Міністерство освіти і науки України
Київський національний університет імені Тараса Шевченка

Факультет інформаційних технологій
Кафедра кібербезпеки та захисту інформації

ДОПУСТИТИ ДО ЗАХИСТУ:
В.о. завідувача кафедри
кібербезпеки та захисту інформації
_____ Іван ПАРХОМЕНКО
«____» червня 2023 р.

ПОЯСНЮВАЛЬНА ЗАПИСКА

кваліфікаційної роботи

галузь знань 12 Інформаційні технології
(шифр і назва галузі знань)

спеціальність 125 Кібербезпека
(код і назва спеціальності)

освітній ступень бакалавр

освітня програма Кібербезпека
(назва освітньо-професійної програми)

на тему: «Засоби виявлення індикаторів компрометації на основі Threat Intelligence»

Виконавець: студентка IV курсу, групи КБ-42

_____ **Юлія ВЛАСЮК**
(підпис) (ім'я, прізвище)

	Ім'я, прізвище	**Підпис**
Керівник	Іван ПАРХОМЕНКО	

Нормоконтроль	Лариса МИРУТЕНКО	

Київ 2023

Міністерство освіти і науки України
Київський національний університет імені Тараса Шевченка

Факультет інформаційних технологій
Кафедра кібербезпеки та захисту інформації

ЗАТВЕРДЖЕНО:
В.о. завідувача кафедри
кібербезпеки та захисту інформації
_____ Сергій ТОЛЮПА
«24» жовтня 2022 р.

ЗАВДАННЯ
на виконання кваліфікаційної роботи

спеціальності 125 Кібербезпека
(код і назва спеціальності)

освітньої програми Кібербезпека
(назва освітньо-професійної програми)

Студентці **КБ-42** (група) **Власюк Юлії Юріївни** (прізвище ім'я по батькові)

Тема кваліфікаційної роботи Засоби виявлення індикаторів компрометації на основі Threat Intelligence

1. ПІДСТАВИ ДЛЯ ПРОВЕДЕННЯ РОБОТИ

Тема кваліфікаційної роботи затверджена на засіданні кафедри кібербезпеки та захисту інформації протокол №3 від 20.10.2022 р.

2. ВИХІДНІ ДАНІ ДЛЯ ПРОВЕДЕННЯ РОБІТ

Існуючі методики, що описують варіації зловмисної поведінки в інформаційних системах, публічні ресурси для поширення індикаторів компрометації.

3. ЗМІСТ РОЗРАХУНКОВО-ПОЯСНЮВАЛЬНОЇ ЗАПИСКИ

Необхідно ознайомитися з існуючими тактиками та техніками, що описують варіації зловмисної поведінки з боку атакуючого для розробки методик виявлення даної зловмисної поведінки.

4. ВИМОГИ ДО РЕЗУЛЬТАТІВ ВИКОНАННЯ РОБОТИ

Практична цінність Розроблений алгоритм дій для пошуку та виявлення індикаторів компрометації використовуючи деякі засоби і способи розвідки загроз.

5. ДАТА ВИДАЧІ ЗАВДАННЯ

Дата видачі завдання: 24 жовтня 2022 року

Завдання видав _____ (підпис) Іван ПАРХОМЕНКО (ім'я, прізвище)

Завдання прийняла до виконання _____ (підпис) Юлія ВЛАСЮК (ім'я, прізвище)

КАЛЕНДАРНИЙ ПЛАН

№ п/п	Найменування етапів робіт	Строки виконання робіт (початок-кінець)	Відмітка про виконання
1	Уточнення постановки задачі	24.10.2022 – 30.10.2022	*виконано*
2	Аналіз відкритих джерел	31.10.2022 – 13.11.2022	*виконано*
3	Обґрунтування вибору рішення	14.11.2022 – 27.11.2022	*виконано*
4	Концепція розвідки загроз та індикаторів компрометації	28.11.2022 – 31.12.2022	*виконано*
5	Аналіз проблем виявлення зловмисної поведінки в інформаційних системах	01.01.2023 – 15.01.2023	*виконано*
6	Дослідження різноманітних варіантів проведення розвідки загроз та збору індикаторів компрометації	16.01.2023 – 29.01.2023	*виконано*
7	Вироблення алгоритмів дій для проведення розвідки загроз в інформаційних системах	30.01.2023 – 19.03.2023	*виконано*
8	Розробка та розгортання тестової інфраструктури для демонстрації одного з варіантів системи для проведення розвідки загроз	20.03.2023 - 07.05.2023	
9	Оформлення пояснювальної записки	08.05.2023 – 28.05.2023	*виконано*
10	Підготовка до захисту кваліфікаційної роботи	29.05.2023 – 12.06.2023	*виконано*

Завдання видав _____ (підпис) Іван ПАРХОМЕНКО (ім'я, прізвище)

Завдання прийняв до виконання _____ (підпис) Юлія ВЛАСЮК (ім'я, прізвище)

Термін подання кваліфікаційної роботи до ЕК 12 червня 2023 року

РЕФЕРАТ

Кваліфікаційна робота складається зі вступу, трьох розділів, загальних висновків, списку використаних джерел, додатків, має 82 сторінки основного тексту, 65 рисунків. Список використаних джерел містить 27 найменування і займає 2 сторінки.

Метою роботи є дослідження алгоритму дій для пошуку індикаторів компрометації інформаційних систем на основі відкритих джерел.

Методи дослідження кваліфікаційної роботи:

- аналіз відкритих джерел;
- порівняння та аналіз існуючих рішень, що використовується для розвідки загроз (TI);
- дослідження методики пошуку IOCs.

Об'єктом дослідження є процес пошуку індикаторів компрометації з використанням фреймворків і програмного забезпечення з відкритим вихідним кодом.

Предметом дослідження є механізми та методи ручного та автоматизованого пошуку та виявлення індикаторів компрометації на основі розвідки з використанням відкритих джерел.

Ключові слова: кібербезпека, індикатори компрометації, розвідка загроз, вразливість, зловмисна активність, інцидент, логи.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ ТА СКОРОЧЕНЬ

OSINT	–	Open Source Intelligence;
APT	–	Advanced Persistant Threat;
TI	–	Threat Intelligence;
IOC	–	Indicator of Compromise;
CVE	–	Common Vulnerabilities and Exposures;
TTP	–	Tactics, Techniques, Procedures;
MITRE ATT&CK	–	MITRE Adversarial Tactics, Techniques, and Common Knowledge;
URL	–	Uniform Resource Locator;
SOC	–	Security Operations Center;
SIEM	–	Security Information and Event Management;
EDR	–	Endpoint Detection and Response;
IDS	–	Intrusion Detection System;
AWS	–	Amazon Web Service;
DMZ	–	Demilitarized Zone;
API	–	Application Programming Interface;
DLP	–	Data Leak Protection;
MISP	–	Malware Information Sharing Platform;

ВСТУП

Сьогодні важко уявити своє життя без використання інформаційно-комунікаційних систем. Вони глибоко проникли в життя сучасного суспільства. Враховуючи те, що ми живемо в умовах війни і Covid-19, ми надаємо перевагу спілкуванню онлайн та використовуємо інтернет навіть для буденних справ.

Таким чином, на простори Інтернету потрапляє безліч інформації, як відкритої, так і конфіденційної. Кожного дня зловмисники атакують веб-ресурси для того, щоб викрасти ці дані.

Завдання фахівців з кібербезпеки завчасно виявляти вразл ивості та мінімізувати ризики витоку конфіденційних інформації для того, щоб забезпечити надійність зберігання даних.

На жаль, існують компанії, які не слідкують за новинами з кіберпростору, ставлячи цим під загрозу безпеку даних, репутацію та наражають себе на великі збитки.

Прикладом можуть бути компанії Nissan Motors, FedEx, China National Petroleum, Renault SA, Deutsche Bahn, Hitachi та ін. Вони найбільше постраждали від вірусу WannaCry, який активно розповсюджувався з 12 по 15 травня 2017 року. Ця програма-вимагач стала однією з найвідоміших і наймасштабніших в історії з точки зору обсягу збитків. Орієнтовна вартість на той момент становила 4 мільярди доларів.

WannaCry, також відома як Wanna Decryptor використовувала експлойт EternalBlue(CVE-2017-0144 або «Уразливість Windows SMB Remote Code Execution Vulnerability»), який дозволяє віддаленим зловмисникам виконувати довільний код за допомогою створених пакетів. Проте, найцікавішим моментом є те, що Microsoft ще 14 березня 2017 року випустила оновлення MS17-010, яке усувало вразливість Windows Server з SMB.

Проаналізувавши дану інформацію, можна зробити висновки, що компанії могли уникнути цієї атаки, якби ідентифікували та знешкодили вразливість EternalBlue до того, як її використали актори атаки WannaCry. Саме тому, відстеження

актуальних новин з кіберпростору та вжиття певних заходів безпеки повинно бути невід'ємною частиною діяльності фахівця з кібербезпеки.

Метою роботи є дослідження алгоритму дій для пошуку індикаторів компрометації інформаційних систем на основі відкритих джерел.

Для досягнення зазначеної мети дипломної роботи поставлено *наступні завдання:*

- Провести аналіз предметної області розвідки загроз.
- Розглянути існуючі методики для аналізу та виявлення дій зловмисника в інформаційних системах.
- Створити алгоритми дій для проведення розвідки загроз в інформаційних система.
- Провести аналіз варіантів збору індикаторів компрометації.
- Розробити та розгорнути тестову інфраструктуру для демонстрації можливого варіанту проведення розвідки загроз.

Об'єктом дослідження є процес пошуку та виявлення індикаторів компрометації з використанням фреймворків і програмного забезпечення з відкритим вихідним кодом.

Предметом дослідження є деякі методи автоматизованого пошуку індикаторів компрометації на основі розвідки загроз із використанням відкритих джерел.

Методи дослідження:

- аналіз відкритих джерел;
- порівняння та аналіз існуючих рішень, що використовується для розвідки загроз (TI);
- моделювання алгоритму дій захисника на кожному етапі MITRE ATT&CK Matrix для пошуку IOCs.

Практичною цінністю є розроблений алгоритм дій для пошуку та виявлення індикаторів компрометації використовуючи деякі засоби і способи розвідки по відкритих джерелах.

РОЗДІЛ 1 ІНДИКАТОРИ КОМПРОМЕТАЦІЇ

1.1 Поняття Indicator of Compromise та Threat Intelligence

Індикатори компрометації безпосередньо пов’язані з поняттям Threat Intelligence. IOC можуть бути використані для пошуку, ідентифікації та класифікації загроз в рамках процесу розвідки загроз. Це дозволяє організаціям реагувати на потенційні загрози та приймати відповідні заходи безпеки для захисту своїх систем та даних.

Cyber Threat Intelligence (кіберрозвідка) - це пошук інформації про потенційних зловмисників, зокрема про серйозні кіберзлочинні групи, які називаються APT-угрупованнями, від Advanced Persistent Threat (ускладнена стійка загроза, або, коротко, цільова кібератака). Ці APT-угруповання являють собою стійке кіберзлочинне співтовариство, в якому ролі та обов'язки атакувальників чітко розподілені: є організатори, є програмісти, є фахівці в галузі соціальної інженерії, є навіть своя техпідтримка. Наприклад, APT29 — група загроз, яку приписують Службі зовнішньої розвідки (СЗР) Росії. Вони часто націлюються на урядові мережі в Європі та країнах-членах НАТО, дослідницькі інститути та аналітичні центри.

Кіберрозвідку можна умовно розділити на:

• стратегічну (пошук даних про потенційно небезпечні для компанії, що захищається, APT-групи, зокрема інформації про їхню підготовку до реалізації кібератаки);

• тактичну (пошук даних про тактики, техніки і процедури атакувальників, скорочено TTPs – Tactics, Techniques, Procedures);

• оперативну (пошук безпосередніх ознак приготування до атаки – специфічних мережевих сканувань для аналізу інфраструктури та пошуку вразливостей, шахрайських вхідних дзвінків і фішингових листів).

Indicator of Compromise (індикатори компрометації) - це ознаки або показники, які вказують на можливість або наявність вторгнення, компрометації або несанкціонованої діяльності в комп'ютерній системі або мережі.

Індикатори компрометації використовуються для виявлення або попередження про потенційно шкідливу або зловмисну діяльність. Наприклад, вторгнення хакерів, атаки зловмисників або поширення шкідливих програм. Ці індикатори можуть бути виявлені шляхом моніторингу за активністю системи, мережевих даних, системних журналів, реєстру або інших джерел інформації. Це дозволяє адміністраторам систем безпеки реагувати швидко і приймати відповідні заходи для запобігання подальшому проникненню або пошкодженню.

Щоб забезпечити систематичний підхід до обробки інформації, рекомендується розподілити індикатори, отримані з Threat Intelligence, на дві широкі категорії - індикатори, пов'язані з хостами, тобто конкретними комп'ютерами або серверами, і індикатори, пов'язані з мережею, тобто мережевими з'єднаннями і трафіком. Виявлення мережевих індикаторів ще не свідчить про однозначну компрометацію системи, а ось детектування хостових індикаторів, як правило, достовірно сигналізує про зловмисну активність.

До мережевих індикаторів належать домени, URL, поштові адреси, сукупність IP-адрес і портів. Хостові індикатори - це запущені процеси, зміни гілок реєстру і файлів, хеш-суми.

1.2 Розповсюдження та користь від індикаторів компрометації

Індикатори компрометації грають ключову роль у виявленні потенційної несанкціонованої діяльності та вторгнень, допомагаючи забезпечити безпеку і захист інформаційних ресурсів.

Серед причин використання IOC є багато переваг для організацій, які прагнуть захистити свої інформаційні системи:

1. Раннє виявлення загроз: IOC дозволяють організаціям виявляти інциденти безпеки та потенційні порушення на ранній стадії. Відстежуючи свої системи та мережі на предмет наявності в них індикаторів компрометації, організації можуть швидко виявляти підозрілі дії та зменшувати ризики до того, як буде завдано значної шкоди.

2. Покращене реагування на інциденти: IOCs надає критично важливу інформацію під час реагування на інциденти. Пов'язуючи IOC з конкретними суб'єктами загроз, сімействами шкідливих програм або шаблонами атак, організації можуть краще зрозуміти природу атаки, відстежити її походження і розробити ефективні стратегії реагування.

3. Проактивний захист: розповсюдження та використання IOC полегшує проактивний захист. Залишаючись в курсі останніх загроз і показників, пов'язаних зі зловмисною діяльністю, організації можуть проактивно адаптувати свої засоби контролю безпеки, оновлювати захист і мінімізувати потенційний вплив кібератак.

4. Індикатори компрометації надають змогу проводити аналіз ризиків, впливів та загроз, що стосуються організації, і допомагають визначати пріоритети або приймати компроміси відносно конкретних ризиків. Таким чином, компанії мають технічну свободу і можливість обирати власні методи захисту та рівень прийнятого ризику.

Розповсюдження індикаторів компрометації полягає в накопиченні, обміні та використанні інформації про відомі загрози, вразливості та шкідливі програми. Цей процес включає співпрацю між різними організаціями.

Індикатори компрометації можуть бути розповсюджені шляхом спільнот безпеки, обміном інформацією між організаціями, фахівцями з кібербезпеки та іншими зацікавленими сторонами. Це сприяє швидкому виявленню нових загроз і поширенню заходів для їх запобігання та реагування.

Основна мета розповсюдження індикаторів компрометації полягає в забезпеченні вчасного виявлення та реагування на кіберзагрози. Швидке поширення актуальних індикаторів дозволяє організаціям захищатися від вторгнень та зменшувати можливі шкоди від кібератак.

1.3 Джерела IOC

Security Operations Center (SOC) - це центр управління та моніторингу кібербезпеки в організації. Це спеціалізована група фахівців, які відповідають за

виявлення, аналіз і реагування на кіберзагрози та інциденти безпеки в комп'ютерних системах і мережах.

Команда SOC (Security Operations Center) може знайти індикатори компрометації для новостворених зразків зловмисного програмного забезпечення за допомогою різних джерел і методів.

Джерела, де індикатори можуть бути зібрані:

- Стрічки аналізу загроз: команди SOC часто підписуються на сервіси аналізу загроз, які надають оновлену інформацію про нові зразки зловмисного програмного забезпечення та відповідні індикатори компрометації. Ці канали збирають дані з різних джерел, зокрема від постачальників засобів безпеки, дослідницьких організацій і державних установ.

- Платформи аналізу зловмисного програмного забезпечення: команди з реагування на інциденти кібербезпеки можуть використовувати платформи аналізу зловмисного програмного забезпечення, як-от системи ізольованого програмного середовища або віртуальні машини, для виконання й аналізу підозрілих файлів. Ці платформи генерують IOC на основі поведінки, мережевого зв’язку та системної взаємодії зловмисного програмного забезпечення.

- Звіти постачальників засобів безпеки: вендори засобів безпеки регулярно публікують звіти про нові зразки зловмисного програмного забезпечення та їх IOC. Ці звіти можуть містити детальну інформацію про можливості зловмисного програмного забезпечення, вектори зараження та відповідні індикатори, які можна використовувати для виявлення та пом’якшення загрози.

- Спільноти онлайн-безпеки: команди SOC можуть брати участь у спільнотах онлайн-безпеки, форумах і списках розсилки, де фахівці з безпеки діляться інформацією про нові зразки зловмисного програмного забезпечення. Ці спільноти часто надають IOC, виявлені під час аналізу зловмисного програмного забезпечення або заходів реагування на інциденти.

- Блоги та дослідницькі статті з питань безпеки: багато дослідників та організацій із питань безпеки публікують дописи і дослідницькі статті, в яких висвітлюються їхні висновки щодо нещодавно виявлених шкідливих програм. Ці

ресурси часто включають технічний аналіз, IOC і сигнатури виявлення, які можуть бути корисними для команд із забезпечення кібербезпеки.

•Реагування на інциденти та полювання на загрози: команди SOC активно розслідують інциденти з безпекою та проводять проактивні дії з пошуку загроз. Під час цих процесів вони можуть натрапити на нові зразки зловмисного програмного забезпечення та отримати IOC із заражених систем або мережевого трафіку, пов'язаного зі зловмисним програмним забезпеченням.

•Open-Source Intelligence (OSINT): команди SOC можуть відстежувати OSINT-платформи та веб-сайти, включаючи сховища зловмисного програмного забезпечення, платформи спільного використання коду, щоб виявляти нові зразки зловмисного програмного забезпечення. Ці джерела можуть надати ранній доступ до зразків і відповідних IOC.

Незважаючи на те, що ці джерела можуть надати цінні IOC, команди SOC повинні бути впевнені в надійності і достовірності інформації, яку вони збирають. Перевірка та підтвердження IOC з багатьох джерел має вирішальне значення, щоб уникнути помилкових спрацьовувань і підвищити точність виявлення та реагування.

Перелік джерел, які можна використовувати для збору індикаторів компрометації:

1. Платформи та канали розвідки загроз:
   - AlienVault OTX [1]
   - Threat Miner [17]
2. Платформи для аналізу шкідливого програмного забезпечення та пісочниці:
   - VirusTotal [2]
   - Hybrid Analysis [3]
   - Joe Sandbox [4]
   - Cuckoo Sandbox [5]
   - Any.Run [6]
3. Звіти та блоги постачальників безпеки:
   - McAfee Labs [7]

- Cisco Talos [8]
- Trend Micro Research [9]
- Palo Alto Networks Unit 42 [10]

4. Платформи розвідки на основі відкритих джерел (OSINT):
   - VirusShare [11]
   - MalwareBazaar [12]
   - AbuseIPDB [13]
   - URLhaus [16]
5. Інтернет-спільноти та форуми з питань безпеки:
   - Reddit's /r/netsec and /r/Malware [14]
6. Урядові джерела та CERT (Computer Emergency Response Team) Sources:
   - CERT-UA (Computer Emergency Response Team of Ukraine) [15]
7. Платформи спільного використання коду:
   - APTnotes— це сховище загальнодоступних документів і блогів (відсортованих за роками), пов’язаних із шкідливими кампаніями/діяльністю/програмним забезпеченням, які були пов’язані з визначеними постачальниками групами та/або наборами інструментів APT (Advanced Persistent Threat) [18].
   - Sophoslabs - індикатори компрометації, розроблені Sophos на основі опублікованих звітів (Sophos - британська компанія, що займається програмним і апаратним забезпеченням безпеки)[19].

Ці ресурси можуть бути цінними для збору IOC та покращення можливостей аналізу загроз. Необхідно завжди оцінювати та підтверджувати IOC на основі конкретних вимог та середовища.

1.4 Pyramid of Pain

Найпопулярнішим рішенням класифікації індикаторів компрометації є “піраміда болю”, введена David Bianco у 2013 році, що зображена на рисунку 1.1. Вона демонструє взаємозв'язок між типами індикаторів, що використовуються для

виявлення діяльності зловмисника, та тим, як багато складнощів йому створює блокування певного рівня.

Дана структура, що використовується для вимірювання потенційної користі даних про загрози, оцінює користь даної інформації, враховуючи складнощі, пов'язані з отриманням цієї інформації та ухиленням від виявлення на різних рівнях (з точки зору зловмисника). Чим вище рівень в піраміді, тим більше часу та зусиль необхідно витратити на виправлення тактик і технік, які використовуються для атак на інфраструктуру.

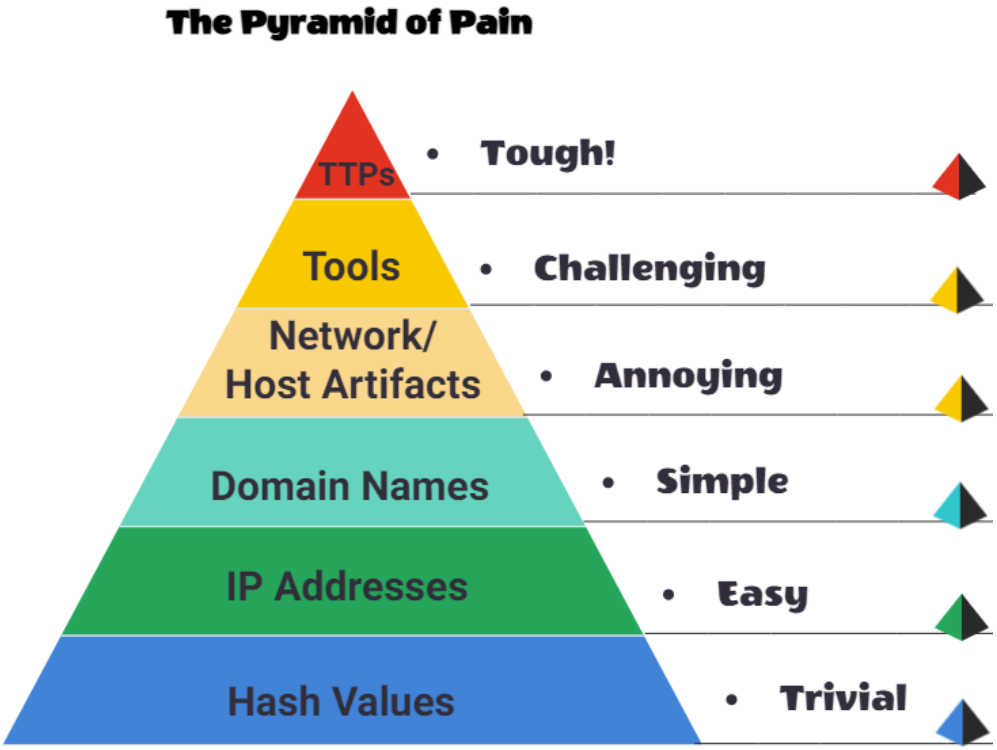


Рисунок 1.1 – Візуалізація піраміди болю

Хеш-значення — це числове значення фіксованої довжини, яке унікально ідентифікує дані. Хеш-значення є результатом алгоритму хешування. Нижче наведено деякі з найпоширеніших алгоритмів хешування:

MD5 — широко розповсюджена криптографічна хеш-функція зі 128-бітним хеш-значенням, яка розроблена Роном Рівестом у 1992 році. Хеші MD5 не вважаються криптографічно безпечними.

SHA-1(Secure Hash Algorithm 1) – був винайдений Агентством національної безпеки США в 1995 році. NIST відмовився від використання SHA-1 у 2011 році та

заборонив його використання для цифрових підписів у кінці 2013 року через те, що він був чутливим до атак грубої сили. Натомість NIST рекомендує перейти від SHA-1 до сильніших хеш-алгоритмів у сімействах SHA-2 і SHA-3.

SHA-2 (Secure Hash Algorithm 2) — алгоритм хешування SHA-2 був розроблений Національним інститутом стандартів і технологій (NIST) і Агентством національної безпеки (NSA) у 2001 році для заміни SHA-1. SHA-2 має багато варіантів, і, мабуть, найпоширенішим є SHA-256.

Фахівці з безпеки зазвичай використовують хеш-значення, щоб отримати інформацію про конкретний зразок зловмисного програмного забезпечення, зловмисний або підозрілий файл, а також як спосіб унікальної ідентифікації та посилання на шкідливий артефакт.

IP-адреса використовується для ідентифікації будь-якого пристрою, підключеного до мережі. Ці пристрої варіюються від настільних ПК до серверів і навіть камер відеоспостереження.

З точки зору захисту, знання IP-адрес, які використовує зловмисник, може бути цінним. Поширеною тактикою захисту є блокування, видалення або заборона вхідних запитів з IP-адрес на вашому периметрі або зовнішньому брандмауері. Ця тактика часто не є такою, що гарантовано забезпечить блокування шкідливого джерела в мережі, оскільки для зловмисника не проблема просто отримати нову публічну IP-адресу.

Доменні імена можна розглядати як відповідність IP-адреси та рядку тексту. Доменне ім’я може містити субдомен, за яким ідуть домен і домен верхнього рівня (fsociety.ecorp.com).

Змінити доменні імена може бути трохи складніше для зловмисника, оскільки їм, швидше за все, доведеться придбати домен, зареєструвати його та змінити записи DNS. Але велика кількість DNS-провайдерів мають слабкі стандарти та надають API – таким чином зловмисникам вдається нескладними маніпуляціями змінити домен.

Артефакти хосту. Артефакти хосту — це сліди або спостережувані елементи, які зловмисники залишають у системі, наприклад значення реєстру, підозрілі

виконання процесів, моделі атак або файли, видалені зловмисними програмами, або будь-що, що стосується поточної загрози.

В разі додання даних індикаторів до систем моніторингу кінцевих девайсів (таких як EDR, HIDS/HIPS, системи моніторингу SIEM) з'являється можливість виявити атаку, що створює проблеми зловмиснику, адже йому потрібно буде змінити інструменти та методології атаки. Це забирає багато часу для нього, і, ймовірно, йому доведеться витрачати більше ресурсів на підбір інструментарію.

Мережеві артефакти. Мережевим артефактом може бути назва агента (браузера) користувача, інформація про C2-сервер або певні URI, за якими слідують запити HTTP POST. Зловмисник може використати рядок агента користувача, який раніше не спостерігався у вашому середовищі або здається незвичайним. Агент користувача визначається RFC2616 як поле заголовка запиту, яке містить інформацію про агента користувача, який надсилає запит.

Мережеві артефакти можна виявити в Wireshark PCAP (файлі, що містять пакетні дані мережі) за допомогою аналізатора мережевого протоколу, такого як TShark , або досліджуючи журнали IDS (система виявлення вторгнень) із такого джерела, як Snort.

Інструменти. Зловмисники використовують утиліти для створення шкідливих макродокументів (maldocs) для спроб фішингу, бекдори, які можна використовувати для встановлення з'єднання з C2-серверами (інфраструктурою для командування та контролю) , будь-які власні .exe та .dll- файли, корисні навантаження або кейлогери. Антивірусні сигнатури, правила виявлення та правила YARA можуть стати чудовою зброєю для боротьби з зловмисниками на цьому етапі.

MalwareBazaar і Malshare є хорошими ресурсами, які надають доступ до зразків, шкідливих каналів і результатів YARA.

TTPs - Tactics, Techniques & Procedures. Це включає в себе всю матрицю MITRE ATT&CK , що визначає всі кроки, вжиті зловмисником для досягнення своєї мети, починаючи від спроб фішингу до стійкості та викрадання даних.

Якщо є можливість швидко виявити TTP і відреагувати на них, у зловмисника не залишається майже жодного шансу реалізувати шкідливі дії. Наприклад, якщо є

можливість виявити атаку Pass-the-Hash за допомогою моніторингу журналу подій Windows і виправити її, можна дуже швидко знайти скомпрометований хост і зупинити горизонтальний рух у мережі. На цьому етапі зловмисник матиме два варіанти:

• Повернутись, провести додаткові дослідження та навчання, підібрати інші інструменти і тактики.

• Зупинитись і спробувати знайти іншу ціль для атаки.

1.5 MITRE MATRIX

Задля проведення якісного TI необхідно обирати для себе оптимальну формальну модель згідно з якою можна передбачити подальші потенційні дії зловмисника, який намагається певним чином скомпрометувати систему. Найкращою і найпопулярнішою моделлю є Cyber Kill Chain. Kill Chain допоможе визначити TTPs і поведінку зловмисників. В цій роботі буде розглянуто і використано життєвий цикл ATT&CK від MITRE[27].

Після вибору моделі наступним кроком є проходження кожного з етапів моделі та визначення дій зловмисника, які можуть становити найбільшу загрозу. Кожен етап в моделі може включати кілька категорій тактик більш високого рівня, які може застосувати зловмисник, і які можна потім можна розбити на низку фактичних дій зловмисника.

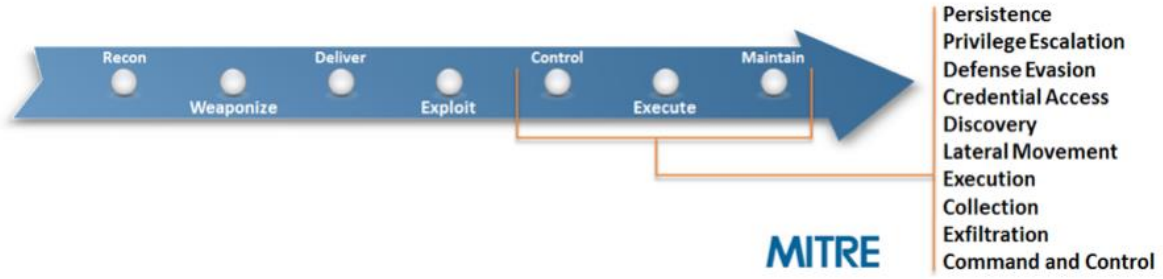


Рисунок 1.2 – Модель MITRE ATT&CK

Наприклад, останні етапи (Control, Execute and Maintain) семиступеневого життєвого циклу MITRE ATT&CK включають такі категорії, як lateral movement і data exfiltration, в рамках яких зловмисником використовується велика кількість технік, а саме:

- Malware Beaconing
- DLL Injection
- Pass the Hash (PtH)
- Shared Webroot
- DNS Tunneling

На основі даних тактик MITRE ATT&CK можна вибудувати алгоритми пошуку та виявлення та виявлення зловмисної активності в системах.

Висновки до розділу 1

У першому розділі було розглянуто поняття індикаторів компрометації та threat intelligence. Наведено декілька переваг індикаторів компрометації таких як раннє виявлення загроз або покращене реагування на інциденти. Крім того, в першому розділі класифікований та наведений перелік джерел, які можна використовувати для збору індикаторів компрометації.

Також, розділ охоплює питання розвідки загроз, що означає збір інформації про потенційні загрози, їхні характеристики та методи атаки. Розвідка загроз є важливим етапом в процесі забезпечення кібербезпеки, оскільки дозволяє заздалегідь виявити можливі шляхи атак та вжити заходів для їх запобігання.

Проаналізовано модель класифікації індикаторів компрометації “The Pyramid of Pain”, що складається з шести рівнів та демонструє взаємозв'язок між типами індикаторів, що використовуються для виявлення діяльності зловмисника.

Також цей розділ підкреслює важливість використання індикаторів компрометації та проведення розвідки загроз для забезпечення кібербезпеки. Знання про індикатори компрометації допомагає виявити потенційні загрози та недоліки у системах забезпечення безпеки, тоді як розвідка загроз надає можливість передбачити можливі атаки та вжити відповідних заходів для запобігання їм. Завдяки цим підходам, організації можуть покращити свою реакцію на потенційні кіберзагрози, зменшити ризик інцидентів та зберегти свою інформацію, системи та активи в безпеці.

