МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ АВІАЦІЙНИЙ УНІВЕРСИТЕТ
КАФЕДРА КОМП'ЮТЕРИЗОВАНИХ СИСТЕМ ЗАХИСТУ ІНФОРМАЦІЇ

ДОПУСТИТИ ДО ЗАХИСТУ
Завідувач кафедри
_____ С.В. Казмірчук
«_____»_____ 2020 р.

На правах рукопису
УДК 004.056.5:510.22(043.3)

КВАЛІФІКАЦІЙНА РОБОТА

ЗДОБУВАЧА ВИЩОЇ ОСВІТИ
ОСВІТНЬОГО СТУПЕНЯ «МАГІСТР»

Тема: **Програмний модуль аналізу подій для ELK стеку**

Виконавець: О.В. Мар'янська

Науковий керівник: к.т.н. О.О. Висоцька

Нормоконтролер: к.т.н. О.О. Висоцька

Київ 2020

ВСТУП

Актуальність. ІТ-інфраструктура сучасних компаній часом досить різноманітна. При цьому, з розвитком технологій, головною проблемою побудови захисту ІКС стало не відсутність інформації, а її обробка. Число джерел, що забезпечують надходження актуальної інформації щодо поточного стану захищеності ІКС, безперервно зростає. Дійсно, сьогодні навряд чи можна знайти корпоративне програмне забезпечення, яке б не вело запис в журнал подій ІБ і т.п. Але, разом зі збільшенням обсягу інформації, адміністраторам ІБ все складніше відстежувати «загальну картину».

Одними з найважливіших характеристик інформаційних систем є їх безперебійне функціонування і захищеність даних користувачів. У зв'язку з цим зростає необхідність в розвитку систем їх захисту від різних загроз, включаючи складні цільові атаки, що виконуються в кілька етапів, часто рознесені в часі. Виявлення таких атак вимагає ретельного аналізу подій ІБ, одержуваних від різних датчиків безпеки і об'єктів хмарної інфраструктури за тривалий період часу. Багато сучасних систем виявлення вторгнень не здатні встановити взаємозв'язки між подіями ІБ у вигляді послідовності етапів виконання атаки, так як не мають інструментів аналізу поточних загроз в часовому контексті. У більшості випадків оцінка достовірності згенерованих подій ІБ не здійснюється, а критичність події ІБ часто не залежить від рівня критичності контрольованих ресурсів. Кореляція цих двох показників дозволила б адміністратору безпеки більш точно розставляти пріоритети подій ІБ для своєчасного реагування на них. Для усунення цих недоліків в системах управління інформаційною безпекою пропонується використовувати в якості складового компонента модуль кореляції подій безпеки.

Але ж якщо своєчасно не аналізувати виникаючі загрози і не намагатися запобігти їм, будь-яка система захисту виявиться марною. У цих умовах саме час замислитися про системи управління подіями і інцидентами інформаційної

безпеки. Крім того, все частіше зловмисники переходять від атак «в лоб» до більш складних і розподілених сценаріїв[2](APT). Термін APT був введений військово-повітряними силами США в 2006 році для опису нового виду атак. Тоді вперше була зроблена спроба проаналізувати проведену атаку, зробити висновки і спробувати протистояти новій загрозі. APT - це не якийсь наворочений експлойт і не новомодний троян. - це парадигма атаки.

Зазвичай атаки типу розвинених сталих загроз відбуваються за декілька етапів:

- Розвідка (англ. data gathering, reconnaissance) — виявлення слабких місць у захисті організації (запити до домену організації, сканування портів та вразливостей сервісів);
- Початкове втручання (англ. initial entry) — виявленні вразливості експлуатуються для закріплення в мережі, для чого використовуються складні техніки (spear-fishing, соціальна інженерія);
- Розширення повноважень (англ. escalation of privileges) — подальша експлуатація, хакери працюють над отриманням якомога більшого контролю над системами та над додатковими системами, встановлюють «бекдори» (англ. backdoor), які спрощують повторний доступ в систему
- Подальша експлуатація (англ. continuous exploitation) — нападники отримують можливість постійного та безперешкодної ідентифікації, компрометації та використання конфіденційних даних.

Проблема ж у тому, що в разі подібної атаки всі засоби захисту можуть «мовчати», так як «вирвані з контексту», інциденти не будуть сприйматися як серйозна загроза. Але в той же самий час, аналіз сукупності інцидентів може явно вказати на атаку. Саме ці магічні властивості приписують сучасним системам управління подіями і інцидентами інформаційної безпеки - здатність виявляти атаки по «крупицях», аномаліям, пост-аналізу подій і т.д.

Відомі підходи до вирішення поставленої задачі. На сьогоднішній день існує безліч інструментальних засобів і методик управління подіями ІБ в ІКС, які можна розділити по функціональності на наступні основні групи[3]:

- збір і агрегація;
- аналіз і кореляція;
- оповіщення;
- візуалізація;
- зберігання;
- експертний аналіз і пошук.

Часто перед фахівцями компанії для підвищення ефективності вирішення завдань захисту інформації виникає питання про вибір відповідного засобу, що задовольняє поточним вимогам інформаційної безпеки. При виборі засобів SIEM, експерт стикається з безліччю питань таких як, наприклад, «Які використовувати параметри?», «Який метод кореляційного аналізу використовується?», «Який набір вбудованих правил кореляції?», «Як розраховується вартість ліцензії?» і т. д. Ці та інші фактори створюють ряд труднощів при виборі відповідних продуктів управління подіями і інцидентами інформаційної безпеки.

Метою даної роботи є розробка програмного модулю аналізу подій з інтеграцією в ELK стек, яку можна використовувати як повноцінну систему управління подіями і інцидентами інформаційної безпеки для організацій.

Досягнення поставленої мети дипломної роботи передбачає завдання:

- дослідити існуючі рішення систем управління подіями і інцидентами інформаційної безпеки;
- дослідити та проаналізувати відомі підходи кореляції подій ІБ, архітектури модулю аналізу подій;
- розробити та протестувати програмний модуль аналізу подій для ELK Stack;
- інтегрувати розроблений модуль в ELK Stack .

Галузь застосування. Розроблений програмний модуль аналізу подій відноситься до галузі інформаційної безпеки і може бути використаним для підвищення рівня захищеності ІКС за рахунок використання ELK стеку з додаванням програмного модулю аналізу подій як SIEM системи.

