. (2021). Survivability and sustainability of smart city information system components. *Municipal Economy of Cities*, *6*(166), 20–27. https://doi.org/10.33042/2522-1809-2021-6-166-20-27
[10] Бойко, В., Василенко, М., Слатвінська, В. (2024). Моделювання живучості та відновлення інформаційно-комунікаційних мереж в умовах дії кіберзагроз. *Інформаційні Технології Та Суспільство*, *1 (12)*, 13–19. https://doi.org/10.32689/maup.it.2024.1.2
[11] *Indicator Removal on Host: Clear Linux or Mac System Logs, Sub-technique T1070.002 - Enterprise MITRE ATTandCK*. (2022, May). https://attack.mitre.org/techniques/T1070/002
[12] Бойко, В., Василенко, М. (2020). Система виявлення вторгненнь з використанням технології зв'язаних списків. *Матеріали XV Міжнародної Конференції" Контроль і Управління в Складних Системах (КУСС-2020)", м. Вінниця, 8-10 Жовтня 2020 р.*, 265–266.http://ir.lib.vntu.edu.ua//handle/123456789/30577
[13] Boyko, V., Vasilenko, M., and Slatvinska, V. (2022). Linked list systems for system logs protection from cyberattacks. *"Information Technologies in Education, Science and Technology" (ITEST-2022) June 23-25, 2022 Cherkasy*, 81–82.https://er.chdtu.edu.ua/bitstream/ChSTU/4121/1/Збірник_тез_ІТОНТ-2022_макет_23_06.pdf#page=81
[14] Boyko, V., Vasilenko, M., and Slatvinska, V. (n.d.). Linked list systems for system logs protection from cyberattacks. In E. Faure, O. Danchenko, M. Bondarenko, Y. Tryus, C. Bazilo, and G. Zaspa (Eds.), *Information technology for education, science, and technics* (pp. 224–234). Springer Nature Switzerland. https://doi.org/10.1007/978-3-031-35467-0_15
[15] Kuszmaul, J. (2019). *Verkle trees*. https://api.semanticscholar.org/CorpusID:218475793
[16] Tas, E. N., and Boneh, D. (2023). *Vector commitments with efficient updates*. 46. https://doi.org/10.48550/ARXIV.2307.04085
[17] Aßmuth, A., Duncan, R., Liebl, S., and Söllner, M. (2021). A secure and privacy-friendly logging scheme. In B. Duncan, Y. W. Lee, and M. Popescu (Eds.), *Cloud computing 2021*. https://www.iaria.org/conferences2021/CLOUDCOMPUTING21.html
[18] Hangxia, Z., Peng, Z., and Yong, Y. (2010). Web log system of automatic backup and remote analysis. *2010 International Conference on Computer Application and System Modeling (ICCASM 2010)*, 469–472. https://doi.org/10.1109/iccasm.2010.5620567

Бойко Віктор Дмитрович — канд. техн. наук, доцент, доцент кафедри кібербезпеки, e-mail: boyko-work@ukr.net;
Валерія Миколаївна Слатвінська — доктор філософії в галузі права, асистент кафедри кібербезпеки. Національний університет «Одеська юридична академія», Одеса;

V. D. Boiko[1]
V. M. Slatvinska[1]

Control and management of distributed intrusion detection systems using logs in adverse conditions

[1]National University "Odesa Law Academy"

This paper proposes an asynchronous system of control and management of the log verification and audit system, which is based on asynchronous procedures for sending "receipts" (push/pull) by nodes of different levels of the network infrastructure and maintains operability in conditions of partial loss of functionality of the network infrastructure. The problems of management and control of distributed intrusion detection systems (SIEM) in adverse conditions (blackouts, cyberattacks, etc.) are considered. Shown the role of SIEM in reducing the time interval between intrusion into a system and its malware infection and detection of the fact of intrusion. The problems of existing SIEMs are listed, the classification of intrusion detection methodologies is given, and the essential role played by system logs (logs) in detecting and eliminating cyber threats is indicated. The work describes various possibilities of control and verification of system logs: a system based on linked lists, systems based on hash trees (Merkle Tree and Verkle Tree). The task of creating a system that is insensitive to full or partial temporary disruption of the connectivity of the network infrastructure, which may occur as a result of adverse influences, is set. The deployment of proposed system, on the one hand, allows preserving and maintaining integrity of logs under conditions of adverse effects, on the other hand, due to asynchrony, it allows you to evenly distribute the load within the network infrastructure without overloading the information audit nodes with a large amount of calculations. The implementation and deployment of the system based on the proposed principles requires fewer resources compared to existing SIEM systems and can be implemented both independently and with integration into an already deployed SIEM, if any. The proposed measures will make it possible to significantly increase security, prompt response to incidents, and the ability to restore existing information networks.
Keywords*: SIEM, Merkle Tree, Verkle Tree, rootkits, malware detection, malware, cyberattack, log, blackout, verification, hash, IDS, IPS, network infrastructure.*

Boiko Viktor Dmytrovych — Cand. Sc. (Eng.), Assistant Professor, Assistant Professor of the Chair of Cybersecurity, e-mail: boyko-work@ukr.net;

Slatvinska Valeriia Mykolaivna — Doctor of Philosophy (PhD) in Law, Assistant Lecturer of the Chair of Cybersecurity;