РОЗДІЛ 2 РОЗВІДКА ЗАГРОЗ НА ОСНОВІ MITRE ATT&CK

2.1 MITRE ATT&CK (Adversarial Tactics, Techniques & Common Knowledge)

MITRE ATT&CK (Adversarial Tactics, Techniques & Common Knowledge) – це загально доступна база знань про тактики, техніки і інформацію про зловмисників протягом реалізації атаки.

У 2013 році MITRE представила цю матрицю (рисунок 2.1) як спосіб описати і класифікувати поведінку зловмисників на основі реальних спостережень. ATT&CK являє собою структурований список відомих типів поведінки зловмисників, які об'єднані в тактики і техніки і згруповані в декількох матрицях. Оскільки цей список досить повно відображає різну поведінку зловмисників при компрометуванні мереж, він корисний для розвідки загроз, оцінки різних захисних заходів, вивчення фактів і тд.

В даному розділі буде зображено узагальнений алгоритм дій захисника інформаційної безпеки на кожному з 14 етапі MITRE Matrix. Варто зауважити, що кожен етап має свої особливості, які потрібно враховувати захищаючи мережу.

Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Execution	Collection	Exfiltration	Command and Control
Accessibility Features	Accessibility Features	Binary Padding	Brute Force	Account Discovery	Application Deployment Software	Command-Line Interface	Automated Collection	Automated Exfiltration	Commonly Used Port
AppInit DLLs	AppInit DLLs	Bypass User Account Control	Credential Dumping	Application Window Discovery	Exploitation of Vulnerability	Execution through API	Clipboard Data	Data Compressed	Communication Through Removable Media
Basic Input/Output System	Bypass User Account Control	Code Signing	Credential Manipulation	File and Directory Discovery	Logon Scripts	Graphical User Interface	Data Staged	Data Encrypted	Connection Proxy
Bootkit	DLL Injection	Component Firmware	Credentials in Files	Local Network Configuration Discovery	Pass the Hash	InstallUtil	Data from Local System	Data Transfer Size Limits	Custom Command and Control Protocol
Change Default File Association	DLL Search Order Hijacking	Component Object Model Hijacking	Exploitation of Vulnerability	Local Network Connections Discovery	Pass the Ticket	PowerShell	Data from Network Shared Drive	Exfiltration Over Alternative Protocol	Custom Cryptographic Protocol
Component Firmware	Exploitation of Vulnerability	DLL Injection	Input Capture	Network Service Scanning	Remote Desktop Protocol	Process Hollowing	Data from Removable Media	Exfiltration Over Command and Control Channel	Data Obfuscation
Component Object Model Hijacking	Legitimate Credentials	DLL Search Order Hijacking	Network Sniffing	Peripheral Device Discovery	Remote File Copy	Regsvcs/Regasm	Email Collection	Exfiltration Over Other Network Medium	Fallback Channels
DLL Search Order Hijacking	Local Port Monitor	DLL Side-Loading	Two-Factor Authentication Interception	Permission Groups Discovery	Remote Services	Regsvr32	Input Capture	Exfiltration Over Physical Medium	Multi-Stage Channels

Рисунок 2.1 – Візуалізація MITRE Matrix

2.2 Алгоритм дій на етапі Reconnaissance

Розвідка є першим етапом. На ньому зловмисник намагається зібрати якомога більше інформації про систему, на яку планує кібератаку. Він виконує активний і пасивний збір інформації.

Пасивний збір даних передбачає використання відкритих джерел для виявлення ір-адрес, доменів і піддоменів, розміщення серверів, хостинг, склад колективу і тд. Особливістю пасивного збору інформації є те, що його дуже складно виявити, адже він не викликає підозр в команди захисників на відміну від активного.

З іншого боку, активна розвідка вимагає від хакерів взаємодії з цільовою системою або мережею. Вони шукають відкриті порти , які можуть бути використані як точки входу для нападу. У результаті вони можуть виконувати ручне тестування або автоматичне сканування, використовуючи різні методи. Активна розвідка набагато ризикованіша через підвищену ймовірність виявлення брандмауером або іншим рішенням безпеки. Проте багато хто продовжує вести активну розвідку для підвищення точності атаки.

Найпростіший спосіб захисту від активної розвідки – це застосування надійних заходів безпеки, таких як належне налаштування IDS/IPS, EDR і підтримка цих механізмів. Захисникам потрібно постійно моніторити мережу на предмет будь-якої незвичної активності та вчасно реагувати на всі підозрілі події в мережі.

Отже, на етапі розвідки, зловмисник намагається виявити слабкі місця цільової системи використовуючи техніки інформаційної розвідки. В свою чергу, захисники, щоб протидіяти цьому, повинні вживати заходів для виявлення будь-якої підозрілої активності.

Нижче наведений структурований алгоритм дій, які варто виконувати для того, щоб виявити кібератаку на її першій стадії.

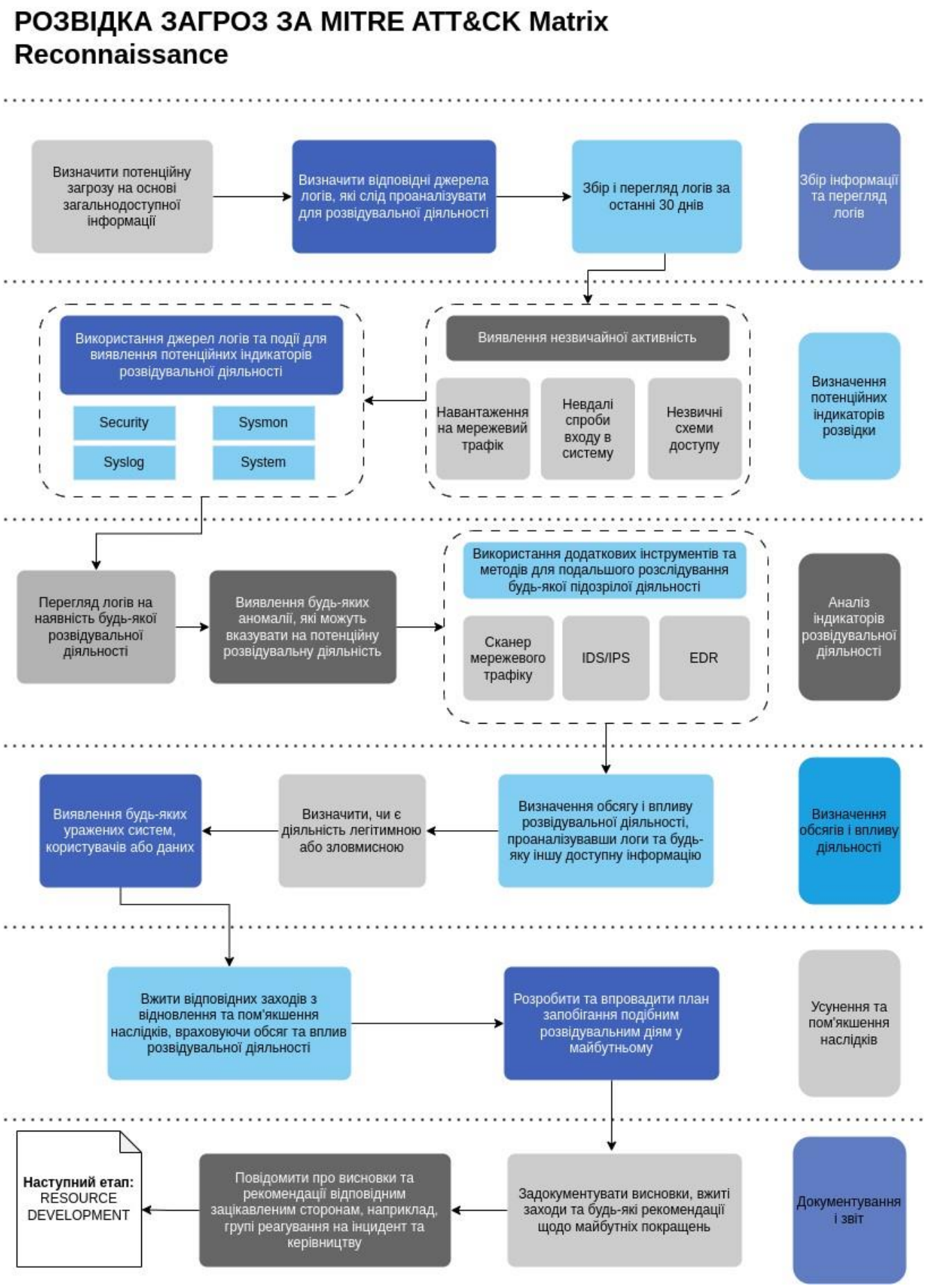


Рисунок 2.2 – Алгоритм дій на етапі розвідки

2.3 Алгоритм дій на етапі Resource Development

На цьому етапі зловмисник, аналізуючи дані, які зібрав на першому етапі, розробляє ресурси, які від буде використовувати для реалізації кібератаки.

На цьому етапі зловмисники можуть:

• купувати або іншим чином отримувати доступ до системи чи мережі;

• купувати або орендувати інфраструктуру(ботнети);

• придбати домени, наприклад, для фішингу;

• орендувати віртуальні приватні сервери, для того, щоб мінімізувати фізичну прив'язку до них;

• придбати та налаштувати безсерверну хмарну інфраструктуру, таку як Cloudflare Workers або функції AWS Lambda, які можуть ускладнити приписування їм інфраструктури, яка використовується під час операцій;

• купувати онлайн-рекламу для розповсюдження зловмисного програмного забезпечення;

• виявляти аутсайдера для отримання доступу до цільової системи;

скомпрометувати облікові записи співробітників для отримання доступу;

розробити зловмисне програмне забезпечення та його компоненти, які можна використовувати під час націлювання.

Щоб захистити інформаційну систему, команда захисників повинна зробити ряд дій, які допоможуть завадити зловмиснику продовжити атаку. На цьому етапі зловмисник може і не взаємодіяти напряму з цільовою системою і це значно ускладнює процес виявлення його в системі. Проте команда забезпечення інформаційної та кібербезпеки повинна діяти, щоб зменшити загрозу та мінімізувати ризики.

Як і на кожному етапі, одним з основних кроків є збір логів: мережевих та з кінцевих точок. Всі зібрані події потрібно проаналізувати на предмет наявності в них будь-якої підозрілої активності. Далі пошук індикаторів компрометації, шляхом виявлення підозрілих процесів, IP-адрес чи доменів, незвичних шляхів, розширень файлів і тд.

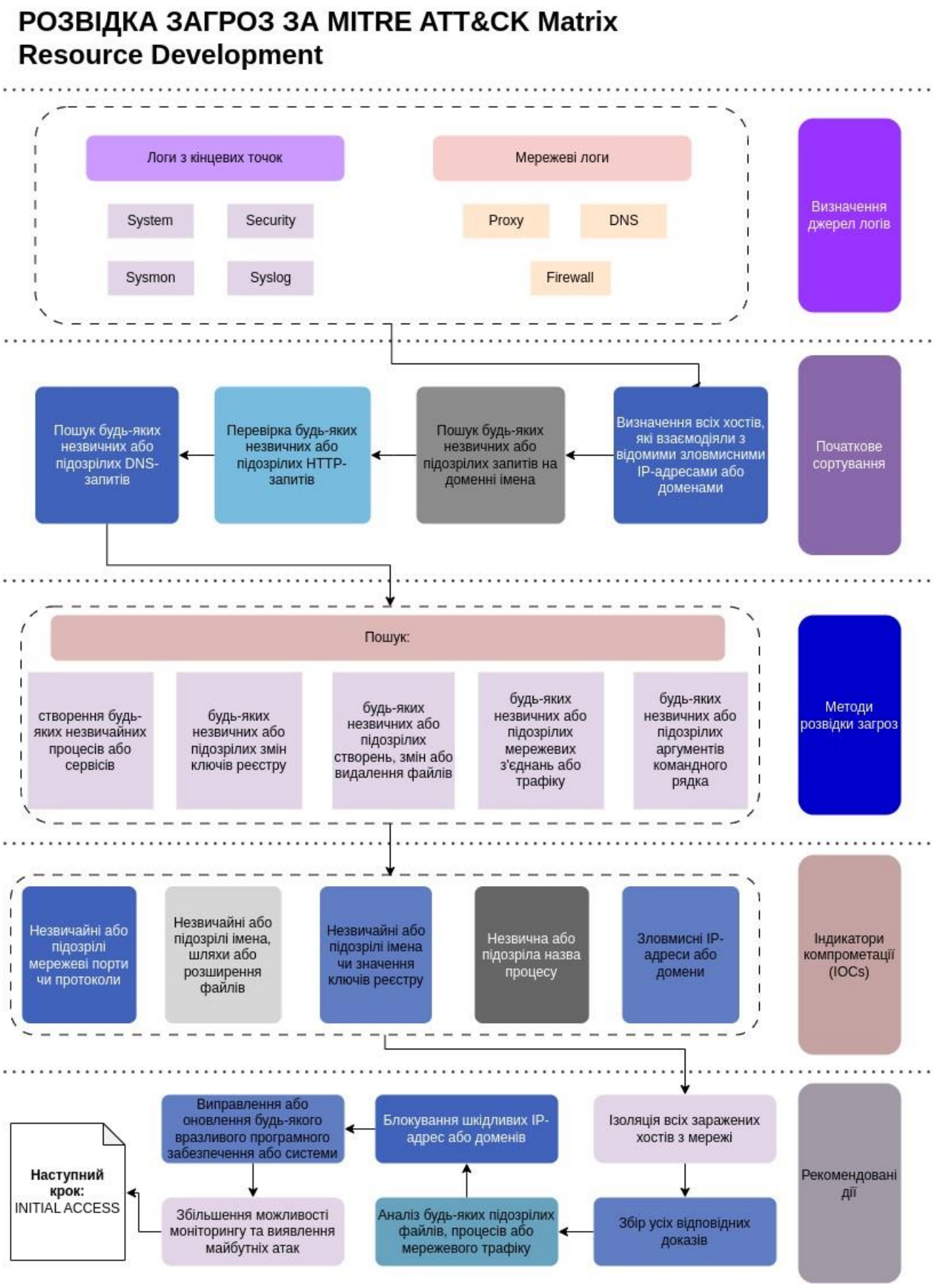


Рисунок 2.3 – Алгоритм дій на етапі розробки ресурсів

2.4 Алгоритм дій на етапі Initial Access

Третій етап MITRE ATT&CK зосереджений на початковому доступі. Зловмисник намагається проникнути на цільову систему. Ця тактика вимагає особливої уваги експертів з безпеки, оскільки без початкового доступу зловмисники не зможуть перевести кібератаку на інший рівень.

На цьому етапі зловмисники можуть використовувати:

• Цілеспрямований фішинг. Спонукати законного користувача виконати звичайну дію, як-от відкрити документ в електронному листі, є одним із найпростіших способів для зловмисників встановити зловмисне програмне забезпечення з бекдором або навіть прямим корисним навантаженням (T1566).

• Експлуатація публічних програм. Техніка (T1190) призначена для ряду складних експлойтів на рівні програмного забезпечення для роботи в Інтернеті. Зловмисники намагаються використовувати помилки, збої, зіткнення та недоліки конструкції, щоб проникнути в системи жертв. Якщо компрометація відбувається в хмарі, яка часто використовується в наші дні, вони намагатимуться уникнути основного екземпляра або контейнера та рухатися вбік або далі вниз до хоста.

• Злом законних облікових записів користувачів. Техніка дійсних облікових записів (T1078) описує ситуацію, коли зловмисники отримують доступ шляхом зламу реальних облікових записів користувачів. Це можуть бути облікові записи працівників, а також довірені облікові записи третіх сторін, наприклад ділових партнерів або підрядників. Ця техніка, як правило, є багатоетапною та передбачає поєднання інших методів початкового доступу, таких як фішинг, зовнішні віддалені служби та експлойт загальнодоступної програми.

Щоб захистити мережу на цьому рівні, захисники повинні вчасно виявляти будь-які зловмисні дії, оперативно реагувати на них. А також використовувати антивірусне програмне забезпечення, сегментувати мережу за допомогою DMZ, застосовувати політики контролю доступу, використовувати IDS.

Нижче наведений алгоритм дій, якого варто дотримуватись, для того, щоб мінімізувати загрозу на цьому етапі.

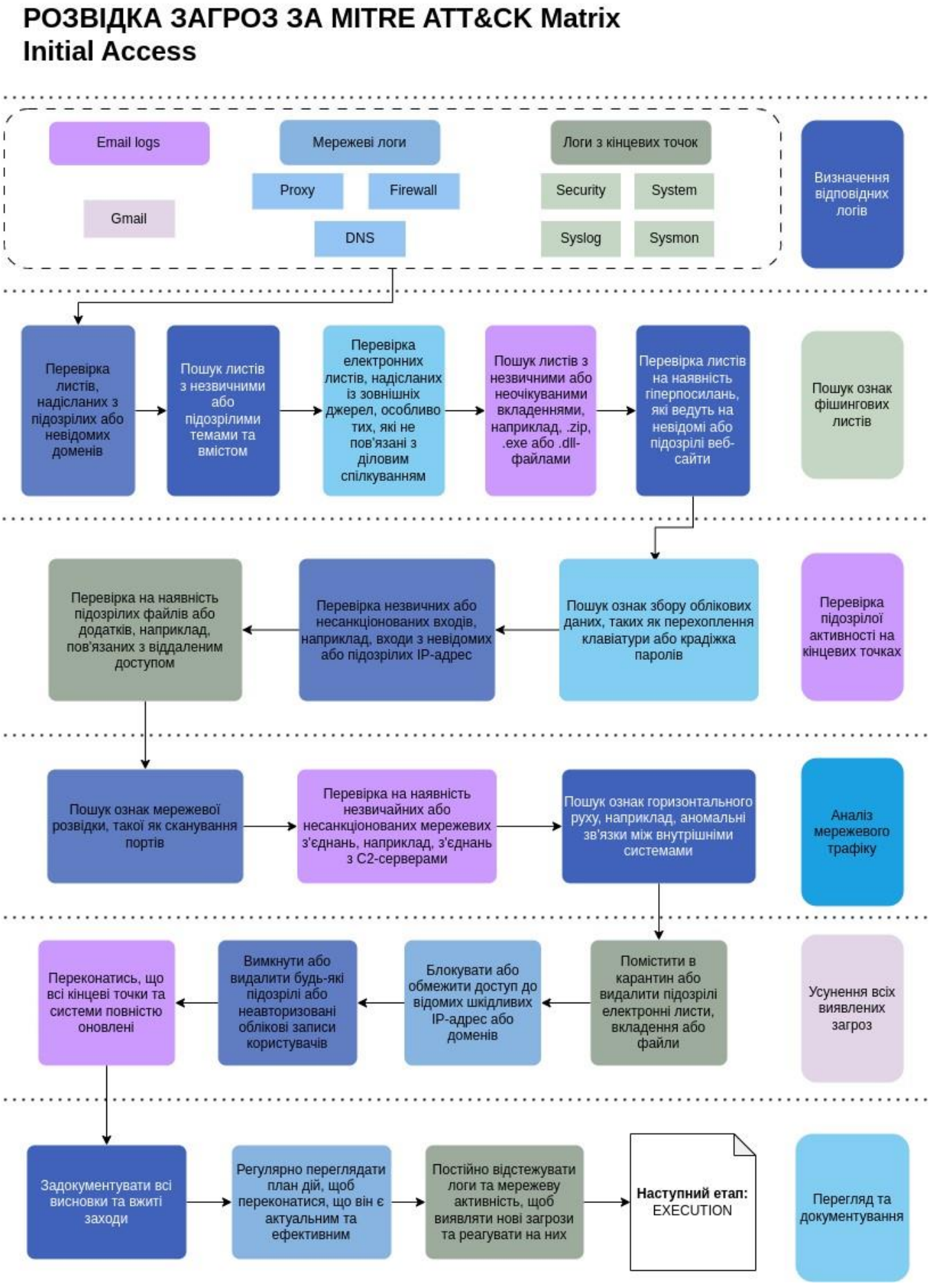


Рисунок 2.4 – Алгоритм дій на етапі початкового доступу

2.5 Алгоритм дій на етапі Execution

Виконання складається з прийомів, які призводять до запуску зловмисного коду у локальній або віддаленій системі. Це може бути виконання коду .exe, .dll або програми Java. Техніки, які запускають шкідливий код, часто поєднуються з техніками інших тактик для досягнення ширших цілей, як-от дослідження мережі чи викрадення даних. Наприклад:

1. Зловмисник може використовувати інструмент віддаленого доступу для запуску сценарію PowerShell, який виконує віддалене виявлення системи.
2. Зловмисники можуть використовувати вразливості програмного забезпечення в клієнтських програмах для виконання коду.
3. Зловмисники можуть зловживати cron утилітою для виконання планування завдань для початкового або повторного виконання шкідливого коду.
4. Зловмисники можуть зловживати командами та сценаріями PowerShell для виконання.
5. Зловмисники можуть зловживати хмарними службами керування для виконання команд у віртуальних машинах або гібридних пристроях.

Для того, щоб вчасно виявляти виконання необхідно розуміти можливу сферу застосування, аналізувати події та реагувати на будь-яку аномальну активність, наприклад, запуск незвичного процесу, підозріле завантаження PowerShell чи виконання з підозрілими аргументами командного рядка. Це і є індикатори компрометації.

Не менш, важливим кроком є вжиття відповідних заходів відносно IOCs знайдених раніше. Будь-яку шкідливу активність необхідно блокувати шляхом оновлення сигнатур IPS чи антивірусу. Вражені хости варто ізолювати для того, щоб мінімізувати можливість поширення шкідливої активності. У разу необхідності потрібно задокументувати та передати інцидент відповідній команді.

Нижче наведений детальний алгоритм розвідки загроз на етапі виконання, який допоможе захистити мережу.

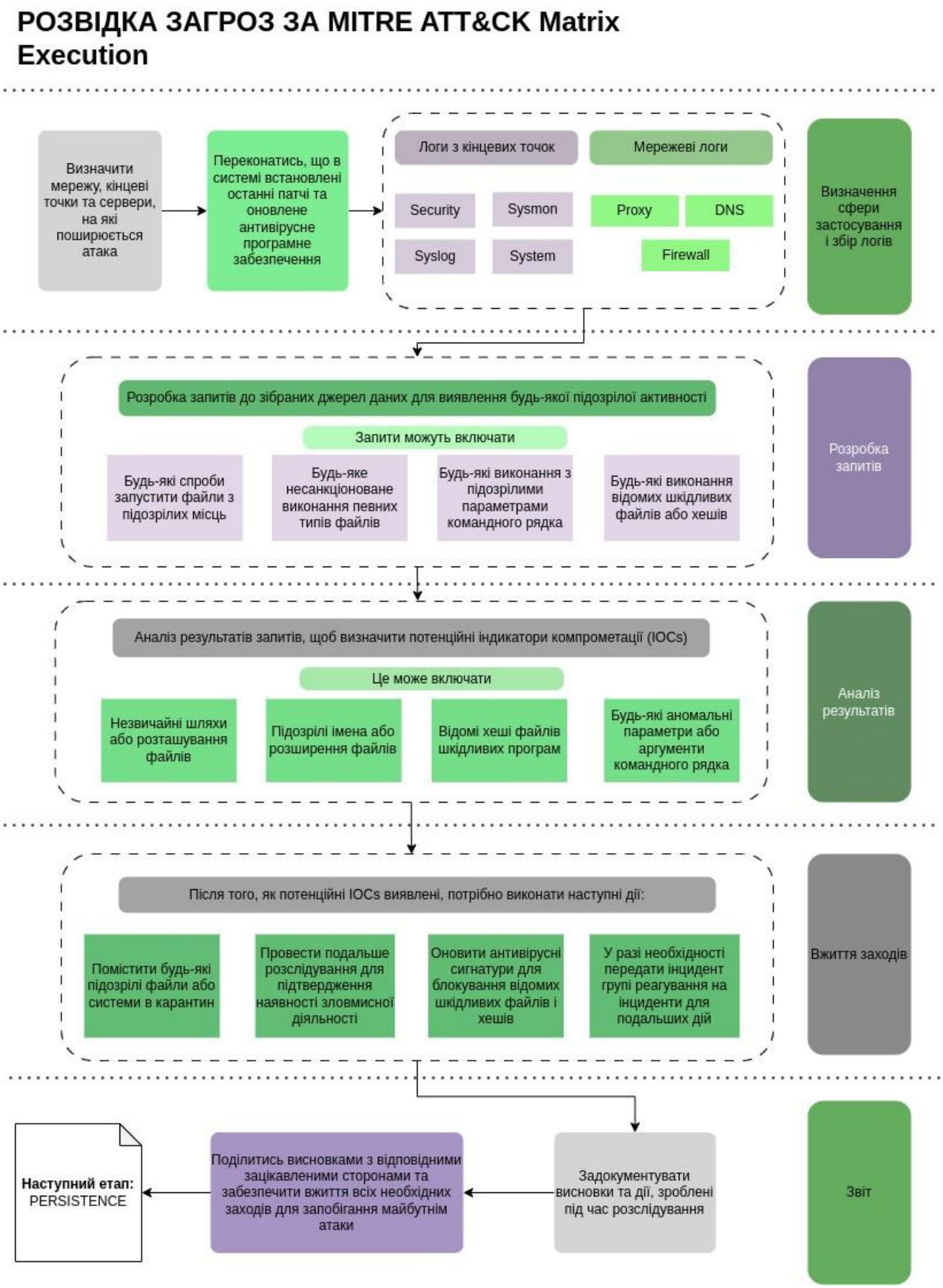


Рисунок 2.5 – Алгоритм дій на етапі виконання

2.6 Алгоритм дій на етапі Persistence

На цьому етапі зловмисник намагається утриматись в цільовій мережі. Основне завдання закріплення доступу полягає в забезпеченні сталої присутності в атакованій системі, адже доступ може бути втрачено у зв'язку з перезапуском цільової системи, втратою облікових даних або блокуванням інструментів віддаленого доступу внаслідок виявлення атаки.

Методи забезпечення сталості в системі можна умовно розділити на 3 категорії:

1. Несанкціоноване створення облікових записів або маніпуляції чи крадіжка наявних облікових даних. Ці дії можуть також включати активність облікового запису, спрямовану на обхід політик безпеки, наприклад, ітеративне оновлення паролів, щоб обійти політику тривалості паролів і продовжити термін дії скомпрометованих облікових даних.

2. Приховане встановлення і запуск засобів віддаленого доступу. Зловмисники можуть налаштувати параметри системи на автоматичне виконання програми під час завантаження або входу в систему, щоб зберегти стійкість або отримати привілеї вищого рівня на скомпрометованих системах.

3. Внесення в конфігурацію цільової системи змін, за допомогою яких стає можливий численний запуск шкідливого коду. Шкідливий код може автоматично запускатися під час кожного завантаження системи або кожного входу користувача в систему, запуску модифікованих або шкідливих служб, запуску користувачем певних програм, запуску процесів оновлення системного або стороннього ПЗ.

Етап закріплення є дуже важливим, оскільки успішність реалізації надає зловмиснику постійний доступ до цільової системи. Проте, якщо адміністратори безпеки будуть дотримуватись певного алгоритму дій розвідки загроз, вони зможуть завадити зловмисникам закріпитись в мережі.

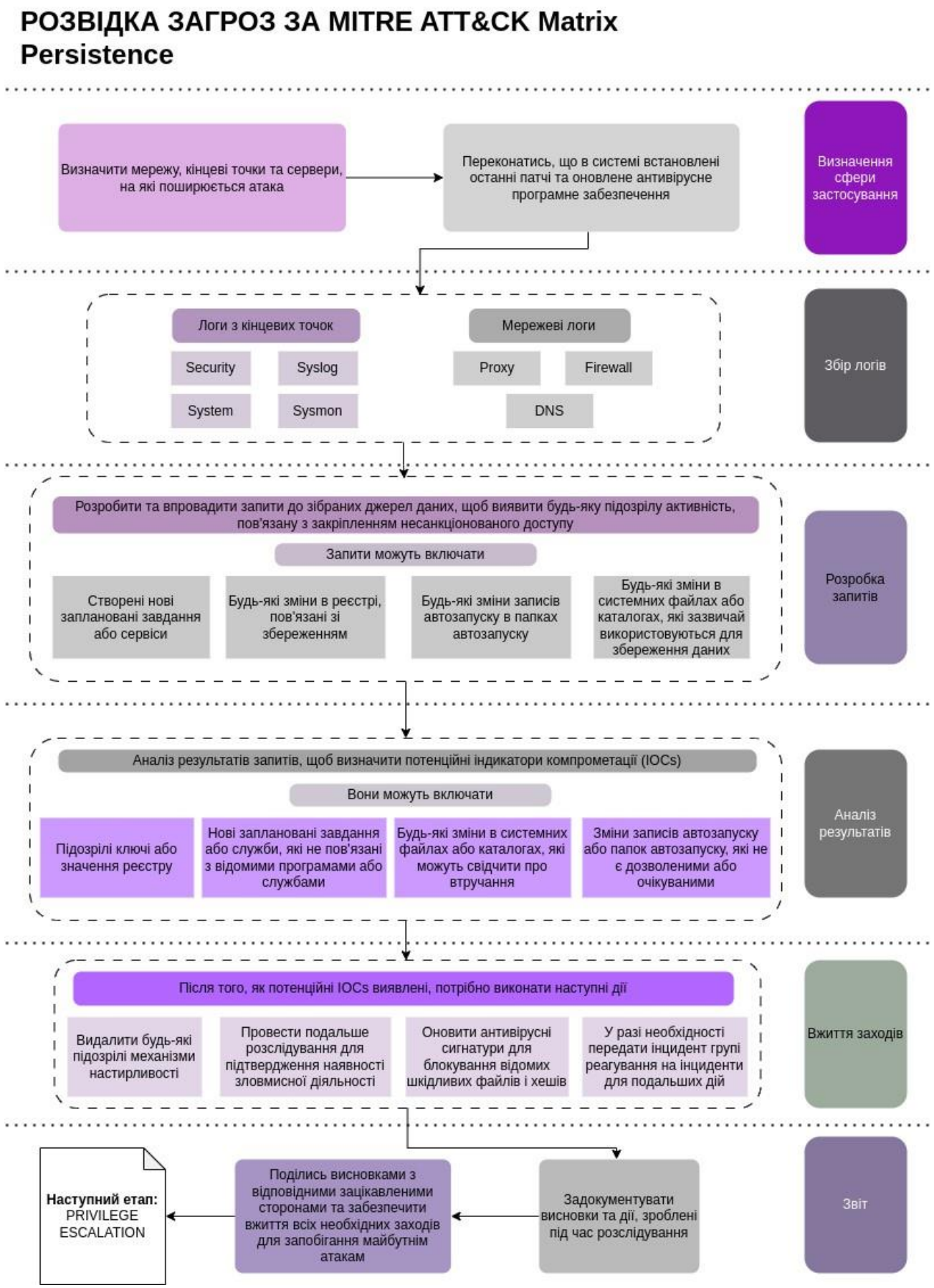


Рисунок 2.6 – Алгоритм дій на етапі закріплення

2.7 Алгоритм дій на етапі Privilege Escalation

Ескалація привілеїв - це результат дій, які дають змогу зловмиснику або шкідливій програмі отримати вищий рівень дозволів в атакованій системі або мережі. Зловмисники часто можуть входити та досліджувати мережу з непривілейованим доступом, але потребують підвищених дозволів для досягнення своїх цілей.

Техніки ескалації привілеїв описують методи, за допомогою яких зловмисник, отримавши непривілейований доступ до атакованої системи, використовуючи різноманітні вразливості системи, може отримати права локального адміністратора, system або root, облікові записи користувачів, які мають доступ до певної системи або виконують певну функцію. Використання зловмисниками облікових записів користувачів із правами доступу до конкретних систем або дозволами на виконання певних операцій також може розглядатися як ескалація привілеїв.

Важливо зазначити, що деякі техніки, що описуються в матриці ATT&CK, одночасно включені в кілька етапів ланцюжка атаки, наприклад, перехоплення пошуку DLL можна використовувати як для закріплення доступу шляхом несанкціонованого виконання шкідливої DLL, так і для підвищення привілеїв шляхом запуску DLL у процесі, що працює в контексті більш привілейованого користувача.