Об’єктом дослідження є процес аналізу подій ІБ та виявлення загроз в ІКС."

Предметом дослідження є алгоритми та засоби аналізу подій ІБ на виявлення загроз ІБ.

Методи дослідження базуються на основі аналіза логіки кореляції подій ІБ(для розробки архітектури системи з розробленим модулем та директив кореляції), та об’єктно-орієнтованого програмування (для програмної реалізації розробленого методу).

Науковою новизною здобутих результатів полягає в наступному: вперше розроблено програмний модуль аналізу подій, інтеграція якого до ELK Stack, за рахунок збільшення функціональних можливостей ELK Stack, робить можливим використання ELK Stack, в якості повноцінної системи управління подіями і інцидентами інформаційної безпеки.

Практична цінність полягає у наступному:

- запропонований програмний модуль може використовуватися в реальних практичних цілях для виявлення загроз ІБ в реальному часі та аналізу спрямованих атак на мережу організації;
- сформульовані практичні рекомендації є корисними експертам з інформаційної безпеки при прийнятті рішень щодо доцільності використання готового продукту управління подіями і інцидентами інформаційної безпеки або ж доопрацювання OpenSource проекту ELK Stack.

РОЗДІЛ 1. СУЧАСНІ СИСТЕМИ УПРАВЛІННЯ ПОДІЯМИ І ІНЦИДЕНТАМИ ІБ

1.1. Система управління подіями і інцидентами інформаційної безпеки

На сьогоднішній день все більше компаній стикається з необхідністю обробки журналів подій, які реєструються в інформаційних системах, з метою виявлення можливих атак. При цьому навіть в невеликій компанії в журналах аудиту може реєструватися до декількох десятків подій в секунду, що робить їх аналіз в ручному режимі тривалим і вкрай неефективним. Для того щоб автоматизувати процес збору та аналізу інформації про події ІБ, можуть використовуватися спеціалізовані системи моніторингу.

Коли компанія починає всерйоз займатися інформаційною безпекою, їй доводиться впроваджувати величезну кількість різнорідних систем. Оскільки безпека - це не тільки стан системи (в класичному визначенні), але і процеси, невід'ємною частиною яких є моніторинг подій ІБ, рано чи пізно з'явиться необхідність централізованого спостереження і аналізу логів, які у величезній кількості можуть генеруватися перерахованими системами.

З завданням проведення аналізу наданих логів журналів подій одночасно з всіх систем ІКС, виникає проблема, оскільки практично у всіх джерел логів свій власний формат. При цьому деякі системи можуть писати логи в декількох різних форматах, що відрізняються рівнем інформативності.

При такому розмаїтті форматів і способів зберігання аналіз логів всіх цих систем може стати досить трудомістким процесом. Звичайно ж, для спрощення подібних завдань вже існує спеціальний клас ПО - система управління подіями і інцидентами інформаційної безпеки(SIEM)[1]. І якщо хороших систем виявлення вторгнень з відкритим вихідним кодом достатньо, то ось SIEM з відкритим вихідним кодом практично немає. Існує досить багато рішень для конкретної системи виявлення атак(IDS), що являє собою програмний або апаратний засіб, призначений для виявлення фактів несанкціонованого доступу в комп'ютерну систему або мережу, або несанкціонованого управління ними в основному через Інтернет. Наприклад, Snorby для Snort або Analogi для OSSEC.

Але можливості додати в ці системи якісь додаткові джерела подій не передбачено. Є системи, які володіють і більш широкими можливостями, але, скоріше, це безкоштовні версії комерційних рішень з цілим набором штучних обмежень.

SIEM не здатна самостійно запобігати інцидентам, як і не має вбудованих захисних функцій. Призначення даної системи полягає в аналізі даних, що надходять від різних інших систем, таких як програмно-апаратних засобів запобігання витокам даних(DLP), систем виявлення атак(IDS), міжмережевих екранів, антивірусів, активного мережевого обладнання, системи контролю доступу і аутентифікації, сканерів вразливостей, і т.д., а також реєстрації та повідомлення про інцидент при виявленні відхилення від норм за заздалегідь заданими критеріями. Метою подібних систем є визначення першопричини того чи іншого інциденту за деякими ознаками, подібно діагностиці хвороби по її симптомах. Так, наприклад, при зборі даних про події з точки зору системи, яка зареєструвала події, їх пріоритет може бути незначним. Однак сукупність цих подій може бути ознакою серйозного інциденту, що загрожує не менше серйозними наслідками.

Для досягнення даної мети перед системами класу SIEM ставляться наступні завдання:

- Консолідація та зберігання журналів подій від різних джерел для подальшого звернення до них при розслідуванні інциденту навіть після видалення або втрати доступу до даних першоджерела;
- Надання інструментів для аналізу подій і розслідування інцидентів з можливістю фільтрації некритичних подій, їх уніфікації та поданні інформації про події в більш читабельній і наочній формі у вигляді звітів, що гнучко налаштовуються:
- Кореляція і обробка подій ІБ за правилами;
- Автоматичне сповіщення і управління інцидентами.

Універсальність системи SIEM обумовлюється гнучкістю її логіки. Однак для її ефективного функціонування необхідні корисні джерела і ретельно написані правила кореляції.

Саме вони, в сукупності з розміром накопиченої статистики в базі, в подальшому визначать кількість хибно-позитивних спрацювань системи, які, на жаль, неминучі на момент початку її експлуатації. Як джерело вхідної інформації для SIEM може бути використаний практично будь-яка подія, наприклад, відкриття дверей конкретної кімнати. Критеріями відбору таких джерел є такі фактори:

- критичність системи (цінність, ризики) та інформації (оброблюваної і яка зберігається);
- достовірність та інформативність джерела подій; покриття каналів передачі інформації (необхідно враховувати не тільки зовнішній, а й внутрішній периметр мережі);
- рішення спектра завдань ІТ та ІБ (забезпечення безперервності, розслідування інцидентів, дотримання політик, запобігання витокам інформації і т.п.).