Розвідка загроз та захист мережі є надзвичайно важливим на цьому етапі. Потрібно не забувати обмежувати права звичайних користувачів та слідкувати за будь-якими змінами в облікових записах юзерів і груп. Також блокувати спроби використання відомих вразливостей по підвищенню привілеїв. Слідувати принципу найменших привілеїв також може бути хорошою практикою, яка дозволить мінімізувати можливість реалізації зловмисної активності. Нижче зображений детальний алгоритм дій, якого варто дотримуватись щоб виявити несанкціоновані спроби підвищення привілеїв.

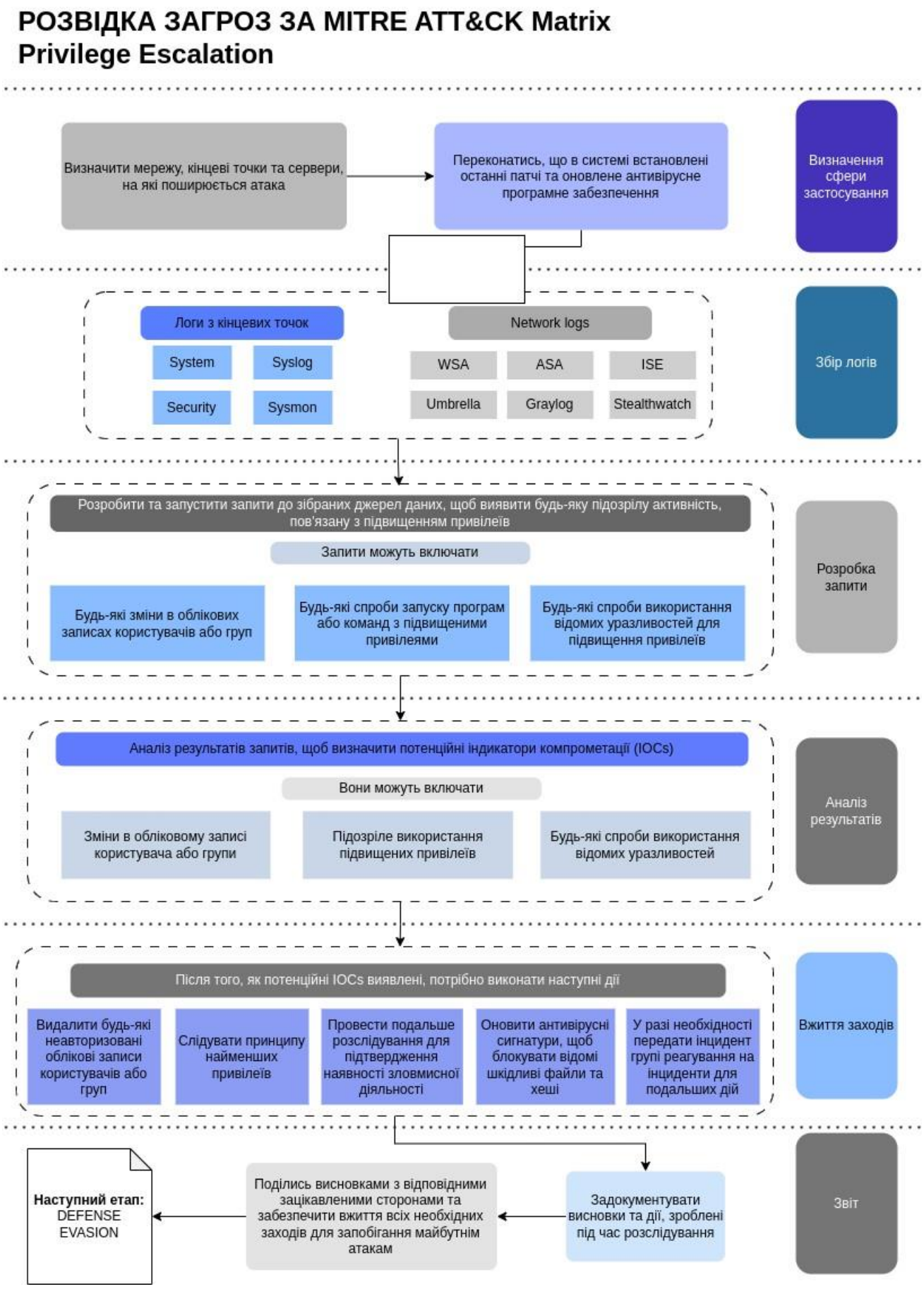


Рисунок 2.7 – Алгоритм дій на етапі підвищення привілеїв

2.8 Алгоритм дій на етапі Defense Evasion

У розділі "Обхід захисту" описуються техніки, за допомогою яких зловмисник може приховати шкідливу активність і запобігти своєму виявленню засобами захисту.

Методи, які використовуються для ухилення від захисту, включають видалення/вимкнення програмного забезпечення безпеки або обфускацію/шифрування даних і сценаріїв. Зловмисники також використовують довірені процеси та зловживають ними, щоб приховати та маскувати своє шкідливе програмне забезпечення.

Для того, щоб захистити свою мережу на цьому етапі потрібно регулярно оновлювати антивірусні сигнатури для виявлення зловмисних програм, ідентифікувати та блокувати потенційно зловмисне програмне забезпечення, видаляти будь-які зловмисні інструменти та скрипти, які можуть використовуватись в зловмисних цілях.

Вектор захисту необхідно спрямувати на запобігання шкідливим діям на більш ранніх етапах атаки, наприклад, на стадії доставки або створення шкідливого файлу в системі, а також на запобігання запуску потенційно-небезпечного та шкідливого ПЗ. Також варто застосовувати парольну політику, дотримуватись рекомендацій з проєктування та адміністрування корпоративної мережі для обмеження використання привілейованих облікових записів на всіх адміністративних рівнях. Обов’язково проводити регулярні перевірки доменних, локальних облікових записів та їхніх прав з метою виявлення тих, які можуть дозволити зловмиснику отримати широкий доступ та моніторити активності облікових записів за допомогою SIEM-систем.

Нижче наведена узагальнена схема дій захисника для того, щоб виявити зловмисника, який намагає приховати свою присутність в системі. Цей алгоритм не передбачає особливостей конкретної системи, тому захисникам варто враховувати окремі моменти захищаючи певну систему.

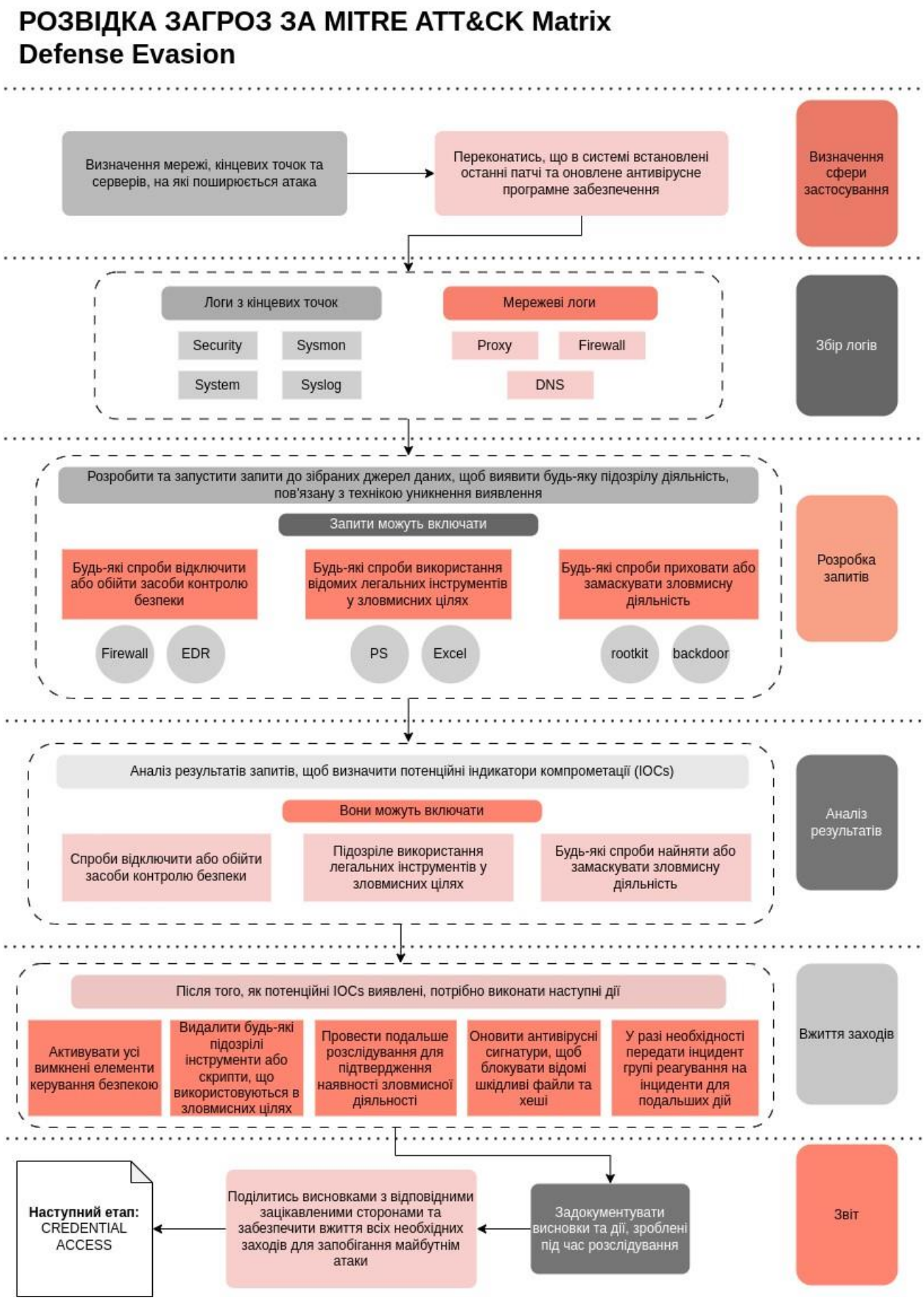


Рисунок 2.8 – Алгоритм дій на етапі обходу захисту

2.9 Алгоритм дій на етапі Credential Access

Доступ до облікових даних складається з методів викрадення облікових даних, таких як імена облікових записів та паролі. Методи, що використовуються для отримання облікових даних, включають перехоплення клавіатури або дамп облікових даних. Використання законних облікових даних може надати зловмисникам доступ до систем, ускладнити їхнє виявлення та надати можливість створити більше облікових записів для досягнення своїх цілей.

Отримавши облікові дані, зловмисник отримує доступ або навіть контроль над системою, доменом або службовими (технологічними) обліковими записами. Зловмисник, імовірно, намагатиметься роздобути легітимні облікові дані призначених для користувача й адміністративних облікових записів, щоб ідентифікуватися в системі й отримати всі дозволи захопленого облікового запису, тим самим ускладнюючи стороні, що захищає, завдання з виявлення шкідливої активності. Зловмисник також, за наявності можливості, може створювати облікові записи з метою їх подальшого використання в середовищі, що атакується.

Убезпечити свою систему від несанкціонованого доступу до облікових даних можна шляхом:

- використання багатофакторної аутентифікації, захисту контролерів домену, забезпечивши обмеження доступу до цих систем;
- застосування політики блокування облікових записів після певної кількості невдалих спроб входу в систему;
- використання складних й унікальних паролів для облікових даних локального адміністратора в усіх системах і сегментах мережі;
- виявленням та блокуванням потенційно-небезпечного та шкідливого програмного забезпечення, яке може бути використане для отримання дампів облікових даних;

Нижче наведений структурований алгоритм дій, які варто виконувати для того, щоб виявити компрометацію облікових даних.

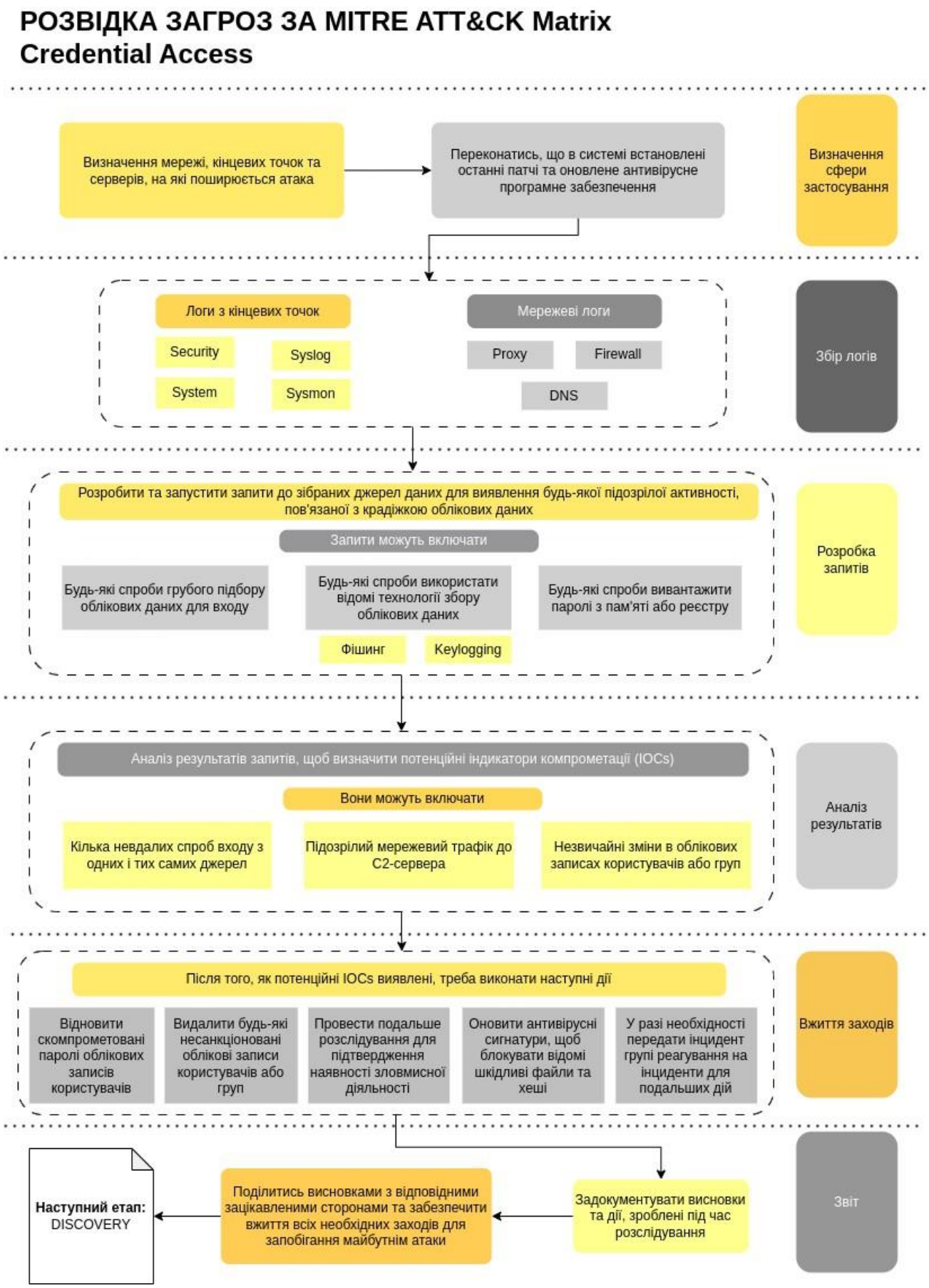


Рисунок 2.9 – Алгоритм дій на етапі доступу до облікових даних

2.10 Алгоритм дій на етапі Discovery

Виявлення складається з методів, які зловмисник може використовувати для отримання інформації про систему та внутрішню мережу. Ці прийоми допомагають супротивникам спостерігати за навколишнім середовищем і зорієнтуватися, перш ніж вирішити, як діяти. Власні інструменти операційної системи часто використовуються для цієї мети збору інформації після зламу. Отримавши, внаслідок первинної компрометації, доступ до системи, зловмисник має "озирнутися", зрозуміти, що він тепер контролює, які можливості в нього з'явилися і чи достатньо поточного доступу для досягнення тактичної або кінцевої мети.

Операційні системи мають безліч вбудованих інструментів, за допомогою яких зловмисник може здійснювати дослідження внутрішнього периметра цільової системи після її компрометації. У Windows для збору інформації можуть використовуватися інструменти прямої взаємодії з Windows API або PowerShell.

Зловмисник застосовує методи сканування під час вивчення середовища, що атакується, тому виявлення подібної активності слід розглядати як частину ланцюжка атаки, за якою послідують спроби просування супротивника мережею.

Як захід, спрямований на виявлення вищеописаної активності в системах, що захищаються, рекомендовано моніторинг процесів і аргументів командного рядка, які можуть використовуватися під час збору інформації про систему або мережу. Загальною рекомендацією щодо запобігання можливості несанкціонованого внутрішнього дослідження системи, що захищається, є проведення аудиту наявності непотрібних системних утиліт і потенційно-небезпечного ПЗ, які можуть використовуватися для вивчення середовища, що захищається, і застосування інструментів блокування їхнього запуску.

Рекомендації щодо захисту: застосовувати IDS/IPS-системи для виявлення і запобігання віддаленого сканування, бути впевненим, що непотрібні порти закриті, невикористовувані служби відключені, а правильна сегментація мережі дотримується для захисту критично-важливих серверів і пристроїв.

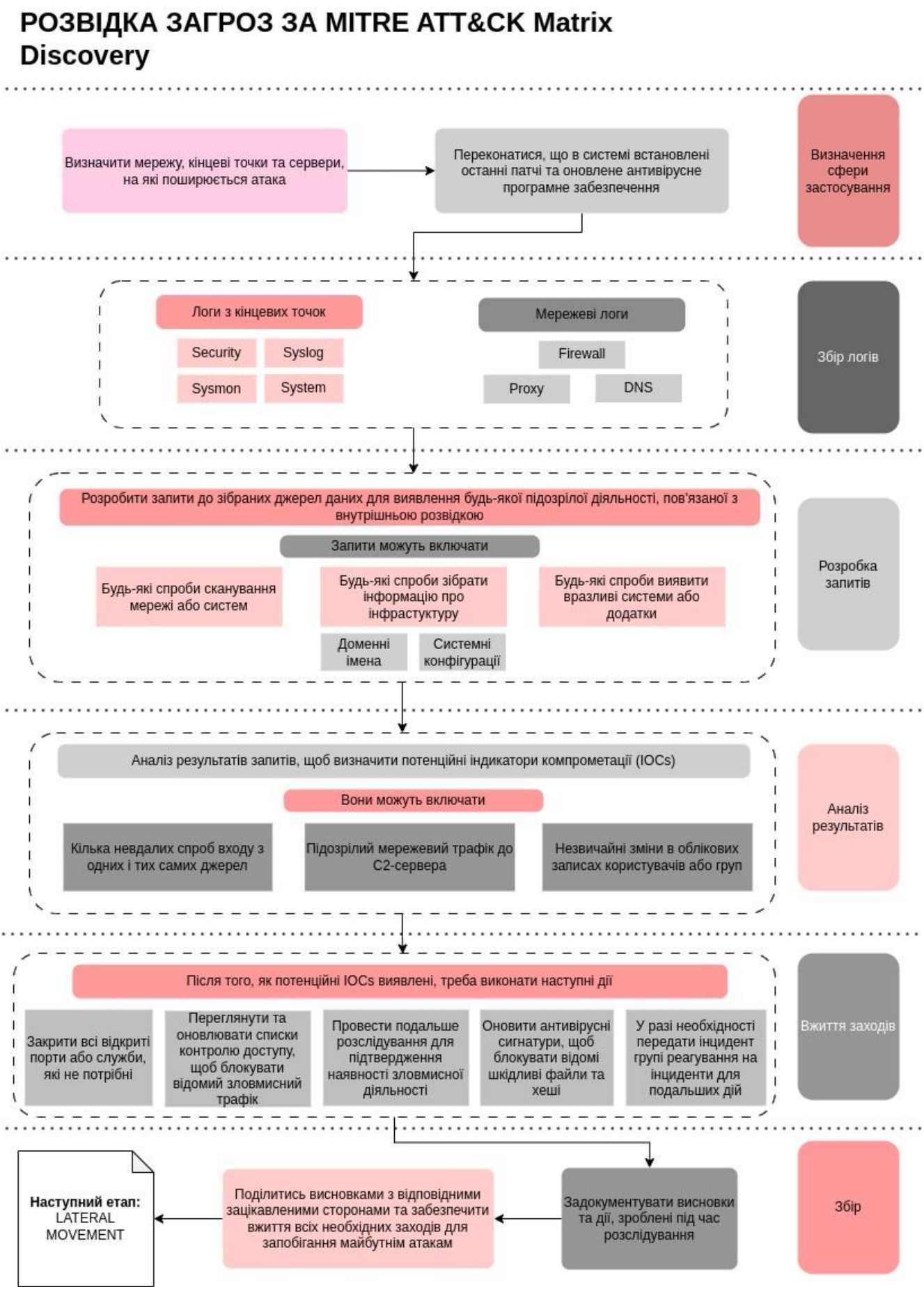


Рисунок 2.10 – Алгоритм дій на етапі виявлення

2.11 Алгоритм дій на етапі Lateral Movement

Тактика горизонтального руху охоплює методи отримання супротивником доступу і контролю над віддаленими системами, під'єднаними до атакованої мережі. Зловмисники можуть установити власні інструменти віддаленого доступу для виконання горизонтального переміщення або використовувати легітимні облікові дані з рідними інструментами мережі та операційної системи, які можуть бути більш прихованими. Бічне переміщення мережею дає змогу зловмиснику отримувати інформацію з віддалених систем.

Запобіжні заходи:

- доступ до систем розгортання застосунків тільки у обмеженого числа авторизованих адміністраторів;
- надійна ізоляція та обмеження доступу до критично важливих мережевих систем за допомогою брандмауерів;
- обмеження привілеїв облікових записів, налаштування групових політик безпеки та багатофакторна аутентифікація;
- регулярне встановлення виправлення та оновлення систем встановлення застосунків, щоб запобігти можливості отримання до них несанкціонованого віддаленого доступу за допомогою експлуатації вразливостей;
- сегментація мережі та системи, щоб зменшити доступ до критично важливих систем і сервісів;
- регулярна перевірка внутрішньої мережі на наявність нових і потенційно вразливих сервісів;
- постійний моніторинг системних і доменних журналів з метою виявлення незвичайної авторизації в системі.

Запобіжні заходи можуть відрізнятись відповідно до особливостей окремих систем та мереж.

Нижче наведений узагальнений алгоритм дій захисника, для того щоб виявляти будь-яку зловмисну активність, пов’язану з горизонтальним рухом.

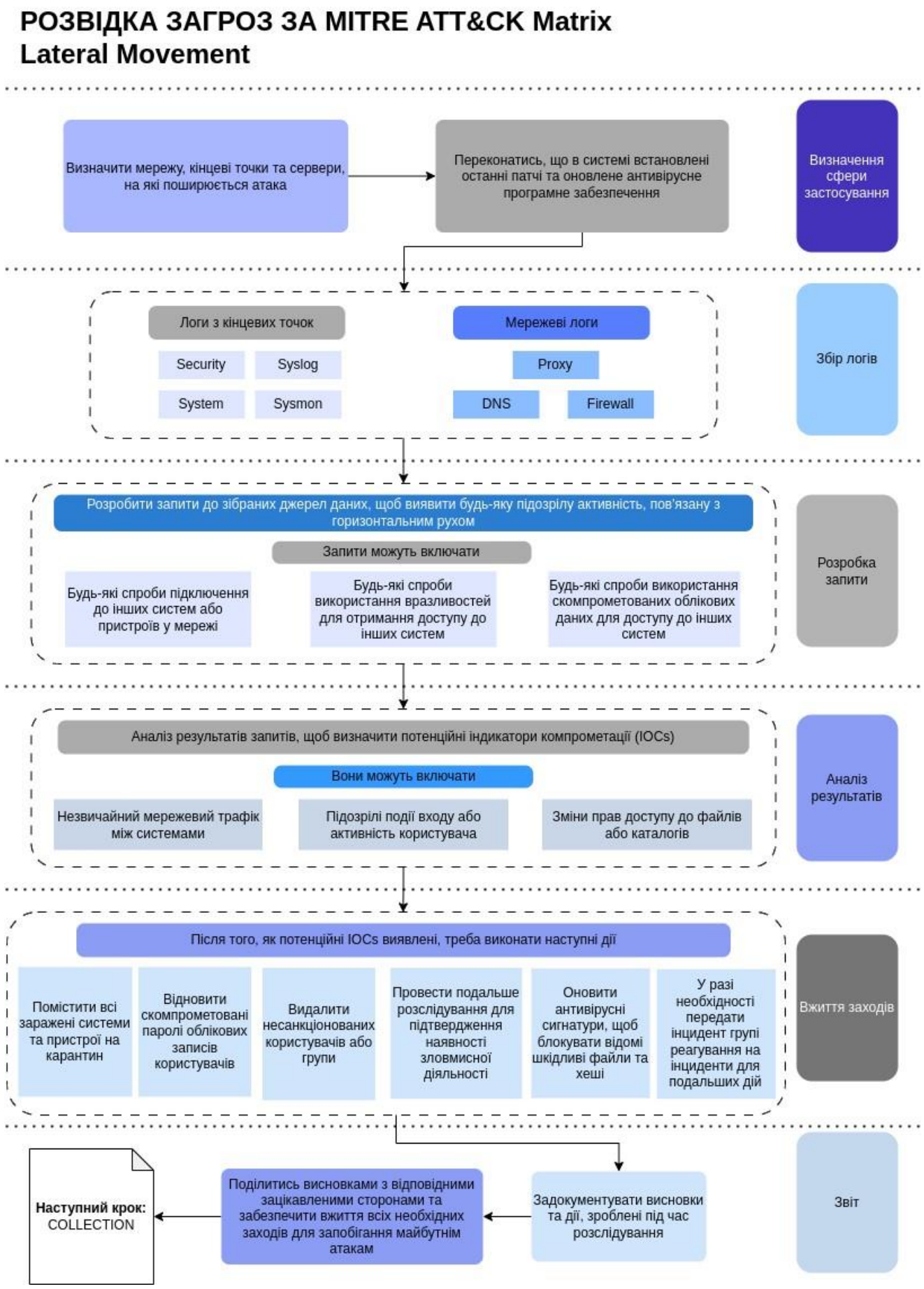


Рисунок 2.11 – Алгоритм дій на етапі горизонтального руху

2.12 Алгоритм дій на етапі Collection

Техніки збору даних у скомпрометованому середовищі включають способи ідентифікації, локалізації та безпосередньо збору цільової інформації (наприклад, конфіденційних файлів) з метою її підготовки до подальшої ексфільтрації. Опис методів збору інформації також охоплює опис місць зберігання інформації в системах або мережах, у яких зловмисники можуть здійснювати її пошук і збір.

Індикаторами реалізації більшості представлених в ATT&CK технік збору даних є процеси, що використовують API, PowerShell, Cmd або Bash для захоплення цільової інформації з пристроїв вводу/виводу або множинного відкриття файлів на читання з подальшим копіюванням отриманих даних в певне місце у файловій системі або мережі. Інформацію під час збирання даних можна шифрувати й об'єднувати в архівні файли.

Як загальні рекомендації щодо захисту від збору даних пропонуються:

- блокування будь-якого несанкціонованого мережевого трафіку на зовнішній IP-адрес або домен;
- перегляд та оновлення прав доступу до файлів або каталогів;
- шифрування і зберігання конфіденційної інформації поза межами системи;
- забезпечення принципу найменших привілеїв;
- реалізація механізмів контролю доступу, які включають в себе як аутентифікацію, так і відповідну авторизацію;
- використання другого фактору для доступу до ресурсів, для того щоб запобігти використання зловмисником чужих облікових записів;
- забезпечення виявлення та блокування потенційно зловмисного програмного забезпечення.

Нижче зображений покроковий алгоритм дій, які повинен зробити захисник, щоб мінімізувати ризики на цьому етапі MITRE Matrix. Варто звернути увагу, що ключовим вектором захисту мережі є обмеження прав доступу користувачів для того, щоб запобігти можливості компрометації облікового запису.

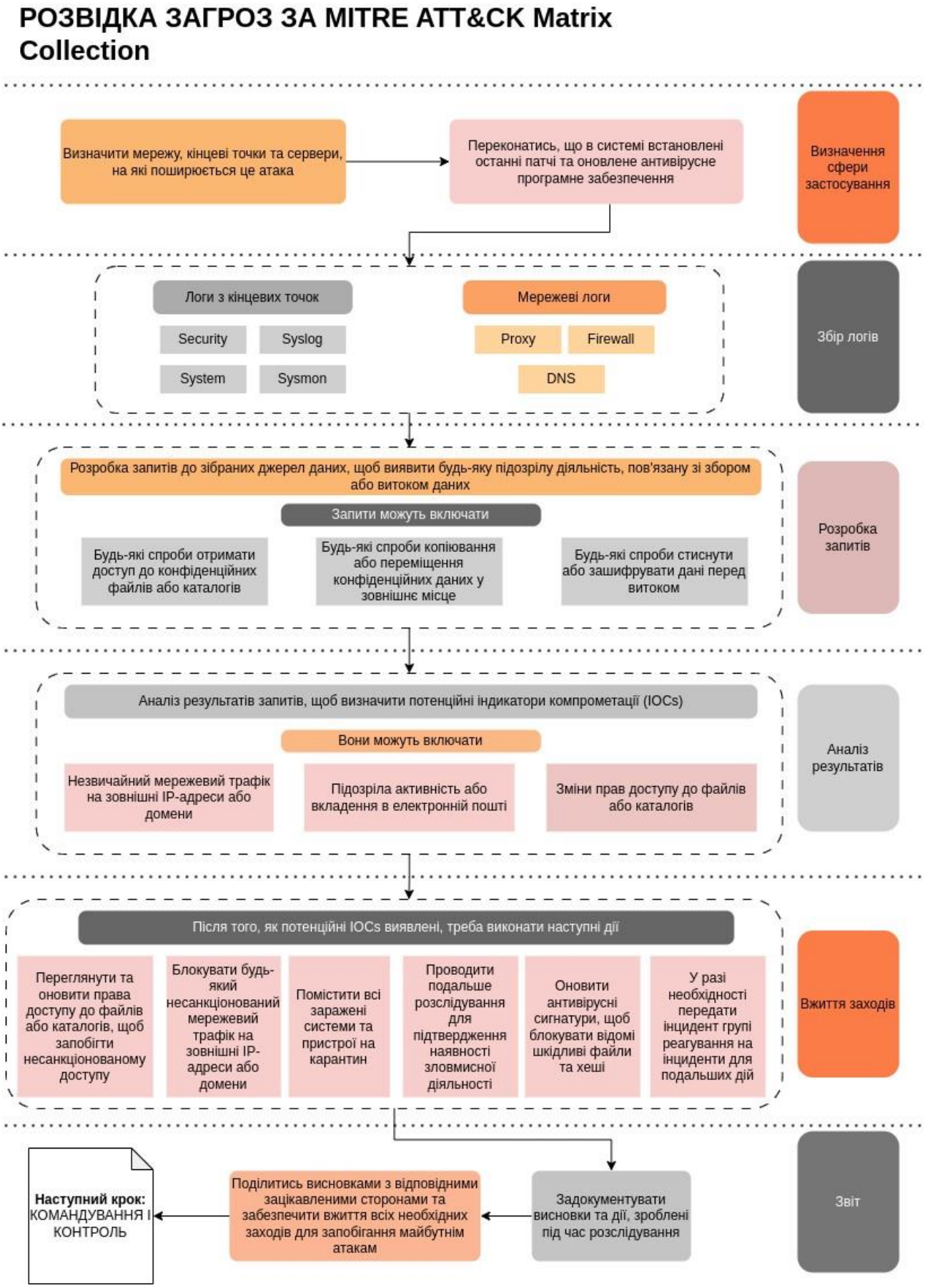


Рисунок 2.12 – Алгоритм дій на етапі збору даних

2.13 Алгоритм дій на етапі Command and Control

Командування і управління включає техніки, за допомогою яких зловмисник комунікує з системами, підключеними до мережі, що атакується, та які перебувають під його управлінням. Зловмисники зазвичай намагаються імітувати нормальний очікуваний трафік, щоб уникнути виявлення. Існує багато способів, за допомогою яких зловмисник може встановити командування та контроль із різними рівнями скритності залежно від структури мережі та засобів захисту жертви. Наприклад:

- Зловмисники можуть спілкуватися за допомогою протоколів прикладного рівня OSI, щоб уникнути виявлення/фільтрації мережі шляхом змішування з існуючим трафіком;
- Зловмисники можуть здійснювати командування та контроль між скомпрометованими хостами в потенційно відключених мережах, використовуючи знімні носії для передачі команд від системи до системи;
- Зловмисники можуть кодувати дані, щоб ускладнити виявлення вмісту командного та контрольного трафіку;

Загальні рекомендації щодо організації заходів із запобігання та виявлення C2:

- IDS/DLP-системи, що використовують сигнатурний аналіз трафіку, можуть застосовуватися для виявлення і блокування відомих засобів C2 і шкідливих програм;
- організація моніторингу викликів API-функцій, пов'язаних з увімкненням або використанням альтернативних каналів зв'язку;
- обмеження вихідного трафіку, дозволяючи на міжмережевих екранах і проксі-серверах тільки необхідні порти через відповідні мережеві шлюзи;
- використання інструментів організації білих списків застосунків, щоб ускладнити інсталяцію стороннього програмного забезпечення;

Основним завданням захисника на цьому етапі є виявлення індикаторів компрометації та вжиття відповідних заходів.

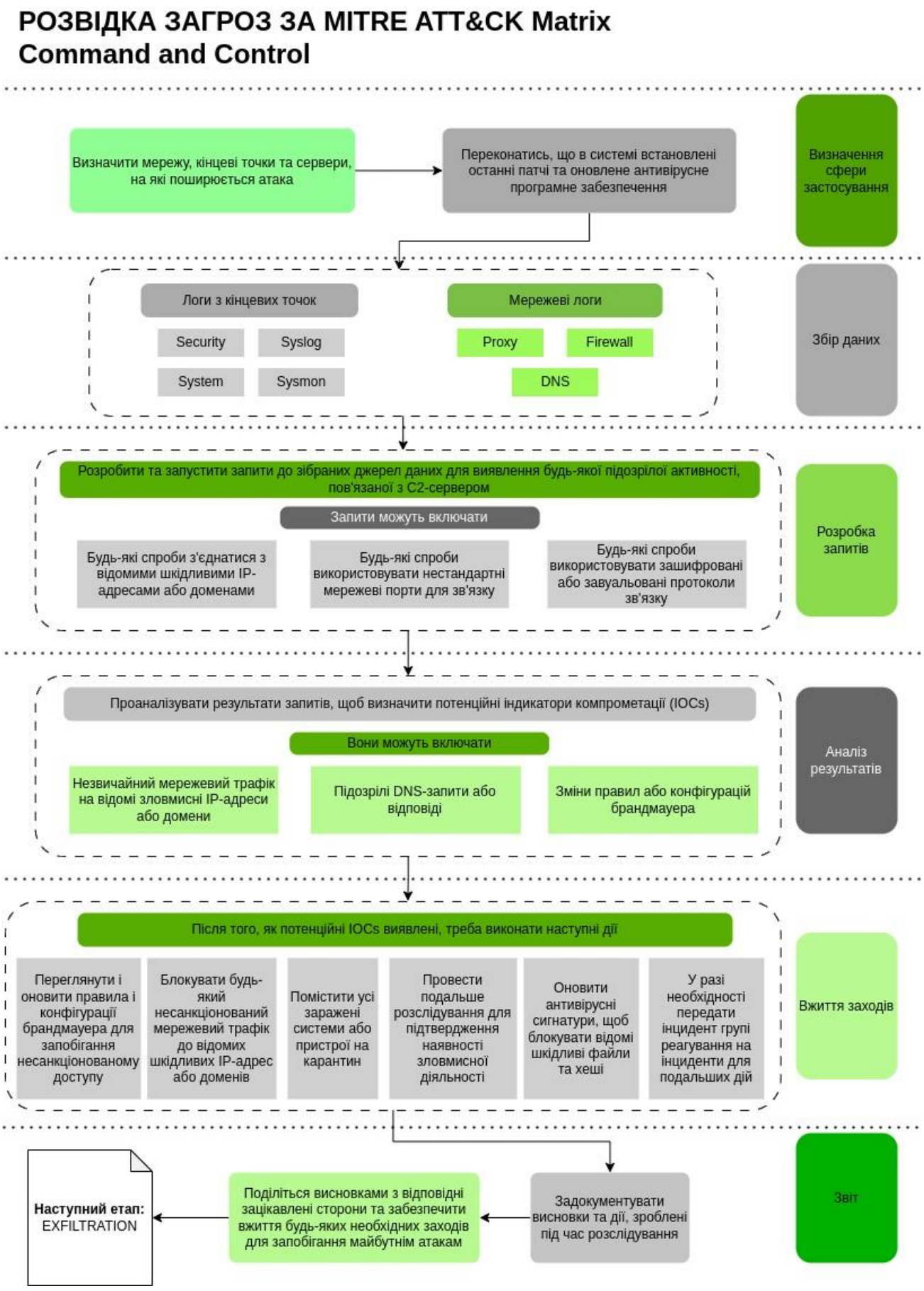


Рисунок 2.13 – Алгоритм дій на етапі командування і управління

2.14 Алгоритм дій на етапі Exfiltration

Викрадання складається з прийомів, які зловмисники можуть використовувати для викрадення даних із цільової мережі. Зібравши дані, зловмисники часто пакують їх, щоб уникнути виявлення під час видалення. Це може включати стиснення та шифрування. Методи отримання даних із цільової мережі зазвичай передбачають передачу їх через їхній канал керування або альтернативний канал, а також можуть включати встановлення обмежень на розмір передачі.

Для виявлення і запобігання відомим способам організації каналу управління та ексфільтрації даних варто застосовувати системи IDS/IPS, що використовують сигнатурний аналіз трафіку. З метою обходу IPS або DLP, які блокують передачу незашифрованими каналами зв'язку файлів певного типу або тих, що містять певний заголовок, зловмисник може перейти на використання шифрування каналу ексфільтрації. ПЗ для стиснення і стислі файли можуть бути завчасно виявлені за допомогою моніторингу процесів і аргументів командного рядка, пов'язаних із викликом відомих утиліт стиснення даних, однак такий підхід передбачає аналіз великої кількості помилкових подій.

Як техніку виявлення також рекомендується аналіз мережевого трафіку щодо незвичайних потоків даних (наприклад, клієнт надсилає значно більше даних, ніж отримує з сервера). Не відповідність використовуваного номера порту і номера порту, встановленого в мережевому протоколі за замовчуванням, може також вказувати на шкідливу активність. Важливо переконатись, що сенсори засобів захисту на хостах підтримують аудит використання всіх мережевих адаптерів і, за можливості, запобігають підключенню нових. Потрібно вимкнути автозапуск змінних пристроїв зберігання інформації та заборонити або обмежити використання змінних пристроїв на рівні політики безпеки організації, якщо вони не потрібні для бізнес-операцій.

Нижче зображена узагальнена схема дій захисника на цьому етапі.

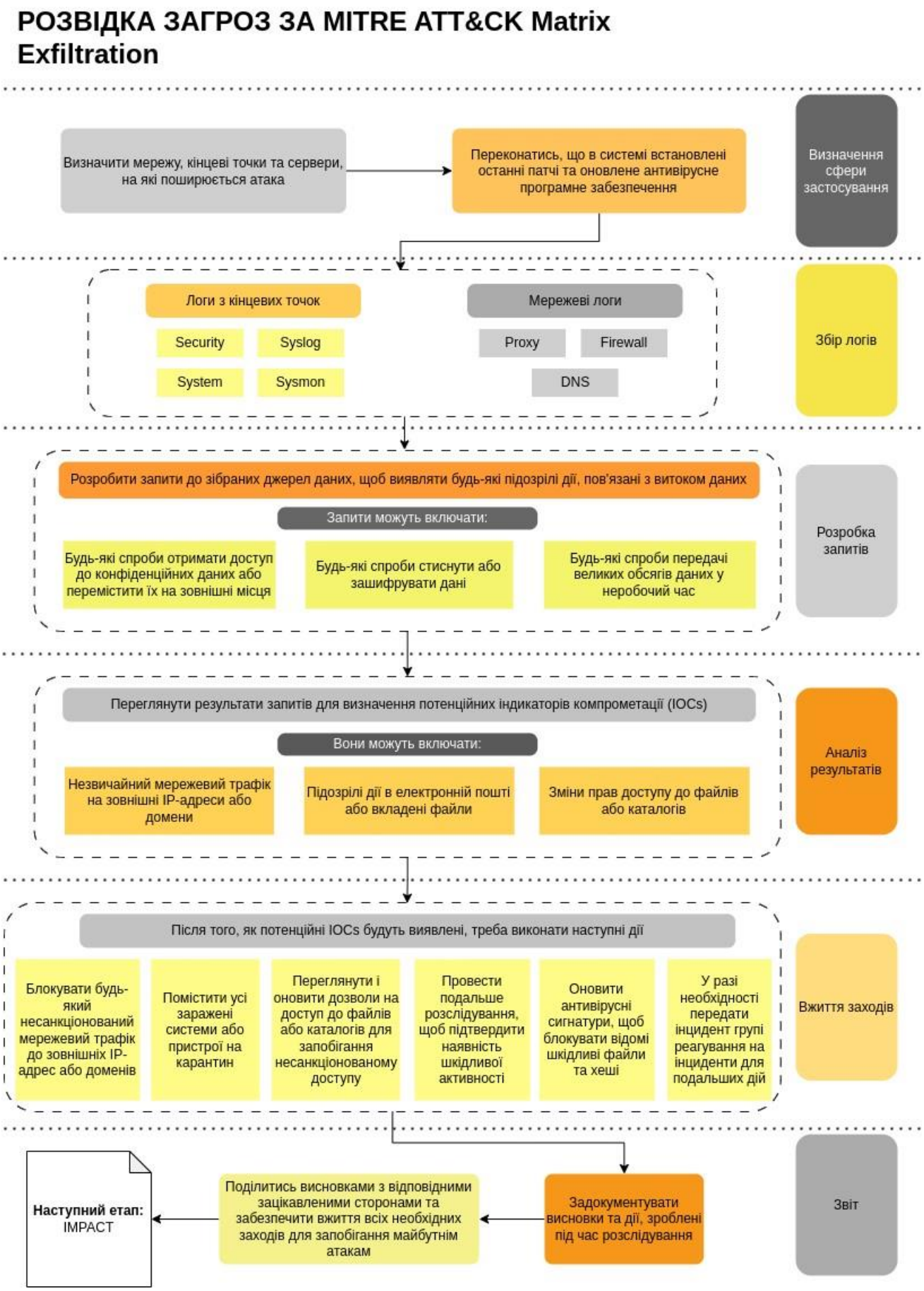


Рисунок 2.14 – Алгоритм дій на етапі викрадання даних

2.15 Алгоритм дій на етапі Impact

На цьому етапі зловмисник намагається вплинути на систему максимально масштабно і негативно. Він може маніпулювати, перервати або знищити ваші системи та дані. Вплив складається з методів, які використовують зловмисники, щоб порушити доступність або скомпрометувати цілісність шляхом маніпулювання бізнес-процесами та операційними процесами. Методи, які використовуються для впливу, можуть включати знищення або підробку даних. У деяких випадках бізнес-процеси можуть виглядати добре, але, можливо, були змінені на користь цілей супротивників. Деякі техніки, які використовує зловмисник на цьому етапі:

- Зловмисники можуть перервати доступність системних і мережевих ресурсів, перешкоджаючи доступу до облікових записів, якими користуються користувачі.
- Зловмисники можуть знищити дані та файли в окремих системах або у великій кількості в мережі, щоб перервати доступність систем, служб і мережевих ресурсів.
- Зловмисники можуть шифрувати дані на цільових системах або на великій кількості систем у мережі, щоб перервати доступність системних і мережевих ресурсів.
- Зловмисник може пошкодити внутрішні системи організації, намагаючись залякати або ввести в оману користувачів, таким чином дискредитуючи цілісність систем.
- Зловмисники можуть здійснювати атаки на відмову в обслуговуванні (DoS), щоб погіршити або заблокувати доступність послуг для користувачів.
- Зловмисники можуть використовувати уразливості програмного забезпечення, які можуть спричинити збій програми чи системи та заборону доступу для користувачів.

Нижче зображений узагальнений план дій зловмисника на останньому етапі MITRE Matrix, який спрямований на мінімізацію впливу зловмисника на систему.

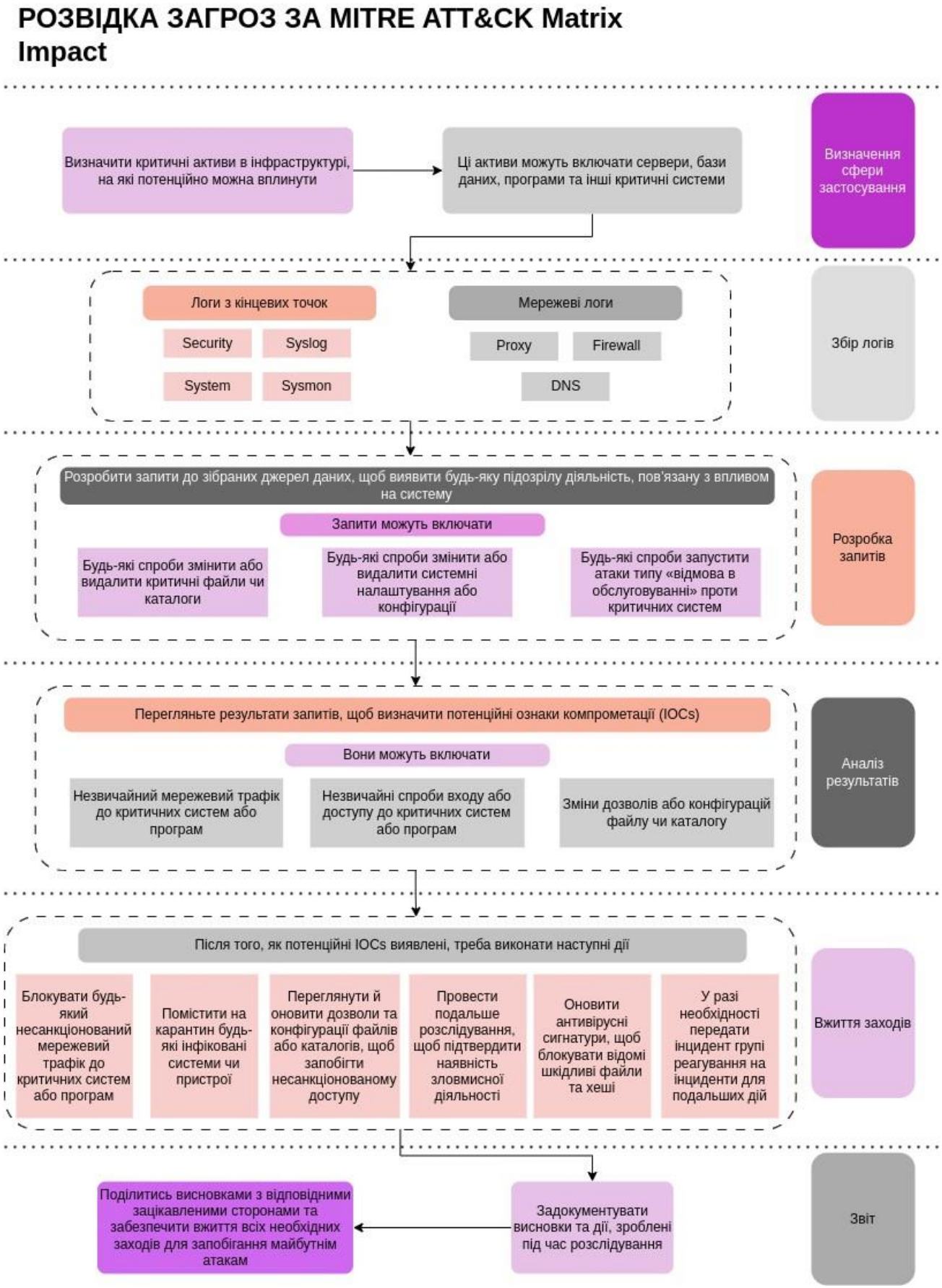


Рисунок 2.15 – Алгоритм дій на етапі впливу

Висновки до розділу 2

MITRE ATT&CK Matrix – це загальнодоступна база знань про тактики та техніки зловмисників протягом реалізації кібератаки. Структура ATT&CK є потужним інструментом для покращення кіберзахисту та розвідки загроз.

В цьому розділі була проведена аналітична робота щодо переліку можливих дій зловмисника на кожному з 14 етапів. Були розглянуті основні тактики, а також деякі техніки, пов'язані з кожною тактикою. Цей огляд надав чітку основу для подальшого аналізу розвідки загроз.

Також було розроблено алгоритм дій захисника мережі на основі MITRE ATT&CK Matrix з метою ефективного виявлення і протидії загрозам кібербезпеки. Розуміння та використання MITRE ATT&CK Matrix є важливим інструментом для захисника мережі, оскільки ця матриця дає повну картину можливих атак та етапів їх виконання.

На кожному етапі MITRE ATT&CK Matrix були визначені ключові кроки та рекомендації для захисника мережі. Спочатку проводилася підготовка до атаки, включаючи визначення сфери застосування та збір логів, розробка запитів до зібраних подій та виявлення індикаторів компрометації. Після виявлення загроз захисник мережі реагував на них шляхом вжиття відповідних заходів безпеки, таких як блокування атакуючого IP-адреси, відключення скомпрометованих облікових записів, патчінг вразливостей і переведення мережі в безпечний стан. Крім того, відбувалася постійна перевірка та оновлення систем безпеки для ефективного виявлення майбутніх атак.

У результаті застосування алгоритму дій на кожному етапі MITRE ATT&CK Matrix, захисник мережі зміцнює безпеку і здатність виявлення загроз. Цей алгоритм надає систематичний та структурований підхід до боротьби з кібератаками, допомагаючи захисникам ефективно виявляти, відповідати та мінімізувати вплив загроз на мережу.

РОЗДІЛ 3 ЗАСОБИ ЗБОРУ ТА ВИЯВЛЕННЯ ІНДИКАТОРІВ КОМПРОМЕТАЦІЇ В ХМАРНОМУ СЕРЕДОВИЩІ

3.1 Ручний збір індикаторів компрометації

Як було з’ясовано раніше, збір і виявлення індикаторів компрометації є однією з ключових аспектів розвідки загроз на кожному етапі MITRE ATT&CK Matrix. Один зі способів виявлення компрометації - це пошук індикаторів компрометації на доступних веб-ресурсах. В першому розділі було зазначено ряд доступних веб-сторінок, на яких можна знайти індикатори компрометації.

Наприклад, на рисунку 3.1 зображена стаття з українського ресурсу CERT-UA, в якій описується шпигунська активність UAC-0063, а на рисунку 3.2 є індикатори компрометації, які її характеризують.

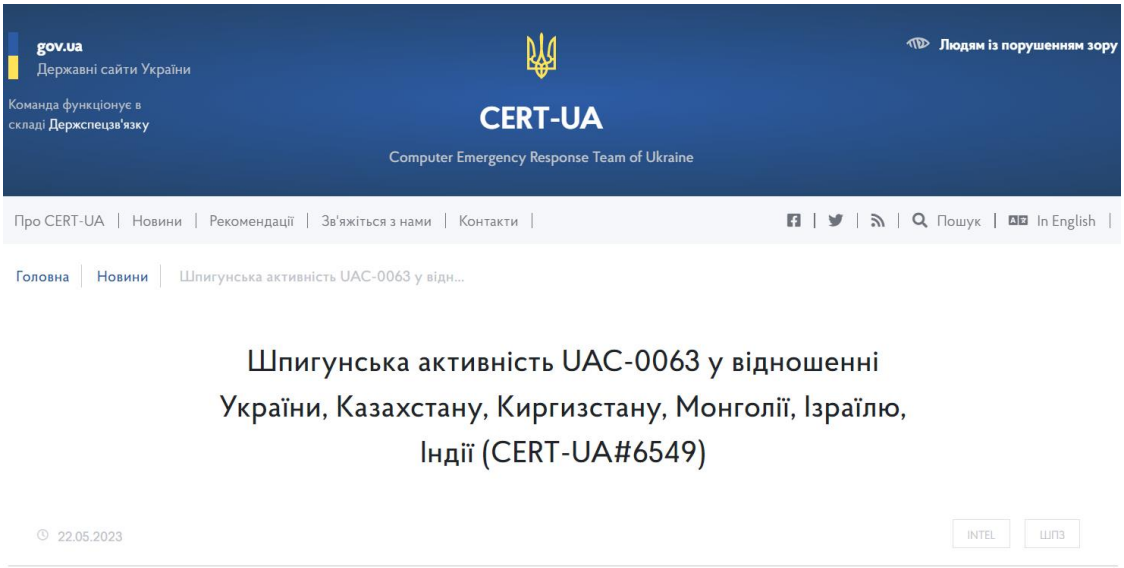


Рисунок 3.1 – Стаття з CERT-UA

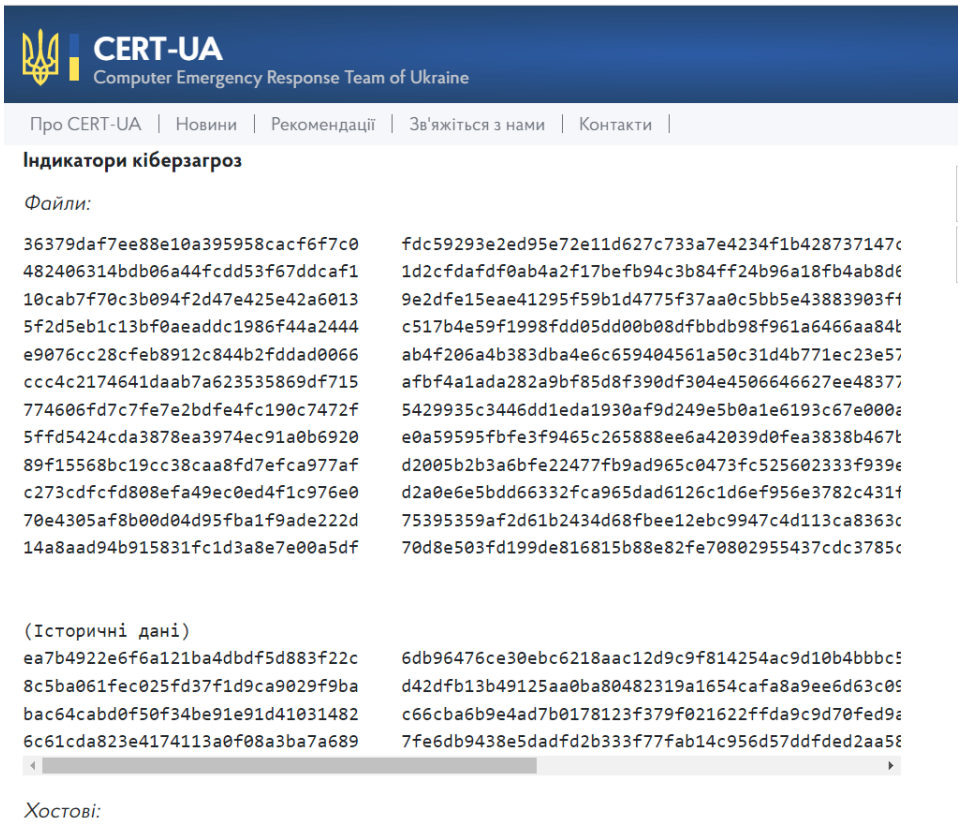


Рисунок 3.2 – Індикатори компрометації з CERT-UA

Іншим відкритим веб-ресурсом, де можна збирати індикатори є Alien Vault. AlienVault, як провайдер рішень з безпеки, також пропонує свої власні індикатори компрометації, які можна використовувати для виявлення потенційно шкідливої або несправедливої активності. На рисунках 3.3 та 3.4 зображено спосіб представлення IOC на цьому веб-ресурсі.

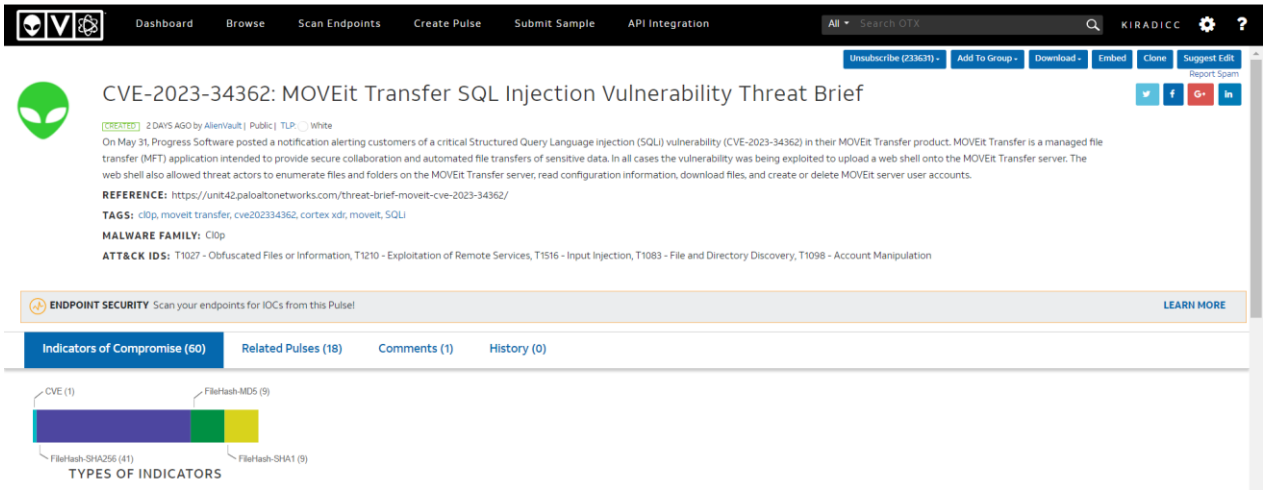
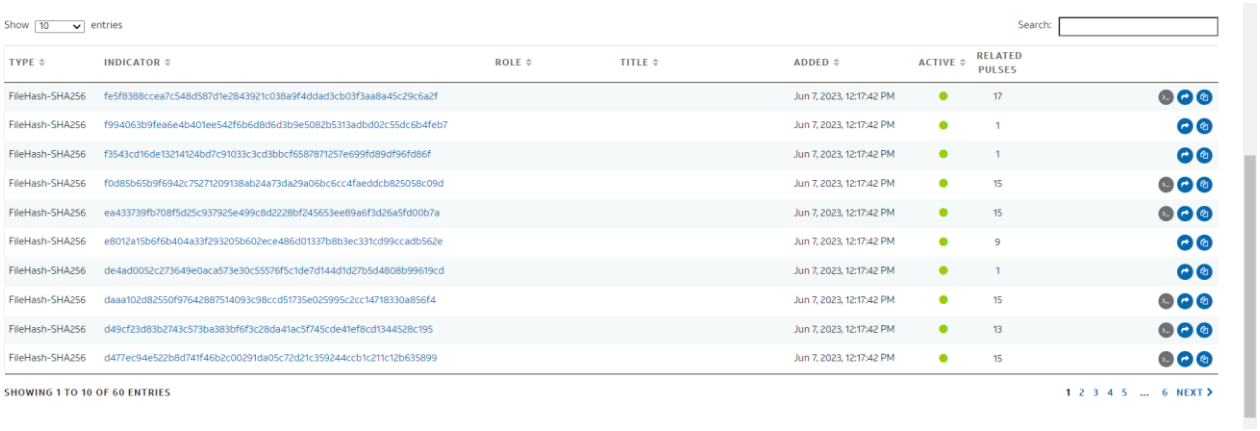


Рисунок 3.3 – Опис CVE-2023-34362 на Alien Vault



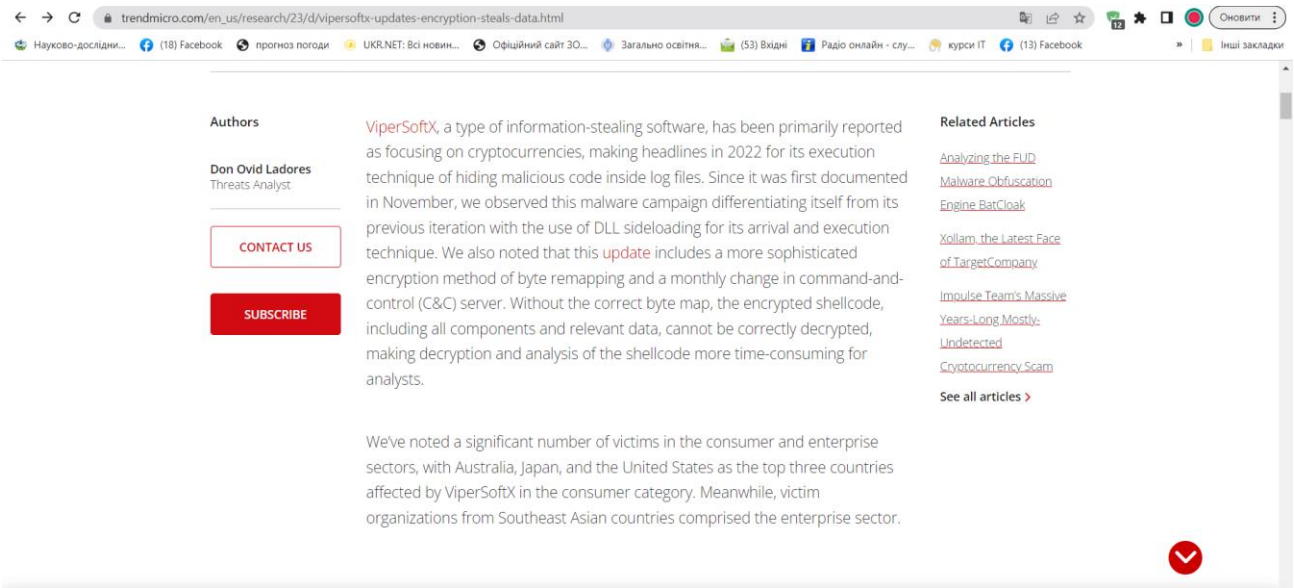
Show 10 entries Search:

TYPE	INDICATOR	ROLE	TITLE	ADDED	ACTIVE	RELATED PULSES
FileHash-SHA256	fe5f8388ccea7c548d587d1e2843921c038a9f4ddad3cb03f3aa8a45c29c6a2f			Jun 7, 2023, 12:17:42 PM		17
FileHash-SHA256	f994063b9fea6e4b401ee542f6b6d8d6d3b9e5082b5313adbd02c55dc6b4feb7			Jun 7, 2023, 12:17:42 PM		1
FileHash-SHA256	f3543cd16de13214124bd7c91033c3cd3bbcf6587871257e699fd89df96fd86f			Jun 7, 2023, 12:17:42 PM		1
FileHash-SHA256	f0d85b65b9f6942c75271209138ab24a73da29a06bc6cc4faeddcb825058c09d			Jun 7, 2023, 12:17:42 PM		15
FileHash-SHA256	ea433739fb708f5d25c937925e499c8d2228bf245653ee89a6f3d26a5fd00b7a			Jun 7, 2023, 12:17:42 PM		15
FileHash-SHA256	e8012a15b6f6b404a33f293205b602ece486d01337b8b3ec331cd99ccadb562e			Jun 7, 2023, 12:17:42 PM		9
FileHash-SHA256	de4ad0052c273649e0aca573e30c55576f5c1de7d144d1d27b5d4808b99619cd			Jun 7, 2023, 12:17:42 PM		1
FileHash-SHA256	daaa102d82550f97642887514093c98ccd51735e025995c2cc14718330a856f4			Jun 7, 2023, 12:17:42 PM		15
FileHash-SHA256	d49cf23d83b2743c573ba383bf6f3c28da41ac5f745cde41ef8cd1344528c195			Jun 7, 2023, 12:17:42 PM		13
FileHash-SHA256	d477ec94e522b8d741f46b2c00291da05c72d21c359244ccb1c211c12b635899			Jun 7, 2023, 12:17:42 PM		15

SHOWING 1 TO 10 OF 60 ENTRIES 1 2 3 4 5 ... 6 NEXT >

Рисунок 3.4 – Індикатори компрометації, які відносяться до CVE-2023-34362

Trend Micro є відомим постачальником рішень з кібербезпеки, і їх продукти також включають індикатори компрометації (IOC), які використовуються для виявлення можливих загроз і компрометації в комп'ютерних системах. На рисунках 3.5 та 3.6 показано в якому вигляді відображаються індикатори компрометації на веб-ресурсі Trend Micro.



trendmicro.com/en_us/research/23/d/vipersoftx-updates-encryption-steals-data.html

Authors

Don Ovid Ladores
Threats Analyst

CONTACT US

SUBSCRIBE

ViperSoftX, a type of information-stealing software, has been primarily reported as focusing on cryptocurrencies, making headlines in 2022 for its execution technique of hiding malicious code inside log files. Since it was first documented in November, we observed this malware campaign differentiating itself from its previous iteration with the use of DLL sideloading for its arrival and execution technique. We also noted that this update includes a more sophisticated encryption method of byte remapping and a monthly change in command-and-control (C&C) server. Without the correct byte map, the encrypted shellcode, including all components and relevant data, cannot be correctly decrypted, making decryption and analysis of the shellcode more time-consuming for analysts.