Так основними джерелами інформації для сучасних систем SIEM є:

- дані контролю доступу та аутентифікації для моніторингу контролю доступу до інформаційних систем і використання привілеїв;
- журнали подій серверів і робочих станцій - для контролю доступу, забезпечення безперервності, дотримання політик інформаційної безпеки;
- мережеве активне обладнання (контроль змін і доступ, лічильники мережевого трафіку);
- засоби виявлення і запобігання вторгнень(IDS / IPS) - події про мережевих атаках, зміна конфігурацій і доступ до пристроїв;
- антивірусний захист - події про працездатність програмного забезпечення (ПО), про бази даних, про зміну конфігурацій і політик, про шкідливі програми;

- сканери вразливостей - інвентаризація активів, сервісів, програмного забезпечення, вразливостей, поставка інвентаризаційних даних і топологічної структури;
- системи для врахування ризиків, критичності загрози, приоритизации інциденту;
- інші системи захисту і контролю політик ІБ - DLP, контролю пристроїв; системи інвентаризації та управління активами - для виявлення нових пристроїв і програмного забезпечення, в тому числі встановлення несанкціонованого; системи обліку трафіку.

Як було сказано раніше, збір даних від джерел здійснюється встановленими на них агентами. У разі відсутності колектора, відповідного джерела, події можуть бути відправлені в форматі стандарту Syslog. Крім того, є можливість віддаленого збору даних (за допомогою з'єднання за протоколами NetBIOS, RPC, TFTP, FTP). Однак при цьому способі виникає проблеми навантаження на мережу, так як передача журналів здійснюється цілком, а не окремими новими записами. Крім цього, в сучасних системах класу SIEM присутня можливість запису всіх мережевих даних для подальшого кореляційного аналізу. Зростающий обсяг оброблюваних даних змусив розробників відмовитися від реляційних СУБД (систем управління базами даних) *на користь не реляційних рішень з метою підвищення швидкості обробки і скорочення обсягів необхідного для зберігання дискового простору*. В ряді випадків та ж тенденція зумовила перехід деяких розробників до технології Великих Даних (BigData), що істотно підвищує вартість кінцевого продукту.

Для виправдання витрат на систему збору та кореляції подій, необхідно, щоб дані не тільки вносилися в консолідоване сховище для їх подальшого розбору за фактом інциденту, а й оброблялися. Очевидно, що інструментарій даної системи дозволить істотно прискорити процес розбору інциденту, проте основним завданням SIEM є своєчасне виявлення, оперативне реагування та запобігання загрозам. Для цього необхідно складання правил кореляції з урахуванням актуальних для компанії ризиків, а також постійна актуалізація

самих правил фахівцями. Як і в випадку IDS, типова загроза буде реалізована, якщо не поставити правило, що дозволяє цю загрозу виявити. Однак, SIEM має перевагу перед IDS, яка полягає в можливості загального опису симптомів і використання накопиченої статистики для відстеження відхилення інформаційних систем і трафіку від нормальної поведінки. У найпростішому випадку в SIEM-системах правила представлені в форматі RBR (RuleBasedReasoning) і містять набір умов, тригери, лічильники, сценарій дій. Наприклад, система здатна враховувати параметри віддаленості двох останніх використанню банківської карти за невеликий інтервал часу: якщо клієнт використовував карту для оплати покупки в Казані, а через 15 хвилин з тієї ж карти намагаються зняти денний ліміт клієнта десь в Китаї, то очевидна спроба шахрайства. В цілому система класу SIEM здатна виявляти факти мережевих атак у внутрішньому і зовнішньому периметрах, вірусних епідемії або окремих заражень шкідливим ПЗ, спроби несанкціонованого доступу до конфіденційної інформації та шахрайства, а також визначати помилки і збої в роботі інформаційних систем, уразливості, помилки конфігурацій в засобах захисту та інформаційних системах.

1.2. Існуючі рішення SIEM

1.2.1. IBM QRadar

Компанія IBM включила в платформу QRadar[4] кілька модулів: QRadar SIEM, Log Manager (управління журналами), Risk Manager (управління ризиками), Vulnerability Manager (управління вразливостями), колектори QFlow і VFlow і Incident Forensics.

Основний модуль IBM QRadar - QRadar SIEM. Даний модуль представляє саму систему, яка збирає, аналізує та управляє подіями та потоками мереж із пристроїв, кінцевих токів, серверів, антивірусів, брандмауерів та різних системних запобігань проникненню.

QRadar SIEM централізовано аналізує та консолідує власні дані за допомогою технології Sense Analytics для виявлення вразливостей та складних загроз безпеки. Цей модуль збирає всі пов'язані між собою події в одному інциденті для того, щоб надати IT-спеціалістам розроблені дані про атаки, а саме час, мету, уявність систем, дані користувачів, відомості про попередні гроші та вторгнення

QRadar SIEM може автоматично виявляти джерела даних, а також має вбудовані шаблони. Це функціональність дає можливість ввести систему в найкоротші терміни. Консоль управління простою у використанні та дозволяє у консолідованому вигляді отримувати важливу інформацію за угрозами та уявленнями. Перевагою є те, що панель управління пропонується як функціональна, тому користувачі можуть самі створювати та налаштовувати свій робочий простір. Деталізація функціональних можливостей дозволяє розширити можливості виявлення, вибору та аналізу подій та мережевих потоків, які безпосередньо пов'язані з порушеннями.

Панель управління IBM QRadar SIEM

Основний елемент IBM QRadar SIEM - база даних, яка зберігає інформацію про події і потоках в мережі. Система оснащена технологією DPI (deep packet

inspection), яка збирає події з брандмауерів і інших мережевих ресурсів і виконує глибокий аналіз мережевих пакетів.

Функціонал QRadar SIEM також забезпечує збір інформації про несанкціоновані дії користувачів (наприклад, доступ до недозволених ресурсів, розсилка спаму, перехід по фішингових посилань) і активності мережі на рівні додатків.

Більш того, IBM також надає підписку на IBM X-Force Threat Intelligence. Дана опція надає список IP-адрес, які можуть нести загрозу і є потенційно шкідливими.

Розгорнути QRadar SIEM можна в локальних і хмарних середовищах.