We've noted a significant number of victims in the consumer and enterprise sectors, with Australia, Japan, and the United States as the top three countries affected by ViperSoftX in the consumer category. Meanwhile, victim organizations from Southeast Asian countries comprised the enterprise sector.

Related Articles

Analyzing the FUD Malware Obfuscation Engine BatCloak

Xollam, the Latest Face of TargetCompany

Impulse Team's Massive Years-Long Mostly-Undetected Cryptocurrency Scam

See all articles >

Рисунок 3.5 – Опис зловмисного ПЗ

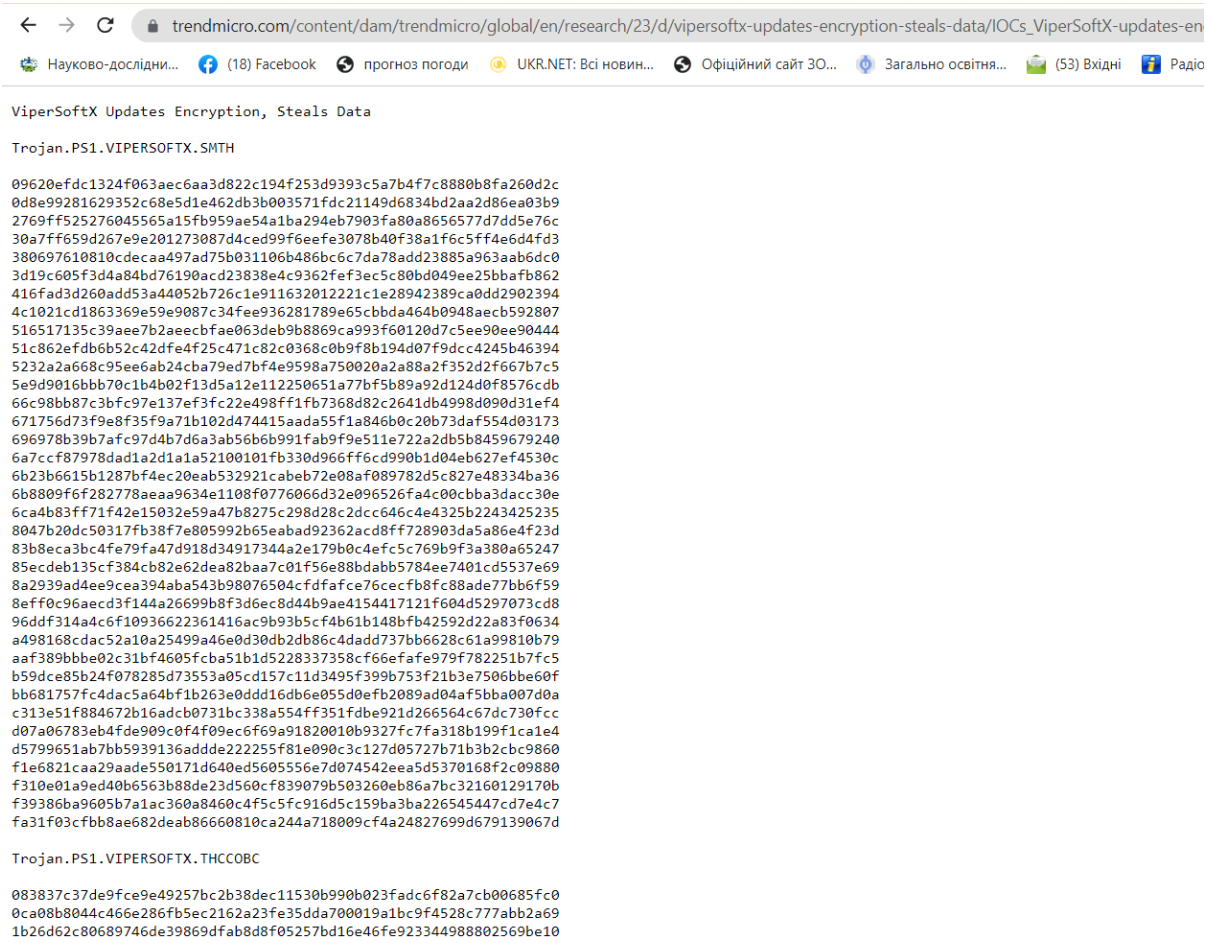


Рисунок 3.6 – Індикатори компрометації, які характеризують це ПЗ

Це лише декілька прикладів відкритих джерел, з яких IOC можуть бути зібрані.

Пошук індикаторів компрометації на доступних веб-ресурсах є важливим кроком для виявлення несанкціонованого доступу до систем і мереж. Комбінація ручного пошуку, використання автоматизованих інструментів та спеціалізованих систем може покращити ефективність процесу пошуку. Слід також враховувати, що цей процес повинен бути постійним, оскільки нові загрози та індикатори компрометації з'являються регулярно.

3.2 Налаштування інфраструктури MISP

Автоматизація збору індикаторів компрометації може значно спростити цей процес, оскільки фахівцю з кібезбезпеки не потрібно тратити багато часу на відвідування доступних веб-ресурсів і збір всіх індикаторів вручну.

Саме тому, для реалізації третього розділу було обрано MISP як платформу для обміну інформацією про загрози та компрометацію. Він надає можливість спільного використання індикаторів компрометації, включаючи IP-адреси, хеш-суми файлів, URL-адреси тощо, з іншими організаціями.

А також Wazuh як потужний інструментом для моніторингу та виявлення загроз в реальному часі. Його здатність аналізувати логи, мережевий трафік та інші дані дозволяє швидко виявляти незвичайну або підозрілу активність в системі.

Для забезпечення максимальної доступності середовища, в котрому будуть зберігатись всі знайдені індикатори компрометації оптимальним рішенням є розмістити його на певній хмарній платформі. Для цього є кількість різноманітних варіантів, котрі пропонують наразі різні вендори, але в даному випадку було обрано саме сервіси від Amazon.

Концептуально мережева інфраструктура буде мати топологію, зображену на рисунку 3.8:

На рисунку 3.8 показано, що було створено віртуальне приватне хмарне середовище (VPC) і розміщено в ньому інфраструктури. VPC є віртуальною мережею, що надає ізольоване та приватне середовище для розгортання різних ресурсів, таких як сервери, бази даних та інші компоненти інфраструктури.

Створення VPC дозволяє забезпечити контроль, безпеку та ізоляцію ресурсів в рамках віртуальної мережі. Вона може бути розгорнута в хмарних платформах, таких як Amazon Web Services (AWS), Microsoft Azure або Google Cloud Platform (GCP).

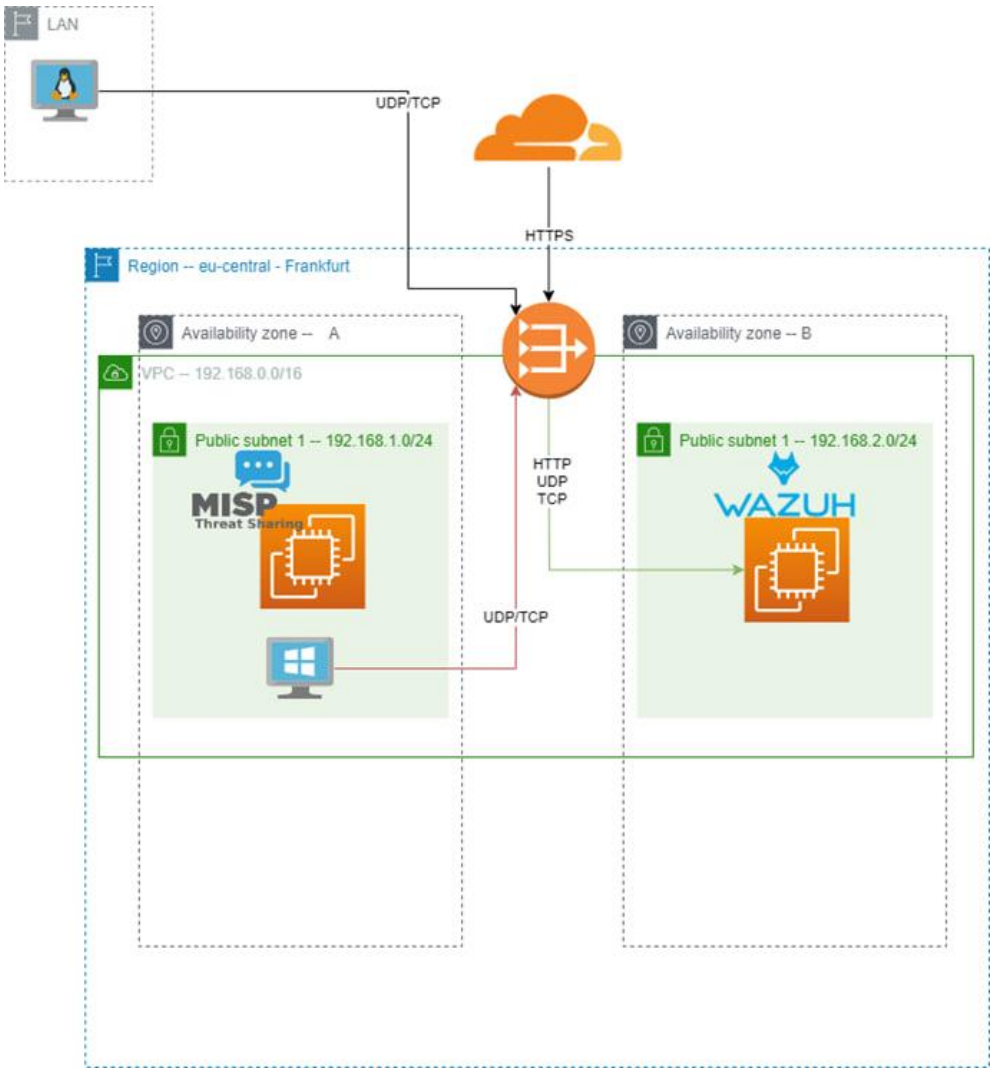


Рисунок 3.7 – Загальна топологія мережі в хмарі

У цьому віртуальному приватному хмарному середовищі була розміщена дана інфраструктура, що включає різні компоненти, які можуть включати в себе сервери, мережеві пристрої, сховища даних та інші ресурси, необхідні для функціонування системи чи додатку.

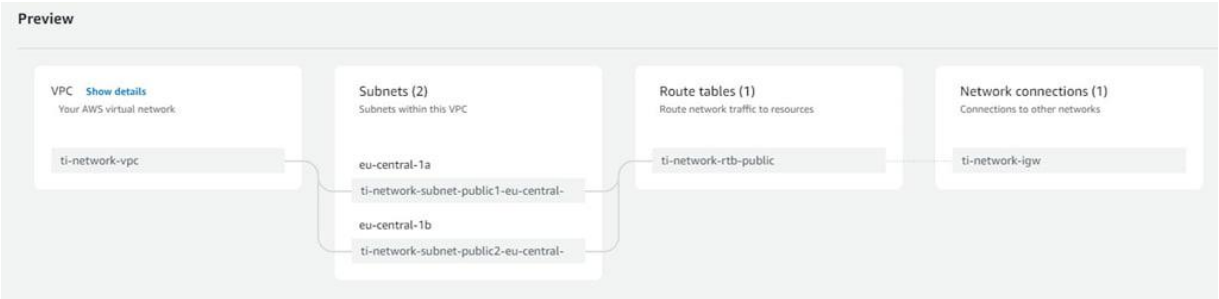


Рисунок 3.8 – Віртуальна приватна хмара, в якій розміщено інфраструктуру

На рисунку 3.9 зображено створення віртуального сервера у публічній підмережі в рамках VPC. Це означає, що цей сервер може бути доступний з Інтернету за допомогою публічно доступної IP-адреси.

У контексті віртуальної приватної хмари (VPC), публічна підмережа - це мережевий сегмент, до якого можна отримати доступ з Інтернету. Вона призначена для розміщення ресурсів, які потребують публічного доступу, таких як веб-сервери, публічні API або інші зовнішні ресурси. Ці ресурси можуть бути доступні з інтернету шляхом публічно доступної IP-адреси.

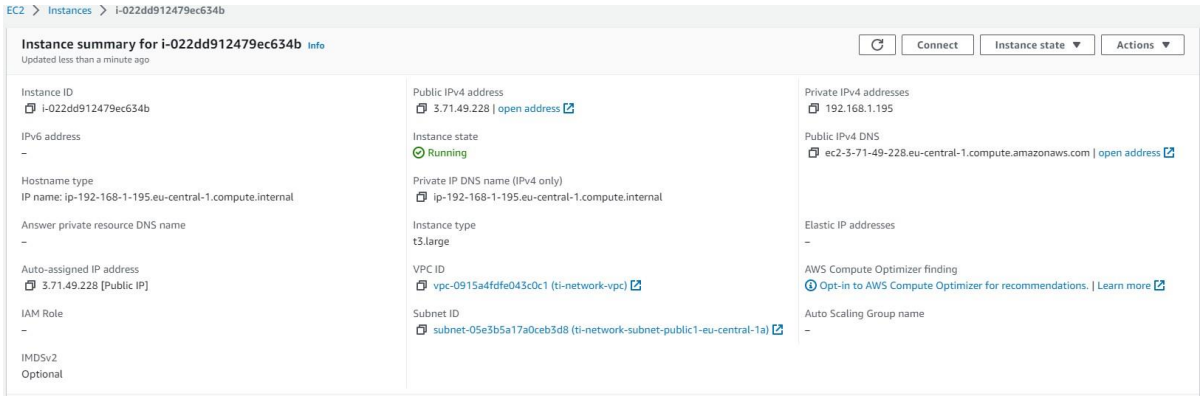


Рисунок3.9 – Віртуальний сервер

На сервері було успішно встановлено та запущено програму MISP. MISP (Malware Information Sharing Platform) є платформою для обміну інформацією про шкідливе програмне забезпечення та інші кіберзагрози між організаціями та співробітниками, які займаються кібербезпекою (рис. 3.10).

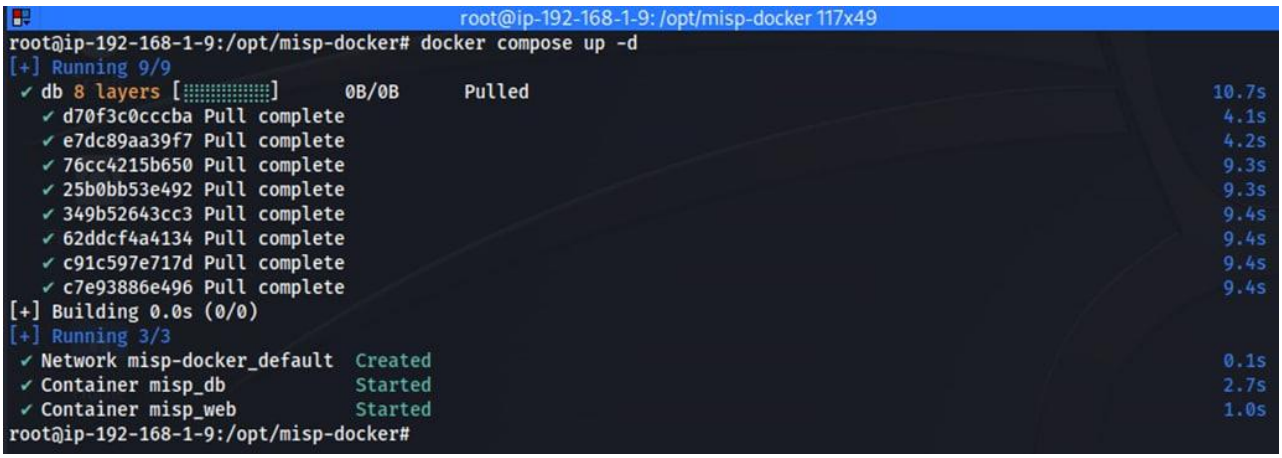


Рисунок 3.10 – Встановлення MISP

Був налаштований сервіс MISP (Malware Information Sharing Platform), який тепер доступний з будь-якого місця у світі, за умови, що користувач має відповідні авторизаційні дані та підключення до Інтернету(рис. 3.11).

Рисунок 3.11 – Сервіс MISP

Наступним кроком було налаштовано вхідні правила, які дозволяють тільки HTTPS трафік, який надходить виключно від серверів CloudFlare (Рис. 3.12).

Inbound rules (16)

Manage tags | Edit inbound rules

Filter security group rules

Security group rule...	IP version	Type	Protocol	Port range	Source	Description
sgr-0b0636931a46e6bf3	IPv4	HTTPS	TCP	443	173.245.48.0/20	–
sgr-00235d39c8c53b2...	IPv4	HTTPS	TCP	443	103.21.244.0/22	–
sgr-0050afbe06ca37a18	IPv4	HTTPS	TCP	443	103.22.200.0/22	–
sgr-03e074ef6ac4b7115	IPv4	HTTPS	TCP	443	190.93.240.0/20	–
sgr-0c063817b3f681593	IPv4	HTTPS	TCP	443	104.24.0.0/14	–
sgr-043fc7cb378c06143	IPv4	HTTPS	TCP	443	188.114.96.0/20	–
sgr-0ccd1844379941c38	IPv4	HTTPS	TCP	443	172.64.0.0/13	–
sgr-0d1397df2a44c0d7c	IPv4	HTTPS	TCP	443	131.0.72.0/22	–
sgr-057bf94b1b7b399...	IPv4	HTTPS	TCP	443	197.234.240.0/22	–
sgr-093c14eacb65c330e	IPv4	HTTPS	TCP	443	162.158.0.0/15	–
sgr-022d123e446a90...	IPv4	HTTPS	TCP	443	103.31.4.0/22	–
sgr-034cc954db11c6510	IPv4	HTTPS	TCP	443	141.101.64.0/18	–
sgr-0e6904226c81f8a95	IPv4	SSH	TCP	22	94.232.214.191/32	–
sgr-0ab25faa4b88b2605	IPv4	HTTPS	TCP	443	104.16.0.0/13	–
sgr-03a6b1a7847cabb...	IPv4	HTTPS	TCP	443	108.162.192.0/18	–
sgr-0595cd79dcccb42da	IPv4	HTTPS	TCP	443	198.41.128.0/17	–

Рисунок 3.12 - HTTPS трафік

Рисунок 3.13 описує реалізацію автоматичного збору IOC з різних джерел, які було прописано в конфігураційному json-файлі [20].

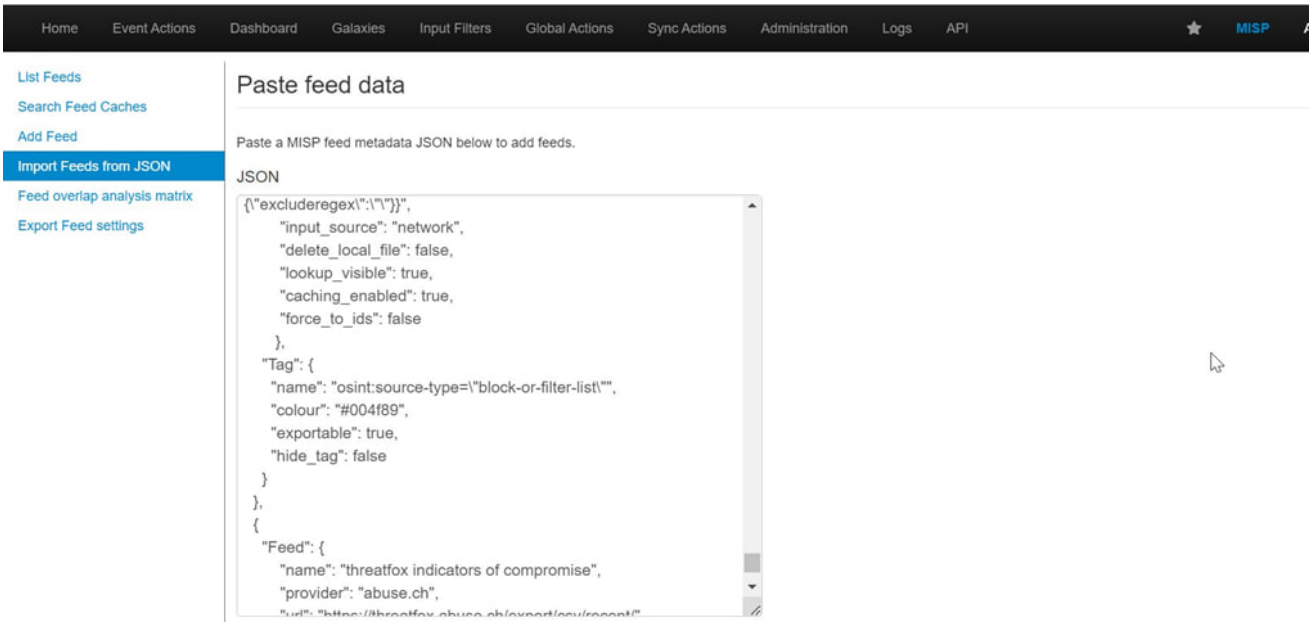


Рисунок 3.13 – Реалізація автоматичного збору IOC

Після збору індикаторів компрометації з різних джерел, необхідно провести завантаження всіх цих даних в базу даних як це зображено на рисунку 3.14 та 3.15

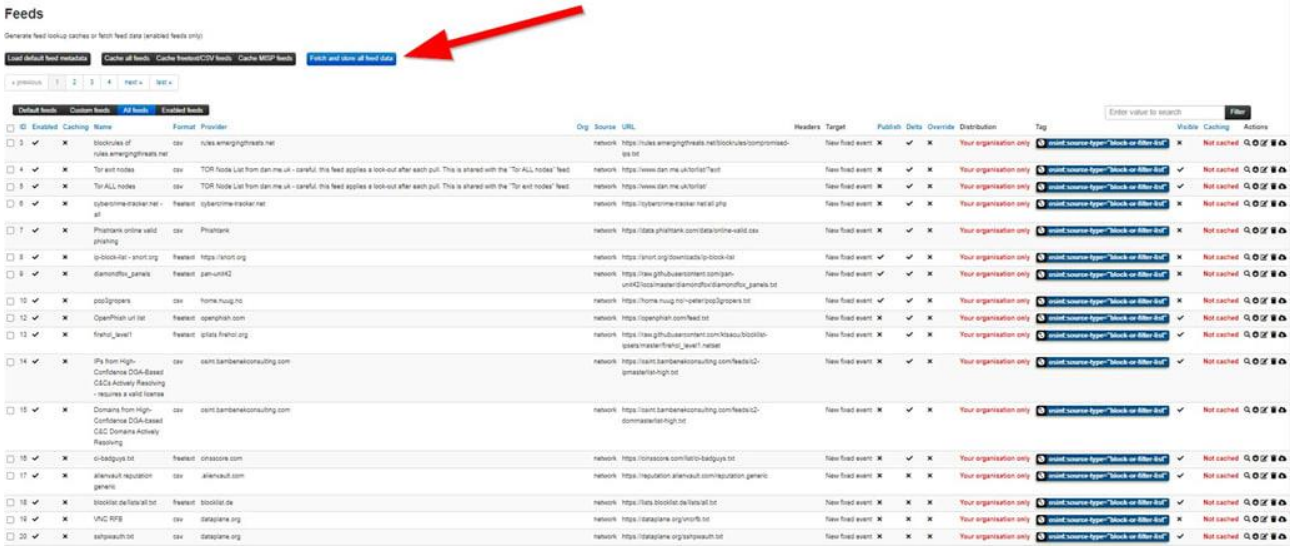
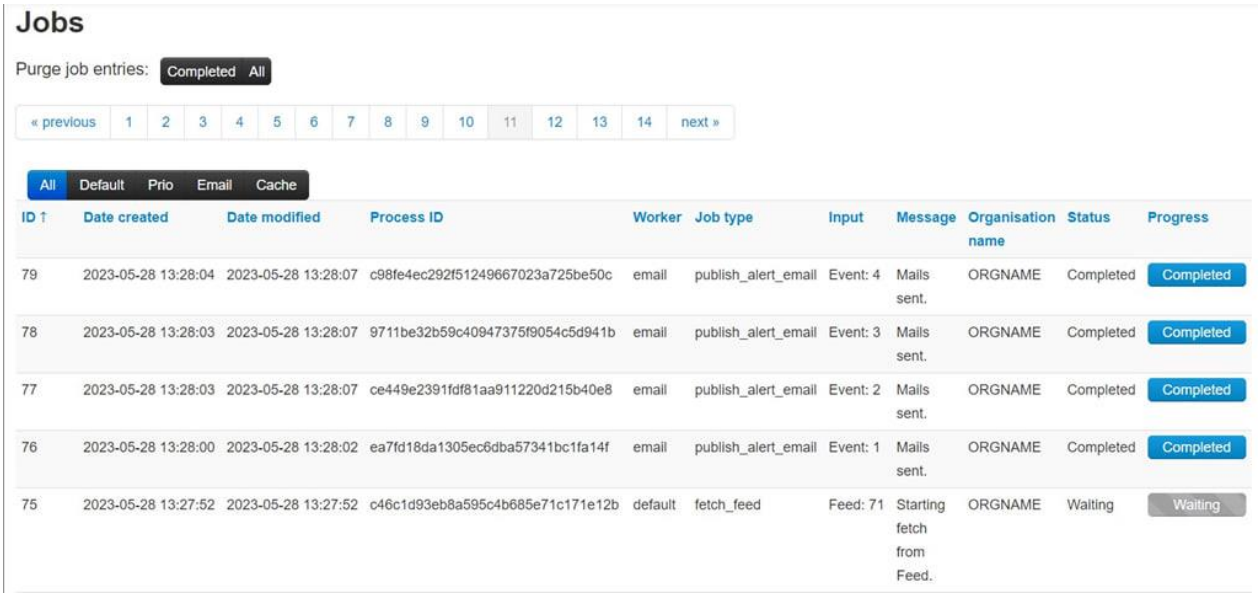


Рисунок 3.14 – Процес завантаження IOC в базу даних



Jobs

Purge job entries: Completed All

« previous 1 2 3 4 5 6 7 8 9 10 11 12 13 14 next »

All Default Prio Email Cache

ID ↑	Date created	Date modified	Process ID	Worker	Job type	Input	Message	Organisation name	Status	Progress
79	2023-05-28 13:28:04	2023-05-28 13:28:07	c98fe4ec292f51249667023a725be50c	email	publish_alert_email	Event: 4	Mails sent.	ORGNAME	Completed	Completed
78	2023-05-28 13:28:03	2023-05-28 13:28:07	9711be32b59c40947375f9054c5d941b	email	publish_alert_email	Event: 3	Mails sent.	ORGNAME	Completed	Completed
77	2023-05-28 13:28:03	2023-05-28 13:28:07	ce449e2391fdf81aa911220d215b40e8	email	publish_alert_email	Event: 2	Mails sent.	ORGNAME	Completed	Completed
76	2023-05-28 13:28:00	2023-05-28 13:28:02	ea7fd18da1305ec6dba57341bc1fa14f	email	publish_alert_email	Event: 1	Mails sent.	ORGNAME	Completed	Completed
75	2023-05-28 13:27:52	2023-05-28 13:27:52	c46c1d93eb8a595c4b685e71c171e12b	default	fetch_feed	Feed: 71	Starting fetch from Feed.	ORGNAME	Waiting	Waiting

Рисунок 3.15 – Процес завантаження IOC в базу даних

3.3 Налаштування системи Wazuh

Система Wazuh була успішно встановлена та налаштована для використання. Це зображено на Рисунку 3.16.

Wazuh є відкритою платформою для розпізнавання загроз, моніторингу безпеки та виявлення вторгнень в реальному часі. Вона надає функціональність зібрання, аналізу та реагування на дані з різних джерел, таких як журнали подій, системні файли, мережевий трафік та інші джерела інформації.

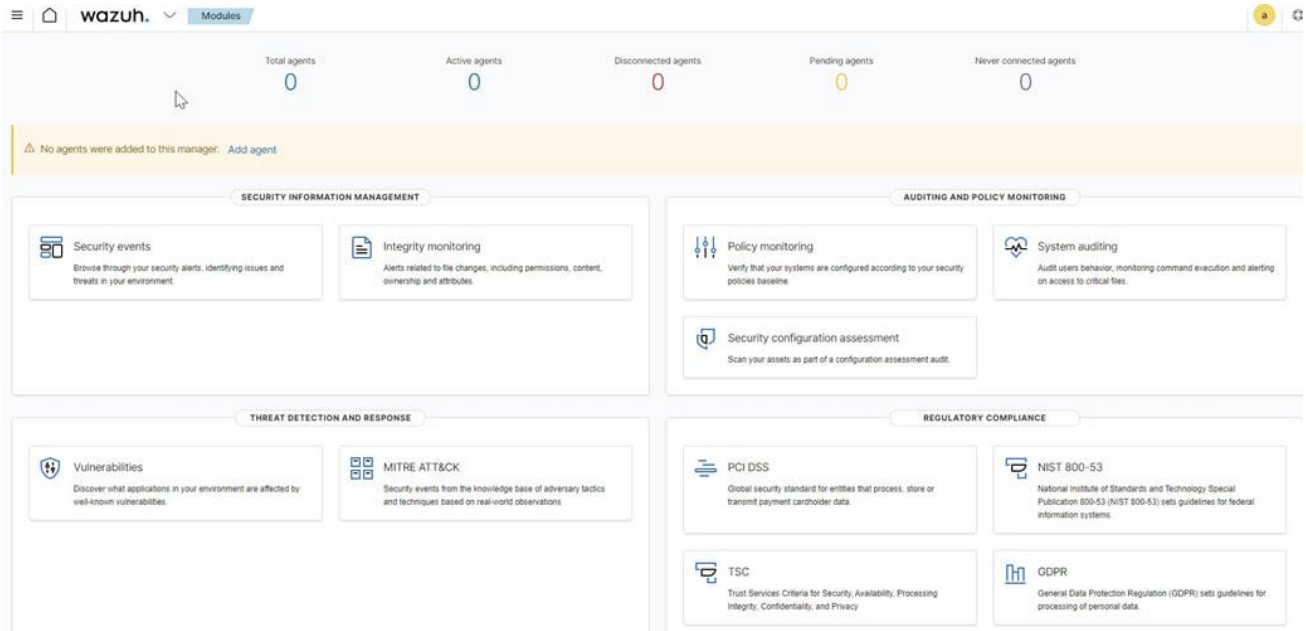


Рисунок 3.16 - Система Wazuh

Наступним кроком було заведено тестову операційну систему Ubuntu 22.04, для того щоб збирати з неї події, які будуть відображатись в Wazuh. Для цього потрібно налаштувати агент, шляхом генерації посилання в консолі, як це зображено на Рисунку 3.17 та 3.18.

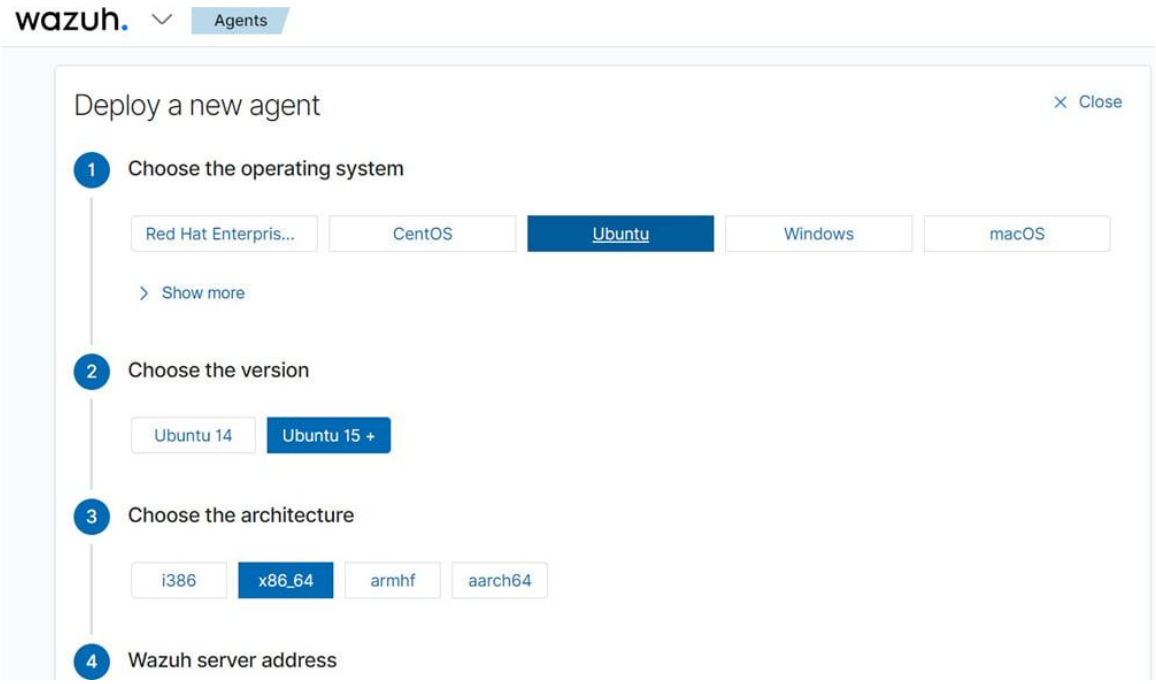


Рисунок 3.17 – Вибір ОС для генерації агента

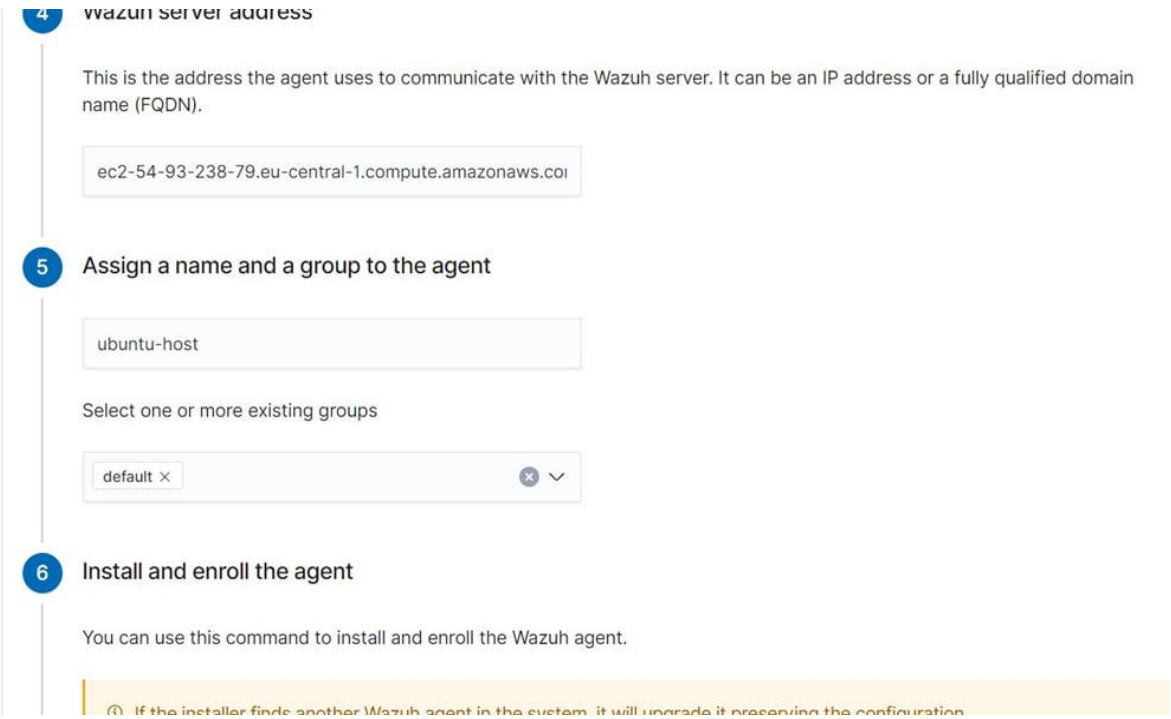


Рисунок 3.18 – Генерація агента

Далі генерується команда, яка зображена на Рисунку 3.19. Її необхідно виконати на хості, для того щоб встановити агент Wazuh:

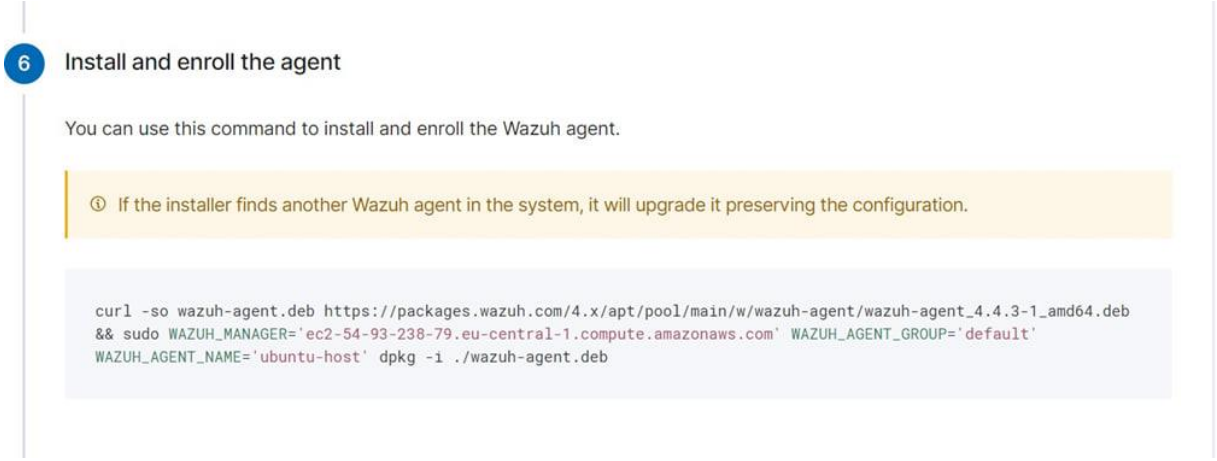


Рисунок 3.19 – Команда для встановлення агента Wazuh

Виконавши цю команду на машині (Рис. 3.20), відбувається успішне встановлення агента Wazuh.

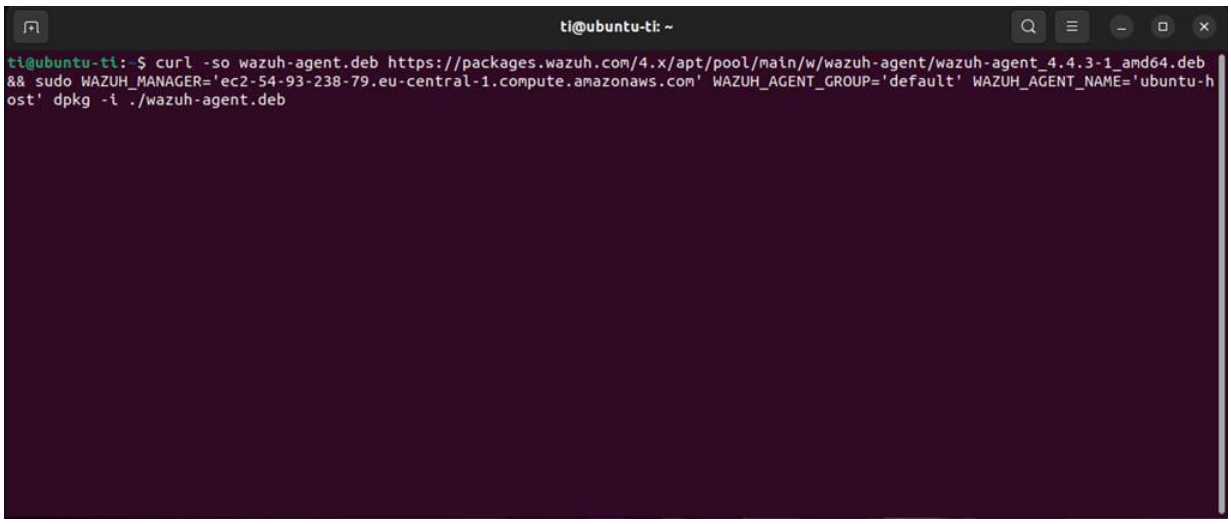


Рисунок 3.20 – Виконання команди, щоб встановити Wazuh

Отже, на Рисунку 3.21 зображено, що агент успішно встановлено та запущено.

```
Selecting previously unselected package wazuh-agent.
(Reading database ... 198476 files and directories currently installed.)
Preparing to unpack ./wazuh-agent.deb ...
Unpacking wazuh-agent (4.4.3-1) ...
Setting up wazuh-agent (4.4.3-1) ...
ti@ubuntu-ti:~$ sudo systemctl daemon-reload
sudo systemctl enable wazuh-agent
sudo systemctl start wazuh-agent
Synchronizing state of wazuh-agent.service with SysV service script with /lib/systemd/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install enable wazuh-agent
Created symlink /etc/systemd/system/multi-user.target.wants/wazuh-agent.service → /lib/systemd/system/wazuh-agent.service.
ti@ubuntu-ti:~$
```

Рисунок 3.20 – Успішне запущення агента

Wazuh успішно встановив зв'язок з цим хостом та отримує логи з нього (рис. 22). На рисунку 3.21 показано, що в консолі Wazuh можна бачити відомості про цей хост, такі як його ідентифікатор, IP-адресу, назву, а також інші характеристики.

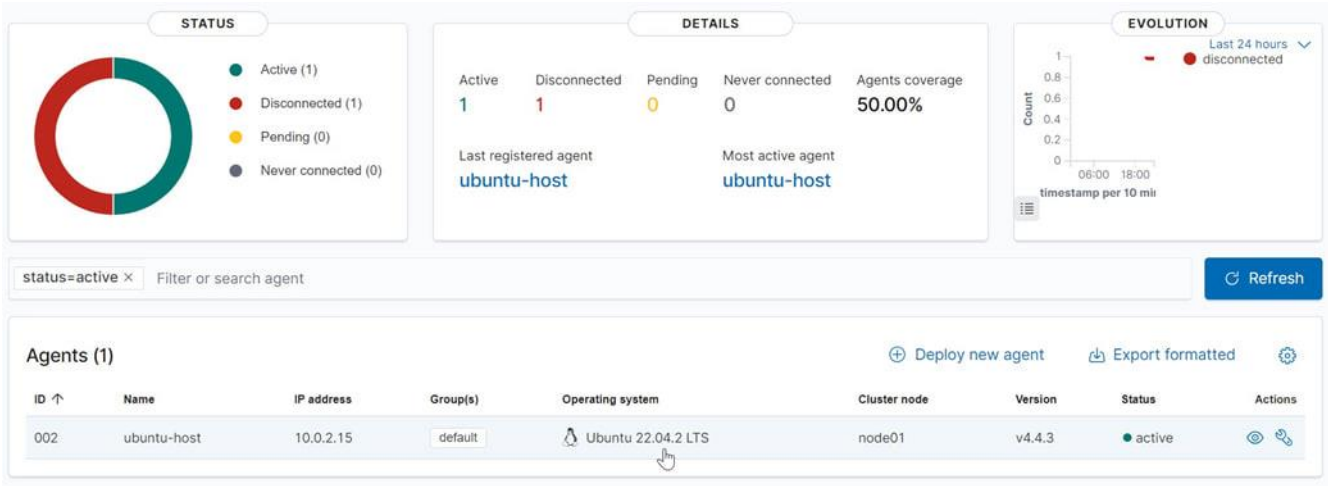


Рисунок 3.21 – Консоль Wazuh з хостом

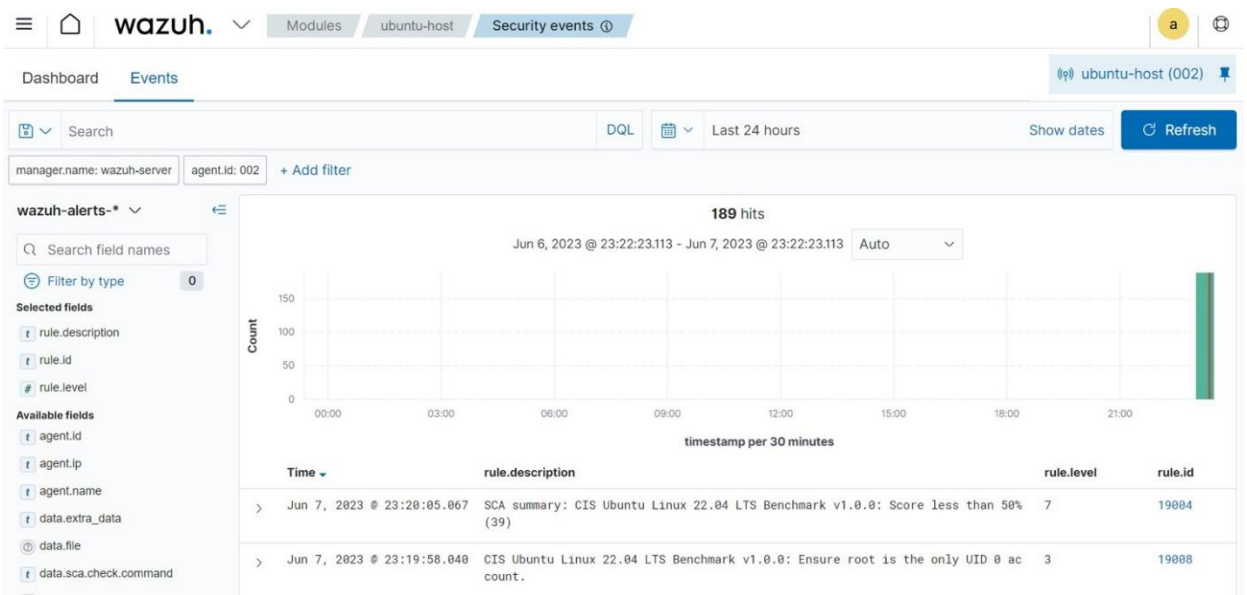


Рисунок 3.22 – Події з хоста

Для того, щоб перевірити працездатність було згенеровано декілька подій, які зображені на рисунку 3.23

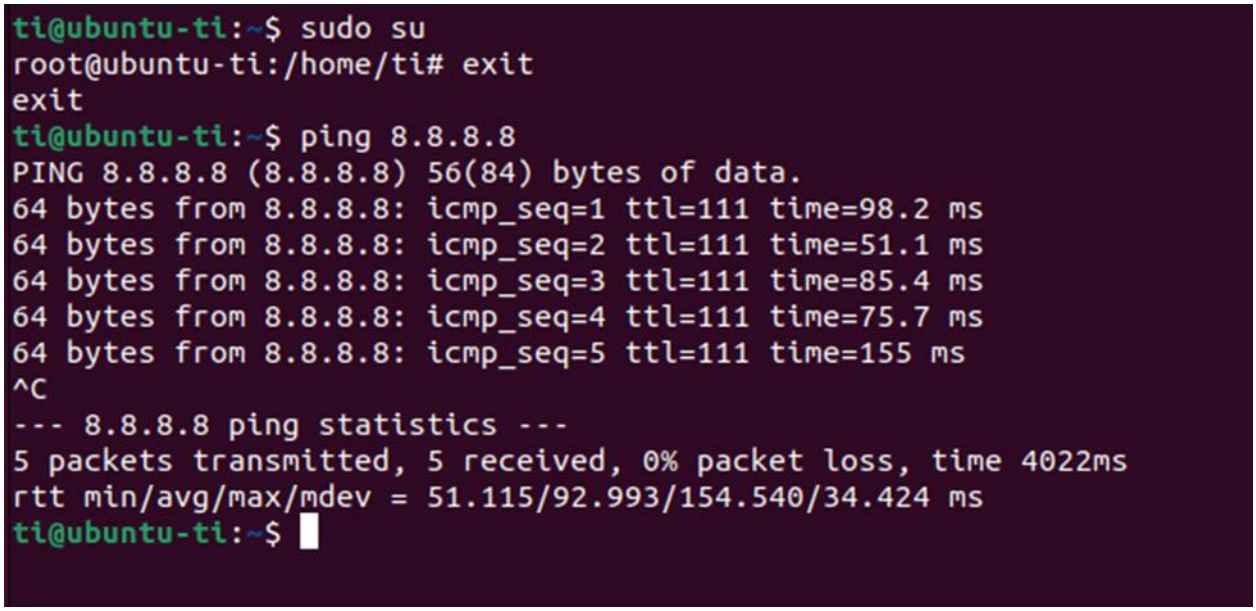
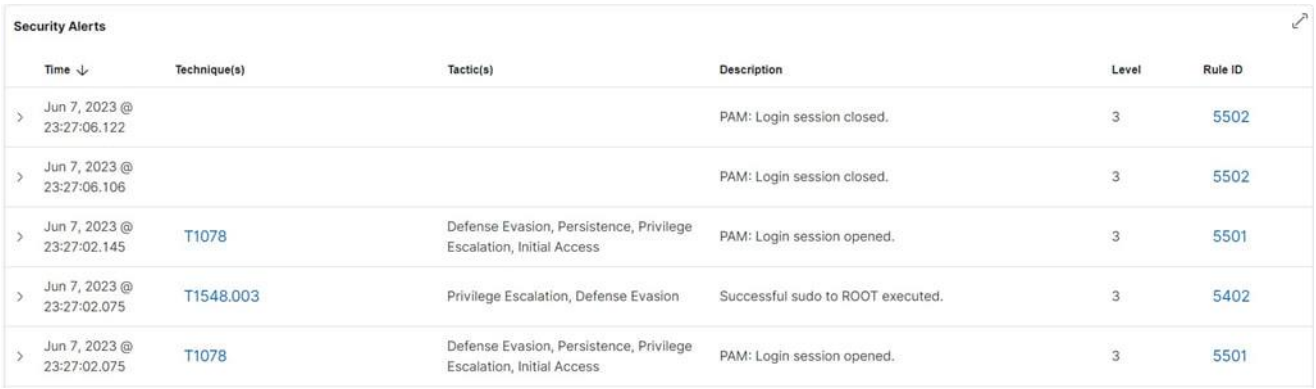


Рисунок 3.23 – Генерація випадкових подій, щоб перевірити працездатність

Як результат, логи оперативно збираються в Wazuh та накладаються на техніки та тактики MITRE ATT&CK Matrix. На рисунку 3.24 показано як цей процес відображається в Wazuh.



Security Alerts

Time	Technique(s)	Tactic(s)	Description	Level	Rule ID
Jun 7, 2023 @ 23:27:06.122			PAM: Login session closed.	3	5502
Jun 7, 2023 @ 23:27:06.106			PAM: Login session closed.	3	5502
Jun 7, 2023 @ 23:27:02.145	T1078	Defense Evasion, Persistence, Privilege Escalation, Initial Access	PAM: Login session opened.	3	5501
Jun 7, 2023 @ 23:27:02.075	T1548.003	Privilege Escalation, Defense Evasion	Successful sudo to ROOT executed.	3	5402
Jun 7, 2023 @ 23:27:02.075	T1078	Defense Evasion, Persistence, Privilege Escalation, Initial Access	PAM: Login session opened.	3	5501

Рисунок 3.24 – Події з хоста

Для зручності інтеграції та збору логів в системі Wazuh на платформі Linux було встановлено Sysmon, як показано на Рисунках 3.25 та 3.26.

Sysmon є інструментом моніторингу системи, розробленим для платформи Windows. Однак, існує модифікована версія Sysmon, яка була адаптована для використання на платформі Linux [21]. Встановлення Sysmon для Linux дозволяє отримати додаткову функціональність і можливості збору логів з Linux-систем.

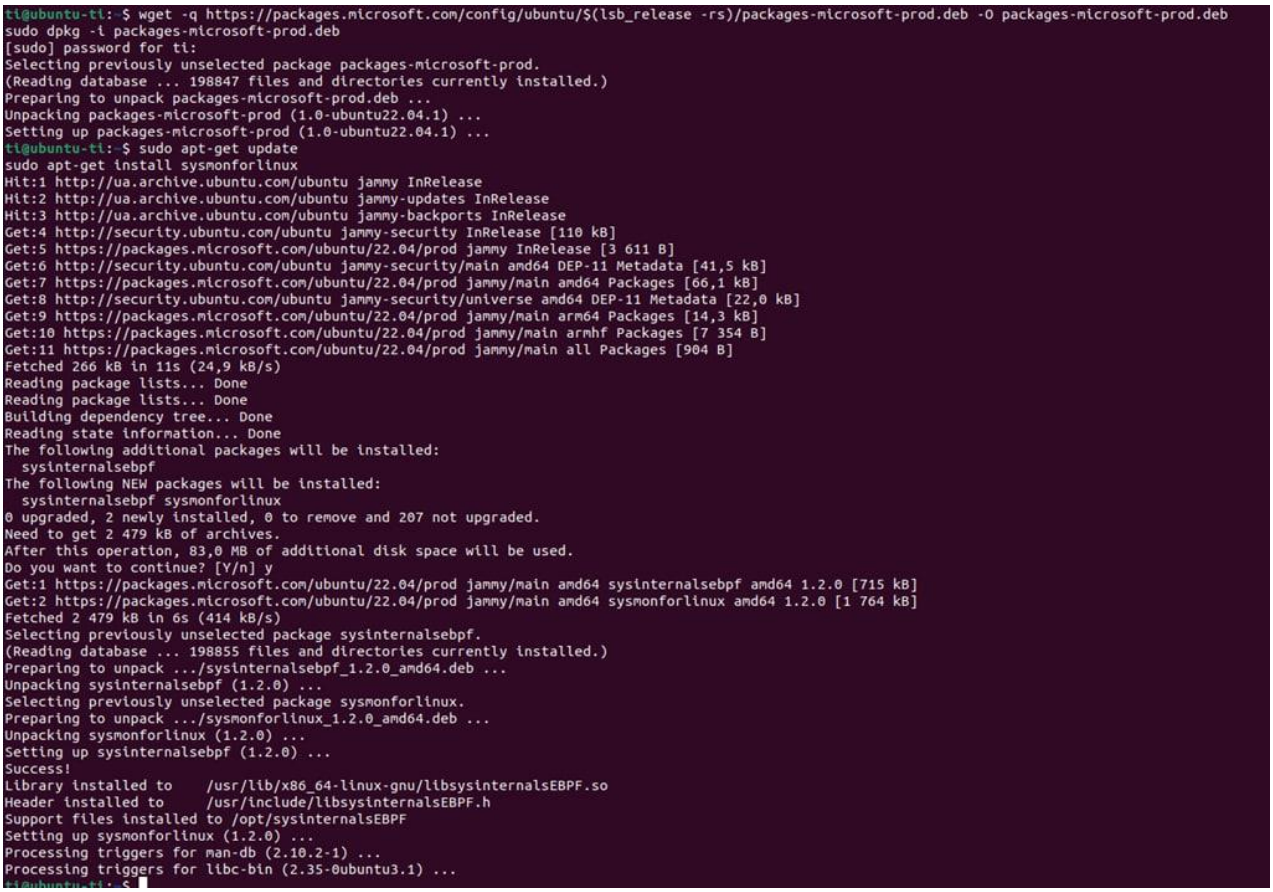


Рисунок 3.25 – Процес встановлення Sysmon

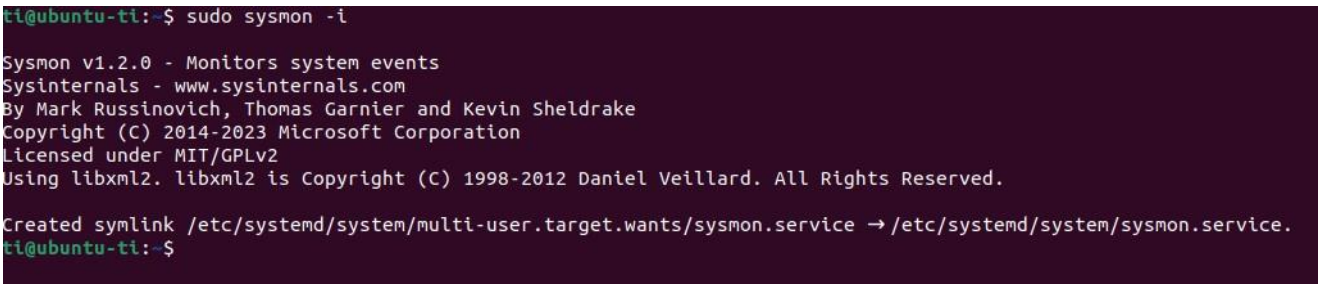


Рисунок 3.26 - Процес встановлення Sysmon

XML логи Sysmon мають наступний вигляд (Рис. 3.27).

Рисунок 3.27 - XML логи Sysmon

Системи Wazuh має обмеження щодо обробки XML-логів, зібраних з Sysmon. Щоб забезпечити здатність Wazuh до аналізу цих логів, необхідно налаштувати декодер [21] і правила для їх парсингу [22]. Це зображено на Рисунках 3.28 та 3.29.

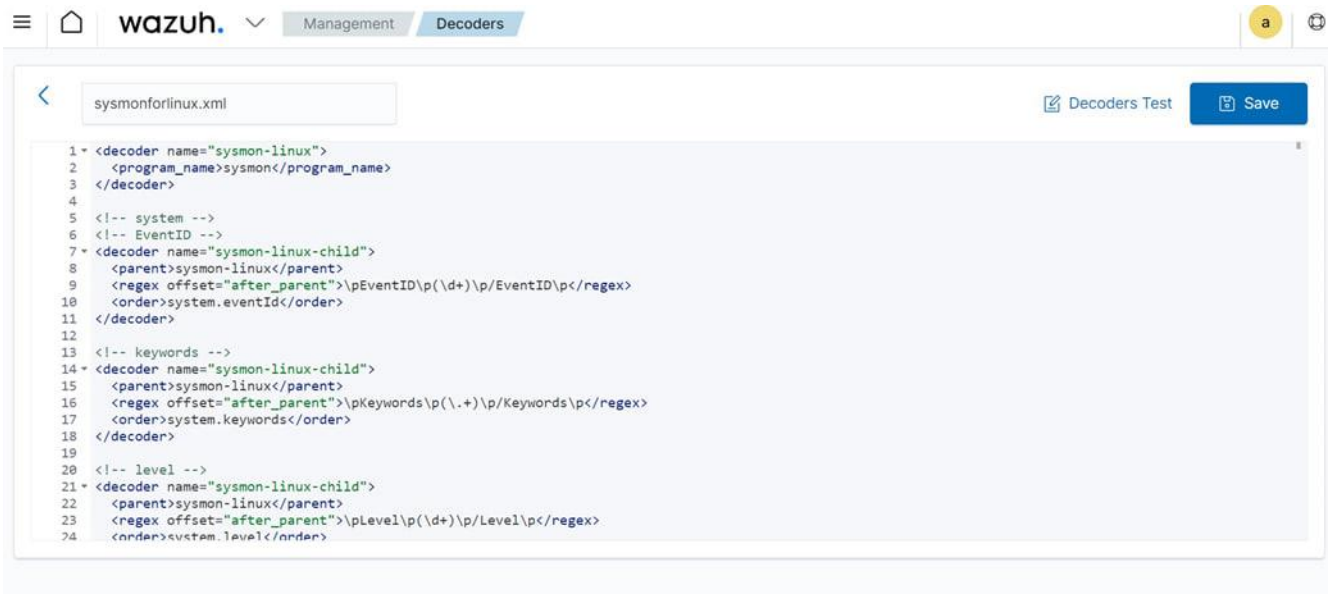


Рисунок 3.28 – Налаштування декодера та правил

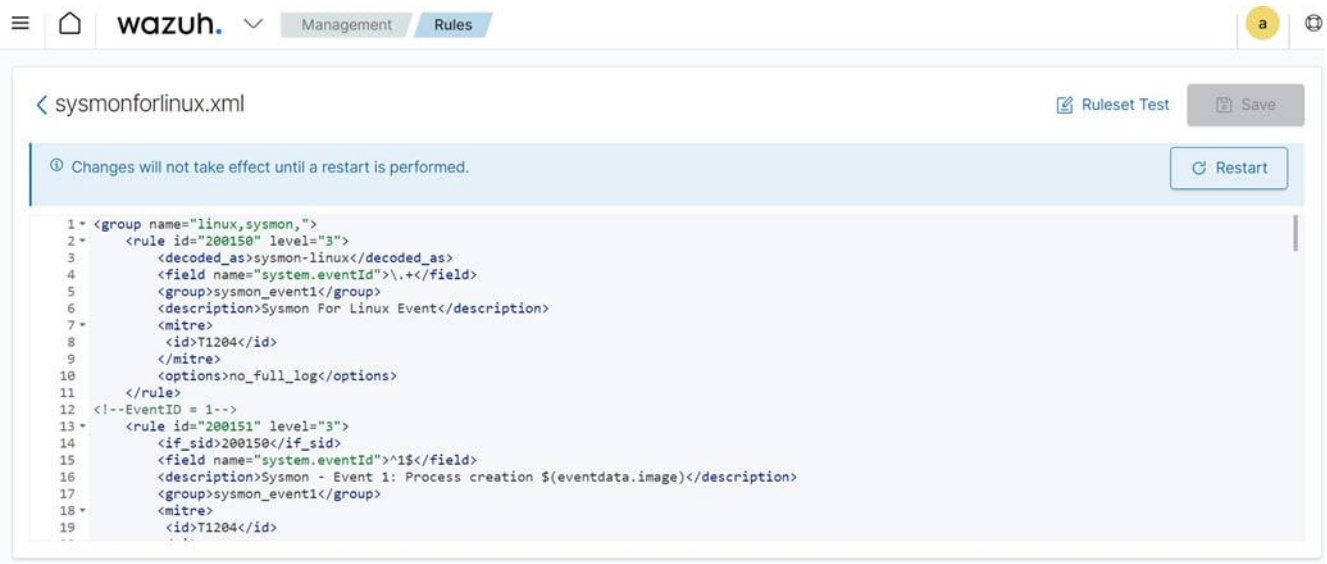


Рисунок 3.29 - Налаштування декодера та правил

Щоб переглянути працездатність попереднього кроку необхідно знову згенерувати декілька подій (рис. 3.30).

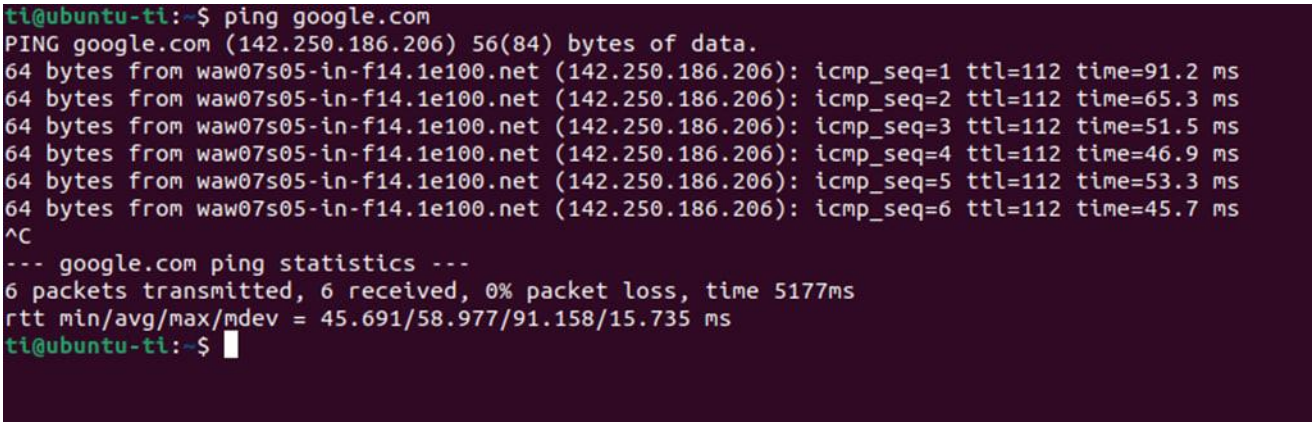


Рисунок 3.30 – Генерація подій

Після налаштування Sysmon на кінцевому хості і налаштування декодера та правил парсингу в Wazuh та генерації декількох подій можна перевірити, чи успішно збираються та відображаються логи Sysmon у консолі Wazuh. Результат показано на Рисунку 3.31.