Переваги IBM QRadar

- Функціональні можливості багато в чому визначають основні переваги для впровадження IBM QRadar на підприємстві. Слід виділити наступні переваги:
- Проста конфігурація і розгортання на локальних ресурсах і в хмарної середовищі.
- Відображення подій в реальному часі або за результатами минулих періодів.
- Комплексне виявлення інцидентів і управління вразливостями.
- Сильні аналітичні можливості завдяки яким QRadar використовує контекст для підозрілих інцидентів, що призводить до масивного скорочення даних і більш високій швидкості виявлення інцидентів.
- Можливість надавати аналіз поведінки і відхилень даних мережевого потоку і журналів.
- Дані потоку (мережевий трафік) і дані подій об'єднані в єдину панель, що дозволяє користувачам точно визначати пріоритети даних про інциденти, зменшуючи кількість помилкових спрацьовувань.

IBM Security App Exchange дозволяє клієнтам, діловим партнерам та іншим розробникам створювати додатки, що розширюють можливості QRadar.

Інтегрується з усіма відповідними продуктами IBM і багатьма сторонніми постачальниками.

Недоліки IBM Qradar

- Втрата подій при значному підвищенні ліцензійних метрик
- Немає можливості збагачувати подію додатковою інформацією
- Слабка система візуалізації і побудови звітів
- Немає можливості редагувати вже обрані фільтри при пошуку, незручно реалізований пошук подій(в порівнянні зі Splunk чи Kibana).

1.2.2. HP ArcSight

Основою продуктової лінійки HP ArcSight[5] є комплексне рішення HP ArcSight Security Intelligence, ядром якого служить продукт HP ArcSight ESM (Enterprise Security Manager). Даний продукт забезпечує збір, обробку та зберігання подій безпеки, які можуть надходити від різних джерел. HP ArcSight ESM підтримує інтеграцію з великою кількістю прикладних систем і пристроїв (більше трьох сотень) і поставляється з декількома сотнями встановлених правил кореляції. До складу поставки також може входити унікальний агент FlexConnector, що дозволяє здійснювати інтеграцію з будь-яким типом програми.

Склад SIEM-системи HP ArcSight Security Intelligence включає в себе:

- HP ArcSight Logger забезпечує економічно ефективне універсальне рішення для управління журналами, яке об’єднує пошук, звітування, оповіщення та аналіз для будь-якого типу даних корпоративних машин;
- HP ArcSight Threat Response зменшує весь план реагування з годин на секунди, забезпечує моментальну реакцію на інциденти шляхом аналізу інформації від HP ArcSight ESM, локалізацію проблеми і застосування відповідних заходів реагування;
- HP ArcSight Configuration Management дозволяє провести конфігурацію мережевого обладнання та налаштувань безпеки;

- HP ArcSight Fraud Detection унікальне рішення для виявлення та запобігання шахрайству в області інтернет-банкінгу і банківських (пластикових) карт..

Технологія функціонування ArcSight передбачає поділ процесу обробки подій безпеки на п'ять основних етапів: фільтрація, нормалізація, агрегування, кореляція і візуалізація. В процесі фільтрації система видаляє події, які не мають прямого відношення до інцидентів інформаційної безпеки. На етапі нормалізації події приводяться до єдиного формату повідомлень ArcSight. Агрегування дозволяє видалити повторювані події, що описують один і той же інцидент. Ця процедура дозволяє значно скоротити обсяг інформації, яка зберігається і обробляється в системі моніторингу. Сформовані повідомлення потім обробляються, використовуючи механізми кореляції, засновані на статистичних методах, а також правила вбудованої експертної системи. І нарешті, ArcSight видає отримані результати на централізовану консоль, що працює в режимі реального часу.

Переваги HP Arcsight

Лідируюче рішення в області моніторингу та управління подіями безпеки (SIEM). Найбільша частка ринку SIEM за звітами IDC. Перше місце серед SIEM рішень протягом 6 років за даними Gartner. За звітами TheInfoPro рішення HP ArcSight є найбільш широко вживаними серед рішень з інформаційної безпеки (SIEM) і управління журналами подій.

Найбільш широкий набір можливостей з моніторингу та управління подіями інформаційної безпеки. Весь необхідний функціонал в одному рішенні: від збору подій і приведення їх до єдиного стандарту, до її обробки та довготривалого зберігання зі зручним пошуком по будь-яким параметрам. А також передналаштовані комплекти правил для контролю відповідності міжнародним стандартам.

Безпрецедентні можливості масштабування, доведені на практиці. Вже є реалізовані проекти, де HP ArcSight ESM справляється з обробкою декількох десятків мільйонів подій безпеки в день!

Найбільш широкий спектр підтримуваних пристроїв і додатків. На даний момент розроблені коннектори для більш ніж 300 пристроїв і додатків більш ніж 200 вендорів. Завдяки FlexConnector SDK можна створити коннектор для додатків і пристроїв, що не входять в цей список.

Вибір варіанту придбання. Завдяки можливості придбання рішення у вигляді ПО або ж передналаштовуваного програмно-апаратного комплексу, замовник може обрати рішення, яке найбільш точно задовольняє його поточним потребам, не переплачуючи за невикористані потужності. Можливість апгрейда дозволить швидко прирости продуктивність тоді, коли це буде необхідно.

Можливість спрощеного розгортання та управління. Завдяки HP ArcSight Express користувач відразу отримує готове до роботи рішення з передналаштованим набором найпоширеніших правил, сигналів загроз і звітів. Дане рішення дозволяє реалізувати повноцінний моніторинг і управління подіями безпеки навіть в компаніях з невеликим штатом IT-фахівців і фахівців з безпеки.

У ArcSight Express в ліцензуванні беруть участь три ключові параметри: кількість подій в секунду(EPS), кількість мережевих потоків (FPM), а також кількість джерел подій. Всього три окремо ліцензованих опції. У ArcSight ESM ядро системи ліцензується за обсягом логів в день (продуктивність). Крім ядра необхідно ліцензувати набір різних параметрів і опцій, наприклад: кількість користувачів, ліцензія на розробку власних конекторів, кількість джерел подій (Вважається окремо за типами джерел), модулі відповідності вимогам, log management і т.д.

Недоліки HP Arcsight

Дослідницька фірма зазначає, що кілька елементів архітектури ArcSight оновлювались до придбання Micro Focus, тому майбутні користувачі повинні переконатися, що Micro Focus і надалі виконуватиме ці зобов'язання щодо вдосконалення функціональності та підтримки.