Jun 8, 2023 @ 00:30:15.246	Sysmon - Event 5: Process terminated /usr/bin/ping	3	200153
Jun 8, 2023 @ 00:30:13.336	Sysmon - Event 5: Process terminated /usr/bin/dash	3	200153
Jun 8, 2023 @ 00:30:13.336	Sysmon - Event 5: Process terminated /usr/bin/last	3	200153
Jun 8, 2023 @ 00:30:13.286	Sysmon - Event 1: Process creation /usr/bin/last	3	200151
Jun 8, 2023 @ 00:30:13.268	Sysmon - Event 1: Process creation /usr/bin/dash	3	200151
Jun 8, 2023 @ 00:30:13.242	Sysmon - Event 5: Process terminated /usr/bin/dash	3	200153
Jun 8, 2023 @ 00:30:13.223	Sysmon - Event 5: Process terminated /usr/bin/sed	3	200153
Jun 8, 2023 @ 00:30:13.221	Sysmon - Event 5: Process terminated /usr/bin/sed	3	200153
Jun 8, 2023 @ 00:30:13.221	Sysmon - Event 5: Process terminated /usr/bin/sed	3	200153
Jun 8, 2023 @ 00:30:13.220	Sysmon - Event 1: Process creation /usr/bin/sed	3	200151
Jun 8, 2023 @ 00:30:13.220	Sysmon - Event 1: Process creation /usr/bin/sed	3	200151

Рисунок 3.31 – Відображення логів після налаштування декодера та правил парсингу

Далі необхідно внести зміни в конфігураційний файл Sysmon (Рис. 3.32) для того, щоб система Wazuh збирала додаткові події, окрім тих, які вже включені (події 1 і 5).

```
ti@ubuntu-ti:~$ nano sys.xml
ti@ubuntu-ti:~$ sudo sysmon -c sys.xml
[sudo] password for ti:

Sysmon v1.2.0 - Monitors system events
Sysinternals - www.sysinternals.com
By Mark Russinovich, Thomas Garnier and Kevin Sheldrake
Copyright (C) 2014-2023 Microsoft Corporation
Licensed under MIT/GPLv2
Using libxml2. libxml2 is Copyright (C) 1998-2012 Daniel Veillard. All Rights Re
served.

Loading configuration file with schema version 4.70
Sysmon schema version: 4.81
Configuration file validated.
ti@ubuntu-ti:~$
```

Рисунок 3.32 – Коригування конфігураційного файлу Sysmon

Код редагування конфігу [23]:

```
<Sysmon schemaversion="4.70">
<EventFiltering>
<!-- Event ID 1 == ProcessCreate. Log all newly created processes -->
<RuleGroup name="" groupRelation="or">
<ProcessCreate onmatch="exclude"/>
</RuleGroup>
```

```
<!-- Event ID 3 == NetworkConnect Detected. Log all network connections -->
<RuleGroup name="" groupRelation="or">
<NetworkConnect onmatch="exclude"/>
</RuleGroup>
<!-- Event ID 5 == ProcessTerminate. Log all processes terminated -->
<RuleGroup name="" groupRelation="or">
<ProcessTerminate onmatch="exclude"/>
</RuleGroup>
<!-- Event ID 9 == RawAccessRead. Log all raw access read -->
<RuleGroup name="" groupRelation="or">
<RawAccessRead onmatch="exclude"/>
</RuleGroup>
<!-- Event ID 10 == ProcessAccess. Log all open process operations -->
<RuleGroup name="" groupRelation="or">
<ProcessAccess onmatch="exclude"/>
</RuleGroup>
<!-- Event ID 11 == FileCreate. Log every file creation -->
<RuleGroup name="" groupRelation="or">
<FileCreate onmatch="exclude"/>
</RuleGroup>
<!--Event ID 23 == FileDelete. Log all files being deleted -->
<RuleGroup name="" groupRelation="or">
<FileDelete onmatch="exclude"/>
</RuleGroup>
</EventFiltering>
</Sysmon>
```

Після збереження змін у конфігураційному файлі Sysmon і його перезавантаження на кінцевому хості, система Wazuh здатна збирати додаткові події, які були вказані у відредагованому конфігураційному файлі Sysmon. Це дозволить розширити обсяг інформації, яку Wazuh може збирати та аналізувати з кінцевого хоста. Результат зображено на Рисунку 3.33.

Jun 8, 2023 @ 08:42:01.097	Sysmon - Event 11: FileCreate by /usr/bin/tar	3	200155
Jun 8, 2023 @ 08:42:01.096	Sysmon - Event 23: FileDelete (A file delete was detected) by /usr/bin/tar	3	200157
Jun 8, 2023 @ 08:42:01.095	Sysmon - Event 11: FileCreate by /usr/bin/tar	3	200155
Jun 8, 2023 @ 08:42:01.095	Sysmon - Event 11: FileCreate by /usr/bin/tar	3	200155
Jun 8, 2023 @ 08:42:01.076	Sysmon - Event 11: FileCreate by /usr/bin/tar	3	200155
Jun 8, 2023 @ 08:42:01.057	Sysmon - Event 11: FileCreate by /usr/bin/tar	3	200155

Рисунок 3.33 – Відображення додаткових подій в консолі

Тепер події, які можуть бути корисними під час threat intelligence збираються за допомогою Sysmon. Наприклад, EventID 3 вказує на подію, пов'язану з мережевим з'єднанням (Network Connection).

Цей тип події може надати цінну інформацію про мережеві активності, такі як встановлення або закриття з'єднань, адреси IP, порти, протоколи і т. д. Він може бути важливим для виявлення аномальної мережевої активності, наприклад, злочинної діяльності, злому або несанкціонованого доступу.

Процес перевірки працездатності зображено на рисунку 3.34 та 3.35.

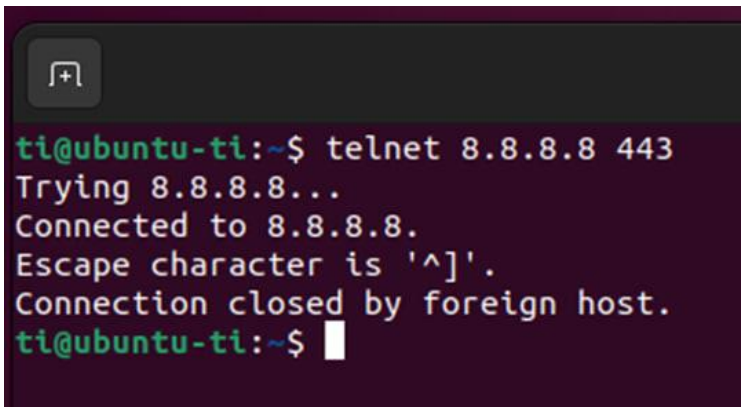


Рисунок 3.34 – Встановлення telnet з’єднання

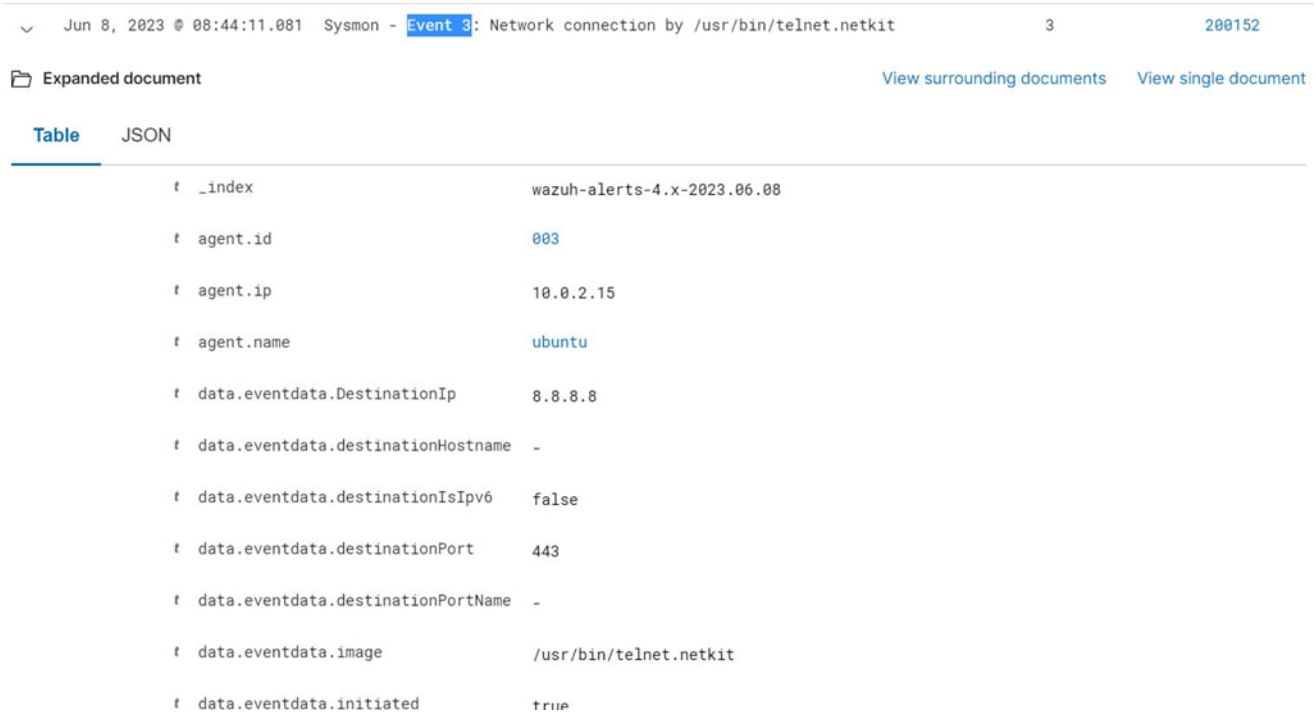


Рисунок 3.35 – Перевірка події в консолі Wazuh

Отже, це надає можливість відслідковувати мережеві з’єднання хоста, що частково нівелює необхідність використання обов’язкового проксі для кінцевих хостів (Рис. 3.36).

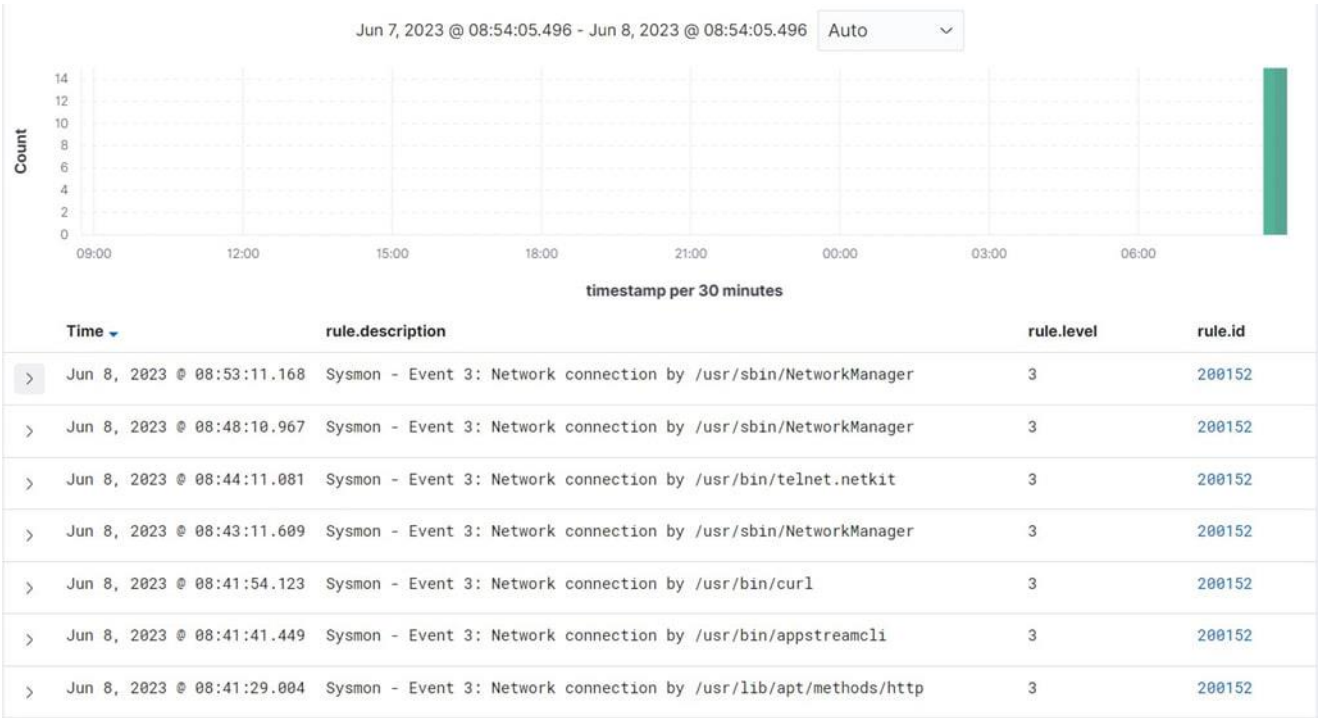


Рисунок 3.36 – Логи з Sysmon в Wazuh

Тепер необхідно провести інтеграцію, для того щоб Wazuh робив запити в MISP, на предмет присутності індикаторів в його базі даних. Для початку створюється токен, по якому Wazuh зможе робити авторизовані запити до MISP. Для цього необхідно дозволити запити лише з нашої хмарної підмережі (Рис. 3.37).

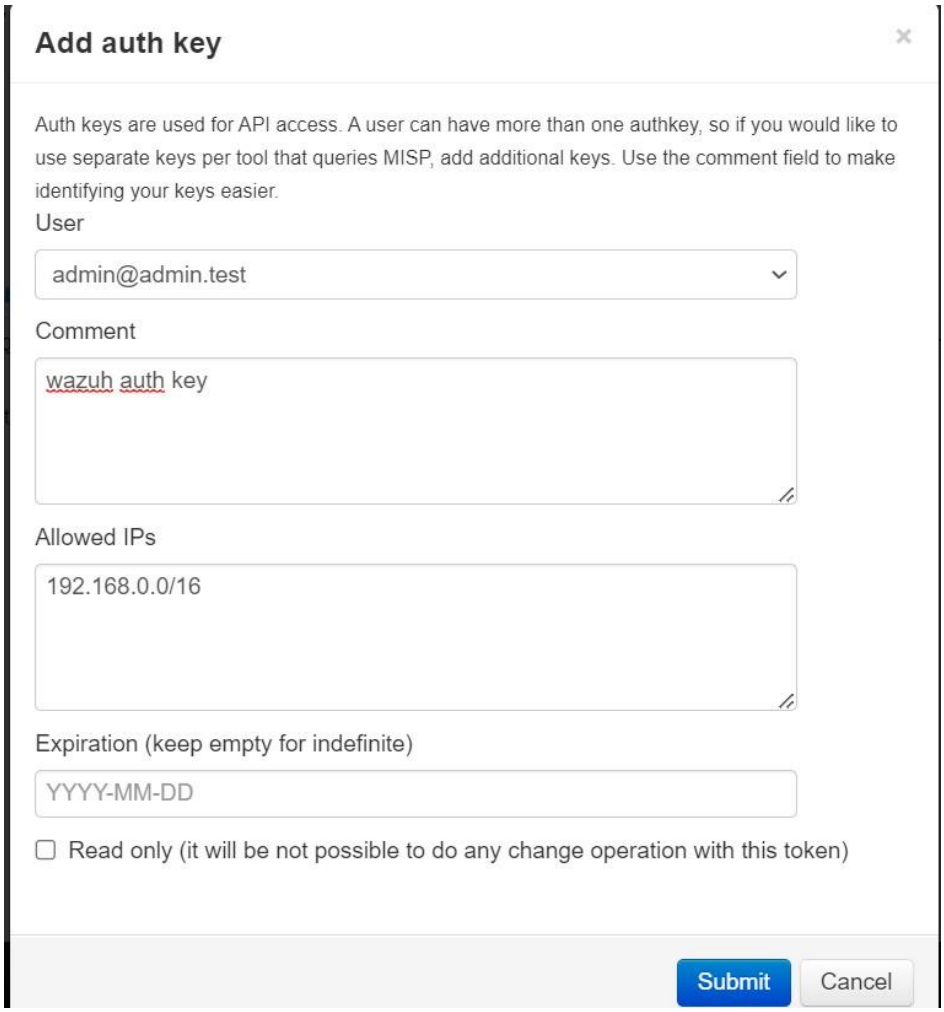


Рисунок 3.37 – Генерація токену для інтеграції з MISP

```
# MISP Server Base URL
misp_base_url = "https://192.168.1.9/attributes/restSearch/"
# MISP Server API AUTH KEY
misp_api_auth_key = "06K0rH365cikJLR8Z1HAwXbLIQUFzaBb14eisBp"
```

Рисунок 3.38 – Налаштування скрипта для інтеграції Wazuh з MISP

```
drwxr-x---  2 root wazuh  146 Jun  8 06:34 .
drwxr-x--- 19 root wazuh  242 May 25 01:57 ..
-rw-r--r--  1 root root  8417 Jun  8 06:34 custom-misp.py
-rwxr-x---  1 root wazuh 4325 May 24 20:05 pagerduty
-rwxr-x---  1 root wazuh 1045 May 24 20:05 shuffle
-rwxr-x---  1 root wazuh 4472 May 24 20:05 shuffle.py
-rwxr-x---  1 root wazuh 1045 May 24 20:05 slack
-rwxr-x---  1 root wazuh 3809 May 24 20:05 slack.py
-rwxr-x---  1 root wazuh 1045 May 24 20:05 virustotal
-rwxr-x---  1 root wazuh 6564 May 24 20:05 virustotal.py
[root@wazuh-server integrations]# chown root:wazuh custom-misp.py
[root@wazuh-server integrations]# chmod 750 custom-misp.py
[root@wazuh-server integrations]#
```

Рисунок 3.39 – Зміна прав доступу для інтеграційного скрипта з MISP

Далі процес налаштування інтеграційного блоку у Wazuh. Це зображено на Рисунку 3.40.

Блок інтеграції у Wazuh забезпечує можливість підключення та взаємодії з іншими системами, сервісами чи інструментами. Це дозволяє розширити можливості Wazuh та покращити інтеграцію з іншими компонентами інформаційної безпеки, в цьому випадку з MISP.

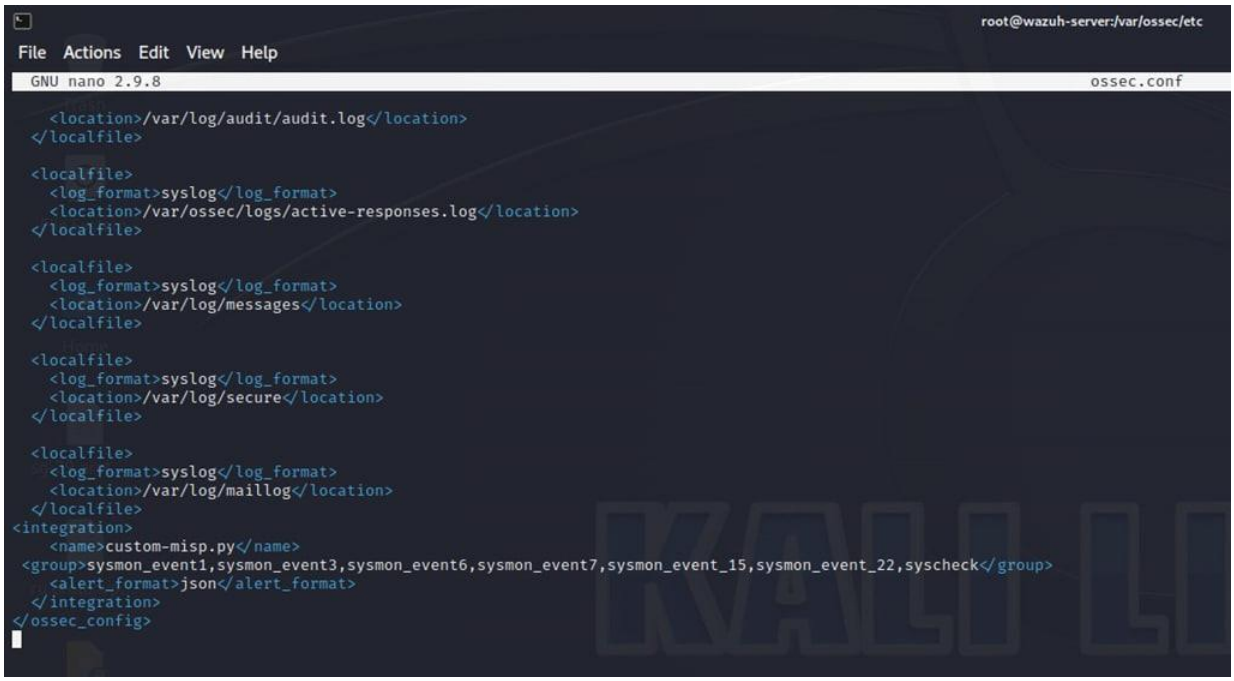


Рисунок 3.40 - Процес налаштування інтеграційного блоку [30]

А також необхідно додати нове правило (рис. 3.41)

Рисунок 3.41 – Додавання правила

```
<group name="misp,">
  <rule id="100620" level="10">
    <field name="integration">misp</field>
    <match>misp</match>
    <description>MISP Events</description>
    <options>no_full_log</options>
  </rule>
  <rule id="100621" level="5">
    <if_sid>100620</if_sid>
    <field name="misp.error">\.+</field>
    <description>MISP - Error connecting to API</description>
    <options>no_full_log</options>
    <group>misp_error,</group>
  </rule>
  <rule id="100622" level="12">
    <field name="misp.category">\.+</field>
<description>MISP - IoC found in Threat Intel - Category: $(misp.category),Attribute: $(misp.value)</description>
    <options>no_full_log</options>
    <group>misp_alert,</group>
  </rule>
</group>
```

Рисунок 3.42 – Правило для Wazuh[25]

Для кращої візуалізації, потрібно завести у Wazuh хост під керуванням Windows. Алгоритм подібний до того, який зображений на рисунках 3.17-3.19. Процес встановлення агента на Windows зображений на Рисунку 3.43.

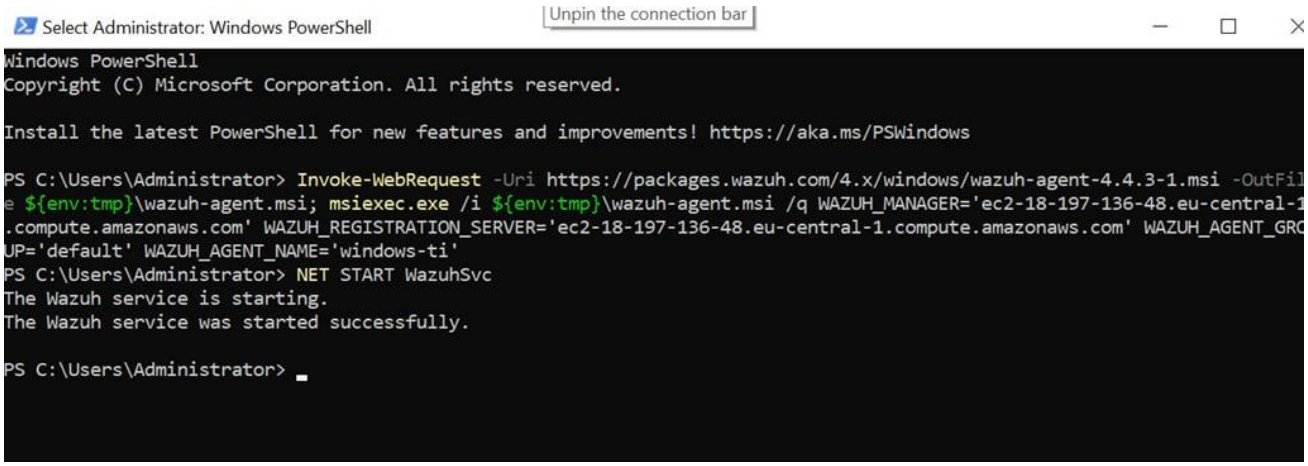


Рисунок 3.43 – Додавання агента на Windows

На рисунку 3.44 зображена консоль Wazuh, в якій відображається дві машини з якої збираються логи.

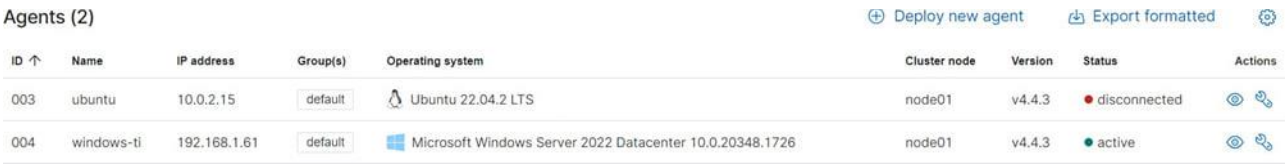


Рисунок 3.44 – Консоль Wazuh з двома машинами

Після встановлення агента Wazuh на хості і забезпечення з'єднання з сервером Wazuh, наступним кроком є налаштування збору даних з хоста за допомогою цього агента (Рис. 3.45).

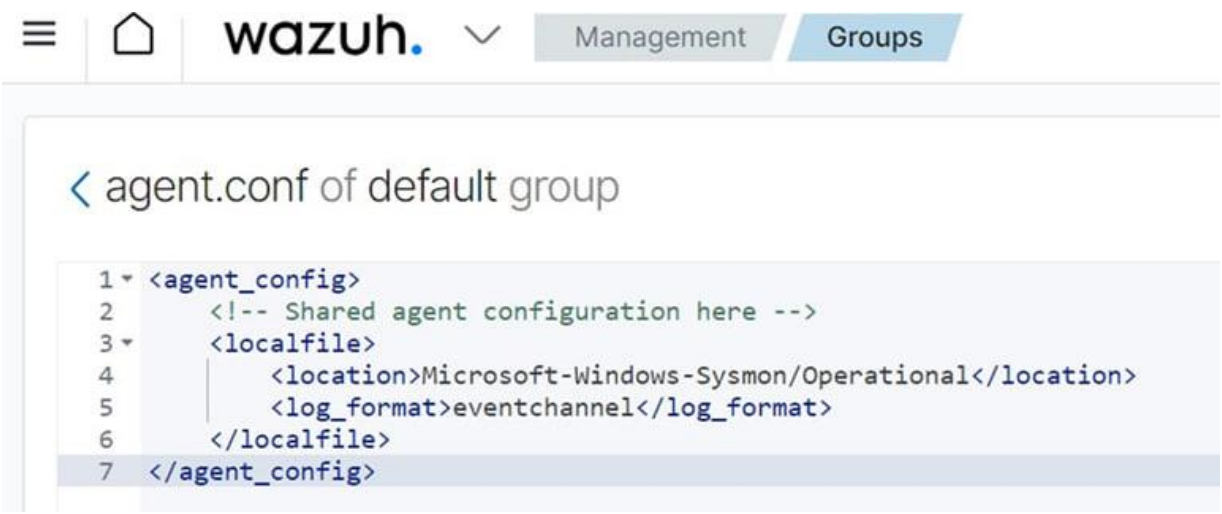


Рисунок 3.45 – Налаштування збору даних з хоста

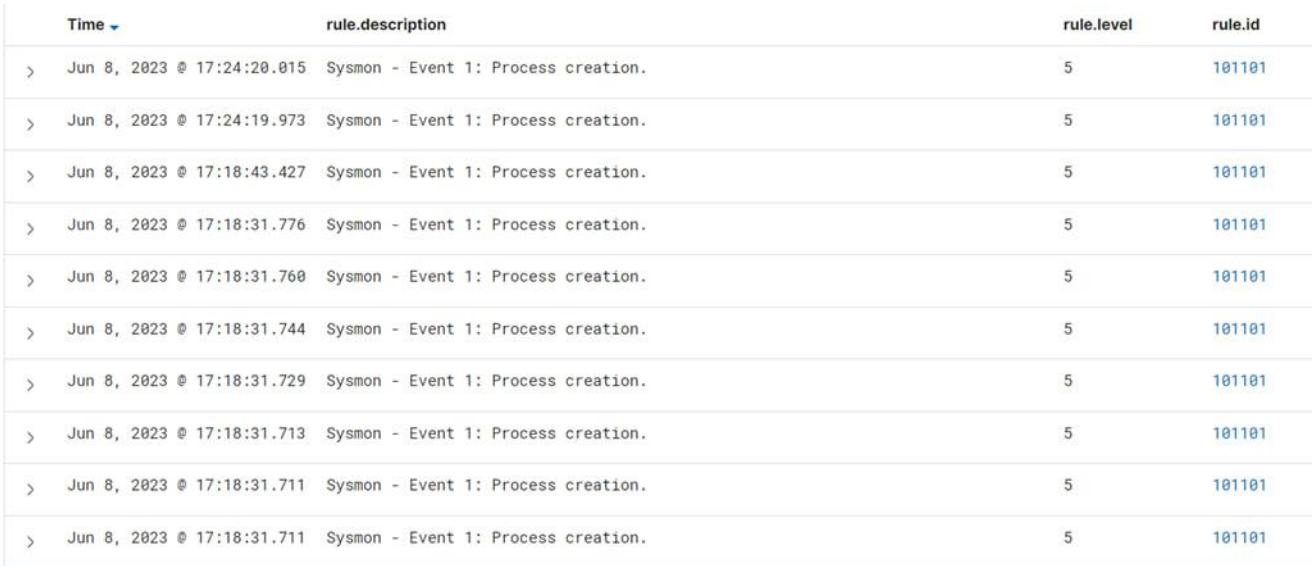
Далі налаштування збору даних за допомогою агента (рис. 3.46)

Рисунок 3.46 – Налаштування збору даних з хоста за допомогою агента

Наступним кроком необхідно додати спеціальне правило [26] для коректного парсингу даних Sysmon, що збираються з хостів під керуванням Windows (рис. 3.47). Це дозволить системі Wazuh розуміти та інтерпретувати ці дані правильно.

Рисунок 3.47 – Налаштування правила для коректного парсингу даних Sysmon

Після налаштування та внесення необхідних змін в систему Wazuh, вона успішно здійснює процес збору та парсингу логів Sysmon з хостів, що працюють під керуванням операційної системи Windows. На рисунку 3.48 зображено, що Wazuh може отримувати дані, що збираються і реєструються Sysmon на хостах Windows, та обробляти їх для подальшого аналізу та виявлення потенційних загроз або аномальних подій.



Time	rule.description	rule.level	rule.id
Jun 8, 2023 @ 17:24:20.015	Sysmon - Event 1: Process creation.	5	101101
Jun 8, 2023 @ 17:24:19.973	Sysmon - Event 1: Process creation.	5	101101
Jun 8, 2023 @ 17:18:43.427	Sysmon - Event 1: Process creation.	5	101101
Jun 8, 2023 @ 17:18:31.776	Sysmon - Event 1: Process creation.	5	101101
Jun 8, 2023 @ 17:18:31.760	Sysmon - Event 1: Process creation.	5	101101
Jun 8, 2023 @ 17:18:31.744	Sysmon - Event 1: Process creation.	5	101101
Jun 8, 2023 @ 17:18:31.729	Sysmon - Event 1: Process creation.	5	101101
Jun 8, 2023 @ 17:18:31.713	Sysmon - Event 1: Process creation.	5	101101
Jun 8, 2023 @ 17:18:31.711	Sysmon - Event 1: Process creation.	5	101101
Jun 8, 2023 @ 17:18:31.711	Sysmon - Event 1: Process creation.	5	101101

Рисунок 3.48 – Логи Sysmon з хоста Windows

Висновки до розділу 3

Концепція даної розгорнутої інфраструктури забезпечує автоматичне збирання індикаторів компрометації та їх виявлення на кінцевих хостах завдяки синхронізації WAZUH та MISP.

Архітектура розгорнутої інфраструктури в хмарі забезпечує відмовостійкість та постійний моніторинг кінцевих хостів, при умові наявності інтернет з'єднання. Цей підхід дозволяє покращити реакцію на потенційні загрози безпеки інформації та сприяє підвищенню рівня захисту в системі. Розгорнута інфраструктура є ефективним інструментом для моніторингу та виявлення зловмисних подій, що дозволяє забезпечити безпеку інформації в організації.

Узагальнюючи, розгорнута інфраструктура для виявлення індикаторів компрометації забезпечує високий рівень безпеки інформації в організації. Її автоматизований характер дозволяє ефективно виявляти потенційні загрози та реагувати на них, зменшуючи можливість негативних наслідків. Розгорнута інфраструктура є гнучкою, масштабованою та відмовостійкою, що робить її надійним рішенням для захисту інформації у сучасних умовах.

ВИСНОВКИ

У першій частині дипломної роботи було розглянуто поняття індикаторів компрометації, їх переваги та методи поширення. Також було розглянуто поняття розвідки загроз та важливість використання при захисті будь-якої інформаційно-комунікаційної системи. На основі цього можна зробити наступні висновки.

По-перше, визначення та аналіз індикаторів компрометації є важливою складовою процесу виявлення та реагування на потенційні загрози. Дослідження показали, що індикатори можуть бути розподілені на кілька категорій, включаючи поведінкові, технічні та контекстуальні індикатори. Це дозволяє організаціям ефективно виявляти незвичайну або підозрілу активність і приймати відповідні заходи щодо захисту своїх інформаційних ресурсів.

По друге, розвідка загроз є однією з найважливіших аспектів захисту від атак на інформаційну безпеку. Дослідження показали, що розвідка загроз включає збір інформації про цільову систему або організацію з метою виявлення слабких місць і можливостей для здійснення атак. Важливо розуміти методи, тактики та техніки, які використовуються розвідниками загроз, аби бути краще підготовленими до їх виявлення та запобігання.

Загалом, цей розділ дипломного проекту розширює знання про індикатори компрометації та розвідку загроз, допомагаючи покращити рівень безпеки інформаційних систем.

У другій частині дипломної роботи було розроблено чіткі алгоритми дій для проведення threat intelligence, що покривають кожну з тактик загальнодоступної бази знань MITRE ATT&CK Matrix. Розуміння MITRE Matrix допомагає організаціям оперативно виявляти та реагувати на будь-яку підозрілу активність.

Ці алгоритми дій є досить узагальненими і захисникам мережі потрібно враховувати особливості окремої інформаційної системи. Тим не менше, ключовими аспектами проведення розвідки загроз на кожному етапі є специфічні логи та виявлення індикаторів компрометації.

Загалом, цей розділ дипломного проекту допомагає розширити розуміння процесу розвідки загроз на основі MITRE ATT&CK і показує його важливість для забезпечення безпеки інформаційних систем. Використання цієї моделі дозволяє покращити виявлення та реагування на потенційні інциденти та підвищити рівень кібербезпеки організації.

У третій частині дипломної роботи була реалізована оптимальна інфраструктура для автоматичного збору індикаторів компрометації та логів з кінцевих хостів за допомогою Wazuh та MISP. Це надає можливість виявлення підозрілої або шкідливої поведінки в напівавтоматичному режимі.

Використання Wazuh у поєднанні з визначеними індикаторами компрометації допомагає покращити виявлення потенційних загроз та забезпечити реагування на них. Використання MISP дозволяє швидко отримувати оновлені дані про виявлені загрози та здійснювати аналіз на основі спільних інформаційних ресурсів.

Також, було розглянуто ручний пошук індикаторів компрометації шляхом відвідування відповідних веб-ресурсів. Цей метод є менш ефективним і більш часозатратним, проте також дієвим.

Загалом, використання Wazuh та MISP в поєднанні зі встановленими індикаторами компрометації є ефективним підходом до пошуку та виявлення потенційних загроз у системі. Це дозволяє організаціям оперативно реагувати на інциденти безпеки, зменшувати можливі наслідки компрометації та підвищувати рівень безпеки своїх інформаційних ресурсів.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. AlienVault OTX [Електронний ресурс] – Режим доступу до ресурсу: https://otx.alienvault.com.

2. VirusTotal [Електронний ресурс] – Режим доступу до ресурсу: https://www.virustotal.com.

3. hybrid-analysis [Електронний ресурс] – Режим доступу до ресурсу: https://www.hybrid-analysis.com.

4. Joe Sandbox [Електронний ресурс] – Режим доступу до ресурсу: https://www.joesandbox.com.

5. Cuckoo Sandbox [Електронний ресурс] – Режим доступу до ресурсу: https://cuckoosandbox.org/.

6. Any Run [Електронний ресурс] – Режим доступу до ресурсу: https://any.run/.

7. McAfee Labs [Електронний ресурс] – Режим доступу до ресурсу: https://www.mcafee.com/.

8. Cisco Talos [Електронний ресурс] – Режим доступу до ресурсу: https://www.talosintelligence.com/.

9. Trend Micro Research [Електронний ресурс] – Режим доступу до ресурсу: https://www.trendmicro.com.

10. Palo Alto Networks Unit 42 [Електронний ресурс] – Режим доступу до ресурсу: https://unit42.paloaltonetworks.com/.

11. VirusShare [Електронний ресурс] – Режим доступу до ресурсу: https://virusshare.com/.

12. MalwareBazaar [Електронний ресурс] – Режим доступу до ресурсу: https://bazaar.abuse.ch/.

13. AbuseIPDB [Електронний ресурс] – Режим доступу до ресурсу: https://www.abuseipdb.com/.

14. Reddit [Електронний ресурс] – Режим доступу до ресурсу: https://www.reddit.com/r/netsec/.

15. CERT UA [Електронний ресурс] – Режим доступу до ресурсу: https://cert.gov.ua/.

16. URLhaus [Електронний ресурс] – Режим доступу до ресурсу: https://urlhaus.abuse.ch/.

17. Threat Miner [Електронний ресурс] – Режим доступу до ресурсу: https://www.threatminer.org/.

18. APTnotes [Електронний ресурс] – Режим доступу до ресурсу: https://github.com/aptnotes/data.

19. SophosLabs IOCs [Електронний ресурс] – Режим доступу до ресурсу: https://github.com/sophoslabs/IoCs.

20. MISP ioc feeds [Електронний ресурс] – Режим доступу до ресурсу: https://github.com/MISP/MISP/blob/2.4/app/files/feed-metadata/defaults.json.

21. Decoder for Linux Sysmon (for Wazuh) [Електронний ресурс] – Режим доступу до ресурсу: https://raw.githubusercontent.com/OpenSecureCo/Demos/main/linux-sysmon.xml.

22. Rules for Linux Sysmon (For Wazuh) [Електронний ресурс] – Режим доступу до ресурсу: https://raw.githubusercontent.com/OpenSecureCo/Demos/main/sysmonforlinux-rules.xml.

23. Sysmon config for linux [Електронний ресурс] – Режим доступу до ресурсу: https://medium.com/@olafhartong/sysmon-for-linux-57de7ca48575.

24. integration block for the Wazuh and Misp synchronisation script [Електронний ресурс] – Режим доступу до ресурсу: https://opensecure.medium.com/wazuh-and-misp-integration-242dfa2f2e19.

25. Rules for Wazuh [Електронний ресурс] – Режим доступу до ресурсу: https://opensecure.medium.com/wazuh-and-misp-integration-242dfa2f2e19.

26. Wazuh rules for sysmon [Електронний ресурс] – Режим доступу до ресурсу: https://raw.githubusercontent.com/OpenSecureCo/Wazuh/main/sysmon.xml.

27. MITRE ATT&CK Matrix [Електронний ресурс] – Режим доступу до ресурсу: https://attack.mitre.org/.

ДОДАТОК А

Правило Wazuh для sysmon:

```
<!-- Log Sysmon Alerts -->
<group name="sysmon">
<rule id="101100" level="5">
<if_sid>61600</if_sid>
<field name="win.system.eventID">^22$</field>
<description>Sysmon - Event 22: DNS Query.</description>
<options>no_full_log</options>
</rule>
<rule id="101101" level="5">
<if_sid>61603</if_sid>
<options>no_full_log</options>
<description>Sysmon - Event 1: Process creation.</description>
</rule>
<rule id="101102" level="5">
<if_sid>61604</if_sid>
<options>no_full_log</options>
<description>Sysmon - Event 2: A process changed a file creationtime.</description>
</rule>
<rule id="101103" level="5">
<if_sid>61605</if_sid>
<options>no_full_log</options>
<description>Sysmon - Event 3: Network connection.</description>
</rule>
<rule id="101104" level="5">
<if_sid>61606</if_sid>
<options>no_full_log</options>
<description>Sysmon - Event 4: Sysmon service state changed.
</description>
</rule>
<rule id="101105" level="5">
<if_sid>61607</if_sid>
<options>no_full_log</options>
<description>Sysmon - Event 5: Process terminated.</description>
</rule>
```

```
<rule id="101106" level="5">
<if_sid>61608</if_sid>
<options>no_full_log</options>
<description>Sysmon - Event 6: Driver loaded.</description>
</rule>
<rule id="101107" level="5">
<if_sid>61609</if_sid>
<options>no_full_log</options>
<description>Sysmon - Event 7: Image loaded.</description>
</rule>
<rule id="101108" level="5">
<if_sid>61610</if_sid>
<options>no_full_log</options>
<description>Sysmon - Event 8: CreateRemoteThread.</description>
</rule>
<rule id="101109" level="5">
<if_sid>61611</if_sid>
<options>no_full_log</options>
<description>Sysmon - Event 9: RawAccessRead.</description>
</rule>
<rule id="101110" level="5">
<if_sid>61612</if_sid>
<options>no_full_log</options>
<description>Sysmon - Event 10: ProcessAccess.</description>
</rule>
<rule id="101111" level="5">
<if_sid>61613</if_sid>
<options>no_full_log</options>
<description>Sysmon - Event 11: FileCreate.</description>
</rule>
<rule id="101112" level="5">
<if_sid>61614</if_sid>
<options>no_full_log</options>
<description>Sysmon - Event 12: RegistryEvent (Object create and
delete).</description>
</rule>
<rule id="101113" level="5">
<if_sid>61615</if_sid>
<options>no_full_log</options>
```

```
<description>Sysmon - Event 13: RegistryEvent (Value Set).
</description>
</rule>
<rule id="101114" level="5">
<if_sid>61616</if_sid>
<options>no_full_log</options>
<description>Sysmon - Event 14: RegistryEvent (Key and ValueRename).</description>
</rule>
<rule id="101115" level="5">
<if_sid>61617</if_sid>
<options>no_full_log</options>
<description>Sysmon - Event 15: FileCreateStreamHash.
</description>
</rule>
</group>
```

ДОДАТОК Б

Скрипт для інтеграції MISP та Wazuh:

```
#!/var/ossec/framework/python/bin/python3## MISP API Integration
#
import sysimport os
from socket import socket, AF_UNIX, SOCK_DGRAMfrom datetime import date, datetime,
timedeltaimport time
import requests
from requests.exceptions import ConnectionErrorimport json
import ipaddressimport hashlib
import re
pwd = os.path.dirname(os.path.dirname(os.path.realpath(___file___)))socket_addr =
'{0}/queue/sockets/queue'.format(pwd)
def send_event(msg, agent = None):
if not agent or agent["id"] == "000":
string = '1:misp:{0}'.format(json.dumps(msg))else:
string = '1:[{0}] ({1}) {2}->misp:{3}'.format(agent["id"], agent["name"],
agent["ip"] if "ip" in agent else "any", json.dumps(msg))
sock = socket(AF_UNIX, SOCK_DGRAM)sock.connect(socket_addr)
sock.send(string.encode())
sock.close()false = False
# Read configuration parametersalert_file = open(sys.argv[1]) # Read the alert file
alert = json.loads(alert_file.read())alert_file.close()
# New Alert Output if MISP Alert or Error calling the APIalert_output = {}
# MISP Server Base URL
misp_base_url = "https://**your misp instance**/attributes/restSearch/" # MISP
Server API AUTH KEY
misp_api_auth_key = "*Your API Key"# API - HTTP Headers
misp_apicall_headers = {"Content-Type":"application/json", "Authorization":f"
{misp_api_auth_key}", "Accept":"application/json"}
## Extract Sysmon for Windows/Sysmon for Linux and Sysmon Event IDevent_source =
alert["rule"]["groups"][0]
event_type = alert["rule"]["groups"][2]
## Regex Pattern used based on SHA256 lenght (64 characters)regex_file_hash =
re.compile('\w{64}')
if event_source == 'windows':
```

```
      if event_type == 'sysmon_event1':try:
      wazuh_event_param          =          regex_file_hash.search(alert["data"]["win"]
["eventdata"]["hashes"]).group(0)
      except IndexError:sys.exit()
      elif   event_type   ==   'sysmon_event3'   and   alert["data"]["win"]["eventdata"]
["destinationIsIpv6"] == 'false':
      try:
      dst_ip        =        alert["data"]["win"]["eventdata"]["destinationIp"]       if
ipaddress.ip_address(dst_ip).is_global:
      wazuh_event_param = dst_ipelse:
      sys.exit()except IndexError:
      sys.exit()
      elif    event_type    ==    'sysmon_event3'    and    alert_output["data"]["win"]
["eventdata"]["destinationIsIpv6"] == 'true':
      sys.exit()
      elif event_type == 'sysmon_event6':try:
      wazuh_event_param          =          regex_file_hash.search(alert["data"]["win"]
["eventdata"]["hashes"]).group(0)
      except IndexError:sys.exit()
      elif event_type == 'sysmon_event7':try:
      wazuh_event_param          =          regex_file_hash.search(alert["data"]["win"]
["eventdata"]["hashes"]).group(0)
      except IndexError:sys.exit()
      elif event_type == 'sysmon_event_15':try:
      wazuh_event_param          =          regex_file_hash.search(alert["data"]["win"]
["eventdata"]["hashes"]).group(0)
      except IndexError:sys.exit()
      elif event_type == 'sysmon_event_22':try:
      wazuh_event_param    =    alert["data"]["win"]["eventdata"]["queryName"]   except
IndexError:
      sys.exit()
      elif event_type == 'sysmon_event_23':try:
      wazuh_event_param          =          regex_file_hash.search(alert["data"]["win"]
["eventdata"]["hashes"]).group(0)
      except IndexError:sys.exit()
      elif event_type == 'sysmon_event_24':try:
      wazuh_event_param          =          regex_file_hash.search(alert["data"]["win"]
["eventdata"]["hashes"]).group(0)
      except IndexError:sys.exit()
```

```
      elif event_type == 'sysmon_event_25':try:
      wazuh_event_param          =          regex_file_hash.search(alert["data"]["win"]
["eventdata"]["hashes"]).group(0)
      except IndexError:sys.exit()
      else:
      sys.exit()
      misp_search_value = "value:"f"{wazuh_event_param}"
      misp_search_url = ''.join([misp_base_url, misp_search_value])try:
      misp_api_response   =   requests.get(misp_search_url, headers=misp_apicall_headers,
verify=False)
      except ConnectionError:
      alert_output["misp"] = {}
      alert_output["integration"] = "misp"
      alert_output["misp"]["error"]    =    'Connection    Error    to    MISP    API'
send_event(alert_output, alert["agent"])
      else:
      misp_api_response = misp_api_response.json()# Check if response includes Attributes
(IoCs)
      if (misp_api_response["response"]["Attribute"]):# Generate Alert Output from MISP
Responsealert_output["misp"] = {}
      alert_output["misp"]["source"] = {}
      alert_output["misp"]["event_id"]          =          misp_api_response["response"]
["Attribute"][0]["event_id"]
      alert_output["misp"]["category"]          =          misp_api_response["response"]
["Attribute"][0]["category"]
      alert_output["misp"]["value"]           =           misp_api_response["response"]
["Attribute"][0]["value"]
      alert_output["misp"]["type"]           =           misp_api_response["response"]
["Attribute"][0]["type"]
      alert_output["misp"]["source"]["description"] = alert["rule"]["description"]
      send_event(alert_output, alert["agent"])elif event_source == 'linux':
      if     event_type     ==     'sysmon_event3'     and     alert["data"]["eventdata"]
["destinationIsIpv6"] == 'false':
      try:
      dst_ip          =          alert["data"]["eventdata"]["DestinationIp"]         if
ipaddress.ip_address(dst_ip).is_global:
      wazuh_event_param = dst_ip
      misp_search_value = "value:"f"{wazuh_event_param}"
      misp_search_url = ''.join([misp_base_url, misp_search_value])try:
```

```
      misp_api_response   =   requests.get(misp_search_url, headers=misp_apicall_headers,
verify=False)
      except ConnectionError:
      alert_output["misp"] = {}
      alert_output["integration"] = "misp"
      alert_output["misp"]["error"] = 'Connection Error to MISP
      send_event(alert_output, alert["agent"])else:
      misp_api_response = misp_api_response.json()# Check if response includes Attributes
(IoCs)
      if (misp_api_response["response"]["Attribute"]):# Generate Alert Output from MISP
Response
      alert_output["misp"] = {}
      alert_output["misp"]["event_id"]
=misp_api_response["response"]["Attribute"][0]["event_id"]
      alert_output["misp"]["category"]                                                =
misp_api_response["response"]["Attribute"][0]["category"]
      alert_output["misp"]["value"]                                                   =
misp_api_response["response"]["Attribute"][0]["value"]
      alert_output["misp"]["type"]                                                    =
misp_api_response["response"]["Attribute"][0]["type"]
      send_event(alert_output, alert["agent"])
      else:
      sys.exit()
      except IndexError:sys.exit()
      else:
      sys.exit()
      elif event_source == 'ossec' and event_type == "syscheck_entry_added":try:
      wazuh_event_param = alert["syscheck"]["sha256_after"]except IndexError:
      sys.exit()
      misp_search_value = "value:"f"{wazuh_event_param}"
      misp_search_url = ''.join([misp_base_url, misp_search_value])try:
      misp_api_response   =   requests.get(misp_search_url, headers=misp_apicall_headers,
verify=false)
      except ConnectionError:
      alert_output["misp"] = {}
      alert_output["integration"] = "misp"
      alert_output["misp"]["error"]    =    'Connection    Error    to    MISP    API'
send_event(alert_output, alert["agent"])
      else:
```

```
      misp_api_response = misp_api_response.json()# Check if response includes Attributes
(IoCs)
      if (misp_api_response["response"]["Attribute"]):# Generate Alert Output from MISP
Response

      alert_output["misp"] = {}
      alert_output["misp"]["event_id"]          =          misp_api_response["response"]
["Attribute"][0]["event_id"]
      alert_output["misp"]["category"]          =          misp_api_response["response"]
["Attribute"][0]["category"]
      alert_output["misp"]["value"]            =           misp_api_response["response"]
["Attribute"][0]["value"]
      alert_output["misp"]["type"]             =           misp_api_response["response"]
["Attribute"][0]["type"]
      send_event(alert_output, alert["agent"])
      else:
      sys.exit()
```

ДОДАТОК В

Конфігураційний файл джерел для MISP:

```
[
 {
  "Feed": {
   "name": "CIRCL OSINT Feed",
   "provider": "CIRCL",
   "url": "https://www.circl.lu/doc/misp/feed-osint",
   "rules": "{\"tags\":{\"OR\":[],\"NOT\":[]},\"orgs\":{\"OR\":[],\"NOT\":[]}}",
   "enabled": true,
   "distribution": "3",
   "default": true,
   "source_format": "misp",
   "fixed_event": false,
   "delta_merge": false,
   "publish": false,
   "override_ids": false,
   "settings": "{\"csv\":{\"value\":\"\",\"delimiter\":\"\"},\"common\":{\"excluderegex\":\"\"}}",
   "input_source": "network",
   "delete_local_file": false,
   "lookup_visible": false
  }
 },
 {
  "Feed": {
   "name": "The Botvrij.eu Data",
   "provider": "Botvrij.eu",
   "url": "https://www.botvrij.eu/data/feed-osint",
   "rules": "{\"tags\":{\"OR\":[],\"NOT\":[]},\"orgs\":{\"OR\":[],\"NOT\":[]}}",
   "enabled": true,
   "distribution": "3",
   "default": true,
   "source_format": "misp",
   "fixed_event": false,
```

```
    "delta_merge": false,
    "publish": false,
    "override_ids": false,
    "settings": "{\"csv\":{\"value\":\"\",\"delimiter\":\"\"},\"common\":{\"excluderegex\":\"\"}}",
    "input_source": "network",
    "delete_local_file": false,
    "lookup_visible": false
   }
  },
  {
   "Feed": {
    "name": "blockrules of rules.emergingthreats.net",
    "provider": "rules.emergingthreats.net",
    "url": "https://rules.emergingthreats.net/blockrules/compromised-ips.txt",
    "rules": "{\"tags\":{\"OR\":[],\"NOT\":[]},\"orgs\":{\"OR\":[],\"NOT\":[]}}",
    "enabled": true,
    "distribution": "0",
    "default": false,
    "source_format": "csv",
    "fixed_event": true,
    "delta_merge": true,
    "publish": false,
    "override_ids": true,
    "settings": "{\"csv\":{\"value\":\"1\"}}",
    "input_source": "network",
    "delete_local_file": false,
    "lookup_visible": false
   },
   "Tag": {
    "name": "osint:source-type=\"block-or-filter-list\"",
    "colour": "#004f89",
    "exportable": true,
    "hide_tag": false
   }
  },
```

```
        {
          "Feed": {
            "name": "Tor exit nodes",
             "provider": "TOR Node List from dan.me.uk - careful, this feed applies a lock-out after each pull. This is shared with the \"Tor ALL nodes\" feed.",
            "url": "https://www.dan.me.uk/torlist/?exit",
            "rules": "{\"tags\":{\"OR\":[],\"NOT\":[]},\"orgs\":{\"OR\":[],\"NOT\":[]}}",
            "enabled": true,
            "distribution": "0",
            "default": false,
            "source_format": "csv",
            "fixed_event": true,
            "delta_merge": true,
            "publish": false,
            "override_ids": false,
            "settings": "{\"csv\":{\"value\":\"\",\"delimiter\":\"\"},\"common\":{\"excluderegex\":\"\"}}",
            "input_source": "network",
            "delete_local_file": false,
            "lookup_visible": true
          },
          "Tag": {
            "name": "osint:source-type=\"block-or-filter-list\"",
            "colour": "#004f89",
            "exportable": true,
            "hide_tag": false
          }
        },
        {
          "Feed": {
            "name": "Tor ALL nodes",
             "provider": "TOR Node List from dan.me.uk - careful, this feed applies a lock-out after each pull. This is shared with the \"Tor exit nodes\" feed.",
            "url": "https://www.dan.me.uk/torlist/",
            "rules": "{\"tags\":{\"OR\":[],\"NOT\":[]},\"orgs\":{\"OR\":[],\"NOT\":[]}}",
            "enabled": true,
```

```
    "distribution": "0",
    "default": false,
    "source_format": "csv",
    "fixed_event": true,
    "delta_merge": true,
    "publish": false,
    "override_ids": false,
    "settings": "{\"csv\":{\"value\":\"\",\"delimiter\":\"\"},\"common\":{\"excluderegex\":\"\"}}",
    "input_source": "network",
    "delete_local_file": false,
    "lookup_visible": true
   },
   "Tag": {
    "name": "osint:source-type=\"block-or-filter-list\"",
    "colour": "#004f89",
    "exportable": true,
    "hide_tag": false
   }
  },
  {
   "Feed": {
    "name": "cybercrime-tracker.net - all",
    "provider": "cybercrime-tracker.net",
    "url": "https://cybercrime-tracker.net/all.php",
    "rules": "",
    "enabled": true,
    "distribution": "0",
    "default": false,
    "source_format": "freetext",
    "fixed_event": true,
    "delta_merge": true,
    "publish": false,
    "override_ids": false,
    "settings": "\"{\\\"csv\\\":{\\\"value\\\":\\\"\\\"}}\"",
    "input_source": "network",
```

```
            "delete_local_file": false,
            "lookup_visible": false
           },
           "Tag": {
            "name": "osint:source-type=\"block-or-filter-list\"",
            "colour": "#004f89",
            "exportable": true,
            "hide_tag": false
           }
          },
          {
           "Feed": {
            "name": "Phishtank online valid phishing",
            "provider": "Phishtank",
            "url": "https://data.phishtank.com/data/online-valid.csv",
            "rules": "",
            "enabled": true,
            "distribution": "0",
            "default": false,
            "source_format": "csv",
            "fixed_event": true,
            "delta_merge": true,
            "publish": false,
            "override_ids": false,
                                                                                "settings":
"{\"csv\":{\"value\":\"2\",\"delimiter\":\",\"},\"common\":{\"excluderegex\":\"\\/^http:\\\\\\/\\\\\\/www.phishtank.com\\/i\"}}",
            "input_source": "network",
            "delete_local_file": false,
            "lookup_visible": false
           },
           "Tag": {
            "name": "osint:source-type=\"block-or-filter-list\"",
            "colour": "#004f89",
            "exportable": true,
```

```
            "hide_tag": false
          }
        },
        {
          "Feed": {
            "name": "ip-block-list - snort.org",
            "provider": "https://snort.org",
            "url": "https://snort.org/downloads/ip-block-list",
            "rules": "{\"tags\":{\"OR\":[],\"NOT\":[]},\"orgs\":{\"OR\":[],\"NOT\":[]}}",
            "enabled": true,
            "distribution": "0",
            "default": false,
            "source_format": "freetext",
            "fixed_event": true,
            "delta_merge": true,
            "publish": true,
            "override_ids": false,
            "settings": "{\"csv\":{\"value\":\"\",\"delimiter\":\",\"},\"common\":{\"excluderegex\":\"\"}}",
            "input_source": "network",
            "delete_local_file": false,
            "lookup_visible": false
          },
          "Tag": {
            "name": "osint:source-type=\"block-or-filter-list\"",
            "colour": "#004f89",
            "exportable": true,
            "hide_tag": false
          }
        },
        {
          "Feed": {
            "name": "diamondfox_panels",
            "provider": "pan-unit42",
                                                    "url":          "https://raw.githubusercontent.com/pan-
unit42/iocs/master/diamondfox/diamondfox_panels.txt",
```

```
    "rules": "",
    "enabled": true,
    "distribution": "0",
    "default": false,
    "source_format": "freetext",
    "fixed_event": true,
    "delta_merge": true,
    "publish": true,
    "override_ids": false,
    "settings": "{\"csv\":{\"value\":\"\",\"delimiter\":\",\"},\"common\":{\"excluderegex\":\"\"}}",
    "input_source": "network",
    "delete_local_file": false,
    "lookup_visible": false
   },
   "Tag": {
    "name": "osint:source-type=\"block-or-filter-list\"",
    "colour": "#004f89",
    "exportable": true,
    "hide_tag": false
   }
  },
  {
   "Feed": {
    "name": "pop3gropers",
    "provider": "home.nuug.no",
    "url": "https://home.nuug.no/~peter/pop3gropers.txt",
    "rules": "",
    "enabled": true,
    "distribution": "0",
    "default": false,
    "source_format": "csv",
    "fixed_event": true,
    "delta_merge": true,
    "publish": true,
    "override_ids": false,
```

```
    "settings": "{\"csv\":{\"value\":\"\",\"delimiter\":\",\"},\"common\":{\"excluderegex\":\"\"}}",
    "input_source": "network",
    "delete_local_file": false,
    "lookup_visible": false
   },
   "Tag": {
    "name": "osint:source-type=\"block-or-filter-list\"",
    "colour": "#004f89",
    "exportable": true,
    "hide_tag": false
   }
  },
  {
   "Feed": {
    "name": "Feodo IP Blocklist",
    "provider": "abuse.ch",
    "url": "https://feodotracker.abuse.ch/downloads/ipblocklist.csv",
    "rules": "{\"tags\":{\"OR\":[],\"NOT\":[]},\"orgs\":{\"OR\":[],\"NOT\":[]}}",
    "enabled": true,
    "distribution": "3",
    "default": false,
    "source_format": "csv",
    "fixed_event": true,
    "delta_merge": false,
    "publish": false,
    "override_ids": false,
    "settings": "{\"csv\":{\"value\":\"2\",\"delimiter\":\",\"},\"common\":{\"excluderegex\":\"\"}}",
    "input_source": "network",
    "delete_local_file": false,
    "lookup_visible": true
   }
  },
  {
   "Feed": {
    "name": "OpenPhish url list",
```

```
    "provider": "openphish.com",
    "url": "https://openphish.com/feed.txt",
    "rules": "",
    "enabled": false,
    "distribution": "3",
    "default": false,
    "source_format": "freetext",
    "fixed_event": true,
    "delta_merge": true,
    "publish": false,
    "override_ids": false,
    "settings": "{\"csv\":{\"value\":\"\",\"delimiter\":\",\"},\"common\":{\"excluderegex\":\"\"}}",
    "input_source": "network",
    "delete_local_file": false,
    "lookup_visible": true
   },
   "Tag": {
    "name": "osint:source-type=\"block-or-filter-list\"",
    "colour": "#004f89",
    "exportable": true,
    "hide_tag": false
   }
  },
  {
   "Feed": {
    "name": "firehol_level1",
    "provider": "iplists.firehol.org",
    "url": "https://raw.githubusercontent.com/ktsaou/blocklist-ipsets/master/firehol_level1.netset",
    "rules": "{\"tags\":{\"OR\":[],\"NOT\":[]},\"orgs\":{\"OR\":[],\"NOT\":[]}}",
    "enabled": true,
    "distribution": "3",
    "default": false,
    "source_format": "freetext",
    "fixed_event": true,
    "delta_merge": true,
```

```
            "publish": false,
            "override_ids": false,
            "settings": "{\"csv\":{\"value\":\"\",\"delimiter\":\",\"},\"common\":{\"excluderegex\":\"\"}}",
            "input_source": "network",
            "delete_local_file": false,
            "lookup_visible": true
          },
          "Tag": {
            "name": "osint:source-type=\"block-or-filter-list\"",
            "colour": "#004f89",
            "exportable": true,
            "hide_tag": false
          }
        },
        {
          "Feed": {
            "name": "IPs from High-Confidence DGA-Based C&Cs Actively Resolving - requires a valid
license",
            "provider": "osint.bambenekconsulting.com",
            "url": "https://osint.bambenekconsulting.com/feeds/c2-ipmasterlist-high.txt",
            "rules": "{\"tags\":{\"OR\":[],\"NOT\":[]},\"orgs\":{\"OR\":[],\"NOT\":[]}}",
            "enabled": true,
            "distribution": "3",
            "default": false,
            "source_format": "csv",
            "fixed_event": true,
            "delta_merge": true,
            "publish": false,
            "override_ids": false,
            "settings": "{\"csv\":{\"value\":\"1\",\"delimiter\":\",\"},\"common\":{\"excluderegex\":\"\"}}",
            "input_source": "network",
            "delete_local_file": false,
            "lookup_visible": true
          },
          "Tag": {
```

```
    "name": "osint:source-type=\"block-or-filter-list\"",
    "colour": "#004f89",
    "exportable": true,
    "hide_tag": false
   }
  },
  {
   "Feed": {
    "name": "Domains from High-Confidence DGA-based C&C Domains Actively Resolving",
    "provider": "osint.bambenekconsulting.com",
    "url": "https://osint.bambenekconsulting.com/feeds/c2-dommasterlist-high.txt",
    "rules": "",
    "enabled": true,
    "distribution": "3",
    "default": false,
    "source_format": "csv",
    "fixed_event": true,
    "delta_merge": true,
    "publish": false,
    "override_ids": false,
    "settings":
```

ДОДАТОК Г
СПИСОК ОПУБЛІКОВАНИХ ПРАЦЬ ЗА ТЕМОЮ ДИСЕРТАЦІЇ

Тези наукових конференцій:

1. Власюк Ю. Використання інструментів OSINT для виявлення технічних характеристик та конфігурації інфраструктури цільової організації / Іван Пархоменко, Лариса Мирутенко, Юлія Власюк, Іван Нечипоренко / VI Міжнародна науково-практична конференція «Проблеми кібербезпеки інформаційно-телекомунікаційних систем» (PCSITS) 27 квітня 2023, Київ, Україна, стр. 80-82.