Деякі з цих змін передбачали впровадження ADP, Investigate та інших компонентів для підтримки розширеної аналітики, підтримуючи при цьому

застарілі функції. "Як результат, вибір замовником щодо розгортання деяких елементів рішення може призвести до дублювання даних".

1.2.3. Splunk Enterprise Security

Система SIEM Splunk має високу оцінку та популярність, але витрати на ліцензування можуть вивести її за межі досяжності деяких МСП. Це найкраще для великих, добре укомплектованих IT-організацій, готових заплатити ціну за високу ефективність безпеки. Gartner за останні кілька років оцінив його як лідера у своєму магічному квадраті SIEM[6], і він продовжує оцінюватись як один з наших найкращих продуктів SIEM.

Флагманська технологія Splunk SIEM, Enterprise Security (ES), показує оригінальність Splunk в аналітиці. Він інтегрується з аналітикою поведінки користувачів (UBA) компанії, набором інструментів машинного навчання та автоматизацією та реагуванням Phantom Security Orchestration (SOAR). Splunk Enterprise Security підтримує всі основні та вдосконалені функції SIEM, а також оркестрацію та автоматизацію інструментів у безпеці та IT-екосистемі, а також аналітику з аномалією та виявленням загроз на основі машинного навчання.

Splunk ES - це керований аналітикою SIEM, який дозволяє командам безпеки виявляти, досліджувати та реагувати на внутрішні та зовнішні атаки, а також спрощувати управління загрозами. Він централізує та агрегує всі події, що стосуються безпеки, у міру їх генерування з їх джерела. Крім того, він підтримує різноманітні механізми прийому / збору та забезпечує спеціальний пошук та звітування для аналізу порушень.

Переваги Splunk Enterprise Security

- Послідовність подій для оптимізації виявлення загроз та прискорення розслідувань
- Бібліотека випадків використання спрощує виявлення та реагування на випадки
- Investigation Workbench зменшує час на утримання та усунення загроз за допомогою централізації даних

- Інформаційна панель, надана для перегляду різних типів журналів, що генеруються
- Забезпечує розширений графічний інтерфейс користувача, завдяки чому кінцевий користувач зможе легко переміщатися по інструменту.
- Інформаційні панелі BI додають найбільше значення для splunk. Можна використовувати це для виправлення вразливості ін'єкцій SQL та журналів.

Недоліки Splunk Enterprise Security

- Рішення преміум класу, тому дуже складна та найвища вартість ліцензування

З повним описом роботи даного продукту можна ознайомитися за посиланням [7].

1.2.4. Порівняльна характеристика рішень SIEM

Для створення корисного інструменту вибору виділимо наступні групи критеріїв порівняння SIEM-систем:

- Загальна інформація - буде корисна при презентації керівництву або організації референс-візитів, відповідності основних показників IT та ІБ-стратегії компанії;
- архітектура рішення, масштабованість, методи управління подіями і схема ліцензування - важливий параметр для Enterprise-установок, де необхідно підрахувати кінцеву вартість володіння рішенням, враховуючи трудомісткість обслуговування;
- функціональні особливості, склад, змінювані параметри, гнучкість налаштування дозволяють оцінити придатність рішення до прийнятої парадигми розвитку процесів забезпечення ІБ (аутсорсинг, централізоване, розподілене використання) компанії. А якість і кількість встановлених з коробки елементів, а також середній час старту дадуть уявлення про терміни впровадження до отримання перших показників ефективності;
- інтеграційні можливості - наявність розвинених вбудованих і інтегрованих підсистем управління вразливостями, інцидентами і активами дозволить в початковому періоді експлуатації обмежитися використанням одного

продукту, без збільшення кількості використовуваних адміністратором і аналітиком консолей. А інтеграція зі сторонніми рішеннями з метою збагачення інформації про події ІБ, відомості про API і підтримуваних джерелах подій вказують на відкриту позицію компанії на ринку, вміння знаходити спільну мову з іншими гравцями, говорить про напрямки розвитку продукту.

Таблиця 1.1

Порівняльна характеристика існуючих рішень SIEM

Критерії		**IBM QRadar**	**HP ArcSight**	**Splunk**
Загальна інформація	Цільовий сегмент	Банківський, державний сектори, великий і середній бізнес	Всі сектори. Великий і середній бізнес	Всі сегменти у всіх галузях, від безкоштовних версій(до 500мб) до найбільших інсталяцій
	Терміни впровадження (на об'єкті з підключенням понад 300 джерел і налаштуванням 15 базових правил кореляції)	Від 1 місяця (залежить від ТЗ, команди виконавця, залучення замовника)	Від 1 місяця (залежить від ТЗ, команди виконавця, залучення замовника)	Від 2 тижнів (при своєчасної залученості замовника, зафіксованих рамках проекту)
Управління інцидентами, вразливимостями, активами	Інформація про інцидент	25 полів	55 налаштованих полів	0-236 штатних полів
	Оповіщення про інцидент (пошта, месенджери, SMS, інтеграції)	SMTP, скрипти	SMTP, SMS,API	Пошта, месенджери, скрипти, інтеграція зі сторонніми сервісами
	Автореєстрація вразливостей (інтеграція зі	Інтеграція з понад 20 сканерами, підтримка	Інтеграція з усіма популярними сканерами великих	Інтеграція зі сканерами по відкритих протоколах. Для

	сканерами)	формату AXIS	вендорів, можливості по інтеграції через API і звіти різних форматів	популярних сканерів є модулі розбору подій (Qualys, Netxpose Rapid7 і ін.)
	Управління правилами кореляції	Об'єктний конструтор	Об'єктний конструтор, мова AQL	Графічний конструктор
Системна архітектура	Обмеження за кількістю оброблюваних подій в секунду	Залежить від ліцензії EPS	ArcSight Connector - 2-4К EPS, Event Broker - більш 500к EPS, ArcSight ESM - 50к EPS	Лімітовані лише апаратними обмеженнями
	Метод збору подій з джерел	Агентський і без агентський	Агентський і без агентський	Агентський і без агентській
Підключення джерел подій	Кількість підтримуваних джерел подій	300+	300+	2000+
Інтеграційні можливості, збагачення даними з інших систем	Використання технологій штучного інтелекту, автоматизації аналітики верхнього рівня	QRadar Advisor With Watson	У ESM немає, є виділене рішення для аналітики - Investigate	Вбудований в Splunk ES 5.0. + Функціональність Workbench Investigator
Ліцензування	Метрики ліцензування - модульність	Кількість подій - EPS. Кількість flow - FPM. Модулі системи. Ліцензії на 1 рік і більше	ArcSight Logger - Raw Гб / день; ArcSight ESM - EPS (події в секунду); ArcSight Investigate - Гб / день; ArcSight UBA - по користувачах	Ліцензування за обсягом зібраних даних в день: ГБ / день. Ліцензії річні і безстрокові. Сервера / ресурси / інсталяції не ліцензуються

1.3. ELK Stack

Стек ELK - це абревіатура, яка використовується для опису стека, що складається з трьох популярних проектів з відкритим кодом: Elasticsearch, Logstash та Kibana. Стек ELK, який часто називають Elasticsearch, надає можливість об’єднувати журнали з усіх ваших систем та додатків, аналізувати ці журнали та створювати візуалізації для моніторингу програм та інфраструктури, усунення несправностей, аналітику безпеки тощо.

Потужні можливості ELK-інфраструктури дозволяють не тільки аналізувати системні логи в рамках завдання адміністрування корпоративного IT-ландшафту. Також ця Big Data система відмінно підходить для вирішення наступних бізнес-задач:

- агрегація товарів з безлічі інтернет-магазинів, фільтрація і пошук різних властивостей продукції;
- агрегація даних з різних систем, обчислення і відображення показників для комплексного аналізу бізнес-процесів;
- перегляд і аналіз різної неструктурованої статистичної інформації;
- автоматизована обробка анкет та опитувальників.

Далі розглянемо докладніше, з чого складається ELK-система і як вона влаштована[8].

E = Elasticsearch

Elasticsearch - це механізм розподіленого пошуку та аналітики з відкритим кодом, RESTful, побудований на Apache Lucene. Підтримка різних мов, висока продуктивність та безсистемні документи JSON роблять Elasticsearch ідеальним вибором для різних аналітичних журналів та випадків використання пошуку.

L = Logstash

Logstash - це інструмент передачі даних із відкритим кодом, який дозволяє збирати дані з різних джерел, трансформувати їх і відправляти до потрібного місця призначення. Завдяки вбудованим фільтрам та підтримці понад 200 плагінів, Logstash дозволяє користувачам легко вводити дані незалежно від джерела даних або типу. Вивчайте більше "

K = Кібана

Kibana - це інструмент візуалізації та дослідження даних із відкритим кодом для перегляду журналів та подій. Kibana пропонує прості у використанні, інтерактивні діаграми, заздалегідь побудовані агрегації та фільтри, а також геопросторову підтримку, що робить його кращим вибором для візуалізації даних, що зберігаються в Elasticsearch.

Звичайно, стек ELK є відкритим. Оскільки IT-організації віддають перевагу продуктам з відкритим кодом, одне лише це може пояснити популярність пакета. Використання відкритого коду означає, що організації можуть набагато легше уникнути блокування постачальників. Відкритий код також означає активну спільноту, яка постійно впроваджує нові функції та інновації та допомагає у разі потреби.

В основі будь-якої системи SIEM лежать дані журналу. Незалежно від серверів, брандмауерів, баз даних або мережевих маршрутизаторів - журнали надають аналітикам вихідну інформацію для отримання розуміння подій, що відбуваються в IT-середовищі.

Дані потрібно збирати, обробляти, нормалізувати, вдосконалювати та зберігати. Ці етапи, зазвичай згруповані під терміном “управління журналами”, є обов’язковим компонентом будь-якої системи SIEM.

Якби управління журналами та аналіз журналів були єдиними компонентами SIEM, стек ELK міг би вважатися дійсним рішенням із відкритим кодом. Але посилаючись на джерело [9] визначаємо, що насправді є системою SIEM, на додаток до управління журналами було перераховано повний список компонентів:

- збирання логів;
- обробка подій;
- зберігання та архівування;
- пошук;
- кореляція подій;
- сповіщення про загрози;

- управління інцидентами.

Тому необхідно глибше заглибитися в питання про те, чи можна використовувати стек ELK для побудови SIEM, чого не вистачає та що потрібно для його перетворення у повнофункціональне рішення SIEM.

Збирання логів

Як вже згадувалося вище, системи SIEM включають агрегування даних з декількох джерел даних. Ці джерела даних будуть різнитися залежно від наданої ІКС, але, швидше за все, необхідно буде витягнути дані про ІКС: рівень інфраструктури (наприклад, сервери, бази даних), засоби управління безпекою (наприклад, брандмауери, віртуальні приватні канали), мережеву інфраструктуру(наприклад, маршрутизатори, DNS), зовнішні бази даних безпеки (наприклад, канали потоків).

Для цього потрібні можливості агрегування, з якими працює один із компонентів стеку ELK . За допомогою комбінації Beats та Logstash надається можливість побудувати архітектуру ведення журналу, що складається з декількох конвеєрів даних. Beats - це легкі пересилачі журналів, які можуть бути використані як агенти на крайніх хостах для відстеження та пересилання різних типів даних, найпоширенішим є Filebeat для пересилання файлів журналів. Потім Logstash можна використовувати для агрегування даних, їх обробки та пересилання до наступного компонента в конвеєрі.

Через кількість залучених даних та різні джерела даних, які використовуються, для забезпечення більш еластичного конвеєру даних, швидше за все, знадобляться кілька екземплярів Logstash. Мало того, потрібно буде розгорнути механізм черги, щоб переконатися, що обробляються підключені пакети даних, а роз'єднання між різними компонентами конвеєра не призводять до втрати даних. Kafka часто є інструментом, використовуваним у цьому контексті, встановленим перед Logstash (також використовуються інші інструменти, такі як Redis та RabbitMQ).

Отже, лише стеку ELK, швидше за все, буде недостатньо, оскільки ваш бізнес зростає, і дані, які він генерує. Організація, яка займається використанням

ELK для побудови SIEM, повинна розуміти, що для збільшення стека потрібно буде розгорнути додаткові компоненти.

Обробка журналу

Збір даних та їх пересилання - це, звичайно, лише одна частина роботи, яку Logstash виконує в конвеєрі реєстрації. Іншим важливим завданням, і надзвичайно важливим у контексті SIEM, є обробка та синтаксичний аналіз даних.

Усі зазначені вище типи джерел даних генерують дані у різних форматах. Щоб досягти успіху на наступному кроці - пошуку даних та їх аналізі - дані потрібно нормалізувати. Це означає розбиття різних повідомлень журналу на значущі імена полів, правильне відображення типів полів у Elasticsearch та збагачення конкретних полів, де це необхідно.

Не можна перебільшувати важливість цього кроку. Без правильного аналізу дані компанії будуть позбавлені сенсу під час будь-якої спроби їх аналізу. Logstash - це потужний інструмент для вирішення цього ключового завдання. Підтримуючи велику кількість різних плагінів для фільтрів, Logstash може розбивати журнали подій ІБ, збагачувати певні поля географічною інформацією, наприклад, приховувати поля, додавати поля тощо.

Знову ж таки, архітектура обробки журналів подій, така як та, що вимагається системою SIEM, може ускладнитися. Зокрема, налаштування Logstash для обробки різних типів журналів потребуватиме декількох файлів конфігурації Logstash та екземплярів Logstash. Важка обробка, результат складних конфігурацій фільтрів, впливає на продуктивність Logstash. Моніторинг конвеєрів Logstash є важливим, і для цього доступний програмний інтерфейс моніторингу, такий як Hot Thread API[10] для ідентифікації потоків Java з високим навантаженням.

Зберігання та архівування

Дані журналу, зібрані з різних джерел даних, потрібно зберігати в сховищі даних. У випадку з ELK Elasticsearch виконує цю роль індексації та зберігання даних.

Elasticsearch сьогодні є однією з найпопулярніших баз даних - це друге за завантаженням програмне забезпечення з відкритим кодом після ядра Linux. Ця популярність пов’язана з різними причинами - вона з відкритим кодом, відносно проста у налаштуванні, швидка, масштабована та підтримується величезною спільнотою.

Звичайно, розгортання кластеру Elasticsearch - це лише перший крок. Оскільки ми говоримо про індексування великих наборів даних, які з часом, швидше за все, збільшаться в обсязі, будь-яке розгортання Elasticsearch, що використовується для SIEM, повинно бути надзвичайно масштабованим та стійким до несправностей.

Для цього потрібна низка конкретних підзавдань. Вже згадувалось про використання механізму черги, щоб переконатися, що дані не втрачаються у разі відключень або сплесків даних, але також потрібно буде стежити за ключовими показниками продуктивності Elasticsearch, такими як швидкість індексації, JVM вузла та процесору. Знову ж таки, є API моніторингу, який можна використовувати для цієї мети. Планування потужності також важливо, і якщо ІКС розгорнута за допомогою хмарних технологій, швидше за все, буде потрібна політика автоматичного масштабування, щоб забезпечити достатньо ресурсів для індексації.

Іншим фактором є зберігання/архівація.

Для ефективної судової експертизи та розслідування знадобиться довгострокова стратегія зберігання. Наприклад, якщо буде помічено великий стрибок трафіку, що походить від певної IP-адреси, потрібно порівняти ці історичні дані, щоб перевірити, чи це аномальна поведінка. Деякі атаки можуть повільно розвиватися протягом місяців і для аналітика ці історичні дані є ключовими для успішного виявлення закономірностей і тенденцій.

Стек ELK не підтримує нестандартну можливість архівування, тому потрібно буде з’ясувати архітектуру для збереження даних самостійно. В ідеалі такий, який фінансово буде доступний для компанії.

Пошук

Після того, як ваші дані зібрані, проаналізовані та проіндексовані в Elasticsearch, наступним кроком є запити даних. Ви можете зробити це за допомогою веб інтерфейсу протоколу HTTP Elasticsearch, але, швидше за все, ви будете використовувати для цього Kibana.

У Кібані запит даних здійснюється за допомогою синтаксису Lucene. Наприклад, загальним типом пошуку є пошук на рівні поля. Наприклад, скажімо, що є завдання пошуку всіх повідомлень журналу, породжені діями, виконаними певною особою в організації. Оскільки поле нормалізоване з назвою ім'я користувача для всіх джерел даних, надана можливість використовувати цей простий запит:

username: "spitkovskaav"

Також можна використовувати цей тип пошуку з логічним твердженням, таким як AND, OR, NOT.

Знову ж таки, якщо задача використовувати стек ELK для побудови SIEM, потрібно буде використовувати потужність синтаксичного аналізу Logstash для обробки даних компанії - і наскільки добре це вдасться вплине на те, наскільки легкий та швидко виконаний запит у кількох джерелах даних, які ви шукали, відбудеться.

Візуалізація

Kibana славиться своїми можливостями візуалізації, підтримуючи широкий спектр різних типів візуалізації та дозволяючи користувачам розділити та відобразити дані будь-яким способом, який їм подобається. Надана функція створювати кругові діаграми, графіки, географічні карти, окремі метрики, таблиці даних тощо, та візуалізувати результати є досить ефективною.

Створення інформаційних панелей у Kibana - непросте завдання, воно вимагає глибокого знання даних подій та різних полів, що створюють повідомлення журналу. Більше того, у Kibana відсутні конкретні можливості, такі як динамічне зв’язування в рамках візуалізації. Є обхідні шляхи, але вбудована функціональність буде величезним бонусом.

Kibana також не підтримує безпечне спільне використання об'єктів. Якщо буде виявлено порушення правил ІБ і бажання поділитися інформаційною панеллю або єдиною візуалізацією з колегою, посилання на спільний доступ у Kibana не токенізується. Є комерційні доповнення, які можна застосувати на версії Kibana (X-Pack) або рішення з відкритим кодом, які можна використовувати.

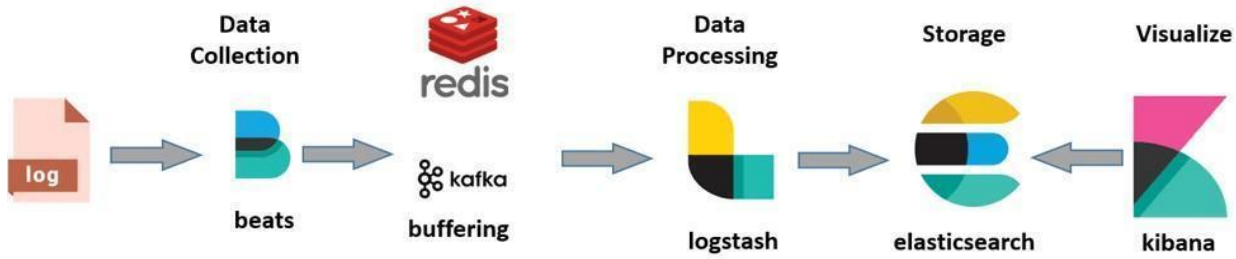

Рис.1.1. Схема роботи ELK Stack

Кореляція

Іншим ключовим компонентом SIEM є кореляція подій. Кореляція подій, як це вже визначили в попередньому дописі, - це зв’язок сигналів, що надходять з різних джерел даних, у шаблон, який може свідчити про порушення безпеки. Правило кореляції визначає конкретну послідовність подій, яка формує цю закономірність.

Наприклад, може бути створено правило, яке визначає, коли більше певної кількості запитів надсилається із певних діапазонів IP та портів протягом певного часу. Іншим прикладом правила кореляції буде пошук ненормальної кількості невдалих авторизацій разом зі створенням привілейованих облікових записів.

Ці правила кореляції надаються різними інструментами SIEM або заздалегідь визначені для різних сценаріїв атак. Звичайно, стек ELK не має вбудованих правил кореляції, тому аналітик повинен використовувати запити Kibana на основі аналізу та обробки, що виконуються за допомогою Logstash, для кореляції між подіями. І для автоматизації процесу аналізу подій за певним набором правил створюється програмний модуль аналізу подій.

Сповіщення

Правила кореляції нічого не означають без попереджень. Будучи попередженим, коли ідентифікується можлива схема атаки, є ключовим компонентом SIEM.

Продовжуючи наведені вище приклади, якщо система реєструє велику кількість запитів із певного діапазону IP або ненормальну кількість невдалих входів, попередження потрібно надіслати потрібній особі чи команді в організації. Суть у швидкості - чим швидше надсилається повідомлення, тим більше шансів на успішне пом’якшення наслідків атаки.

Стек ELK у своїй формі з відкритим кодом не постачається із вбудованим механізмом попередження. Щоб додати цю можливість, стек ELK потрібно доповнити плагіном для попередження або надбудовою. Знову ж таки, один із варіантів - X-Pack(який не надається безкоштовно). Інший варіант - додати ElastAlert - фреймворк з відкритим кодом, який можна додати поверх Elasticsearch.

Управління інцидентами

Проблему виявлено, аналітик попередив. Що тепер? Наскільки добре ваша організація реагує на інцидент, визначатиме результат. SIEM побудована, додаткові модулі знайдено або розроблено, щоб допомогти в наступних кроках аналітики ІБ - виявляти інцидент, ескалацію, якщо потрібно, пом'якшення та сканування на наявність вразливостей.

Стек ELK чудово допомагає аналітику визначити інцидент, але не має багато чого запропонувати для управління ним. Навіть якщо надбудова для попередження реалізована поверх стека, для ефективного управління інцидентами потрібен спосіб управління опрацьованими попередженнями. В іншому випадку існує ризик потонути в попередженнях і пропасти під час важливих подій. Автоматизація процесу ескалації та створення інцидентів загроз також важлива для ефективної обробки подій.

У сирому вигляді, що складається з Logstash, Elasticsearch, Kibana та Beats - стек ELK НЕ є рішенням SIEM. Але з додатковими модулями аналізу та кореляції подій, сповіщенням та управління інцидентами(чого не має у

функціоналі стеку) може точно замінити дорогі комерційні ліцензійні рішення продуктів SIEM.

1.4. Висновки до розділу

SIEM необхідна для автоматизації процесу виявлення шкідливої активності і різних системних аномалій. Робота SIEM дозволяє побачити більш повну картину активності мережі та подій безпеки. Коли звичайні засоби виявлення окремо не бачать атаки, але APT може бути виявлена при ретельному аналізі і кореляції інформації з різних джерел. Тому багато організацій розглядають використання SIEM-системи в якості додаткового і дуже важливого елемента захисту від цілеспрямованих атак.

Більш того, без SIEM неможливо побудувати такі системи і центри моніторингу та реагування, як SOC (Security Operation Center), так як SIEM допомагає вирішувати цілий ряд ключових завдань: збирати і зберігати лог-файли в єдиному централізованому сховищі, надавати спеціалізовані звіти аудиторів для відповідності вимогам законодавства, галузевим стандартам і виконувати кореляцію подій між різними джерелами. Слід приділити особливу увагу налаштуванню SIEM під клієнта, його інфраструктуру і системи безпеки. Добре налаштовані правила кореляції дозволять оператору аналізувати дійсно важливі повідомлення про інциденти, відсіваючи зайве.

Інформація про безпеку та системи управління подіями (SIEM) надають додаткові значні витрати бюджетам компаній. За даними IT-аудиторської компанії Netwrix Corporation, 69 відсотків компаній прагнуть зменшити витрати на SIEM. Опитування під назвою “Огляд ефективності SIEM 2016 року”[11] вивчало думки 234 великих підприємств, що використовують рішення SIEM для моніторингу безпеки та IT-інфраструктури.

Результати показали, що SIEM є досить дорогим інструментом. Окрім того для обслуговування SIEM необхідний найм та навчання аналітиків та адміністраторів SIEM.

У стандартному складі з Logstash, Elasticsearch, Kibana та Beats - стек ELK не являє собою рішення SIEM. Але є одним із потужних OpenSource проектом з такими функціями як:

- збирання логів;
- обробка подій;
- зберігання та архівування;
- пошук.

А з додатковими модулями аналізу та кореляції подій, сповіщенням та управління інцидентами(чого не має у функціоналі стеку) може точно замінити дорогі комерційні ліцензійні рішення продуктів SIEM.

РОЗДІЛ 2. ДОСЛІДЖЕННЯ ТА АНАЛІЗ ПІДХОДІВ ДО КОРЕЛЯЦІЇ, АРХІТЕКТУРИ ТА ЗАВДАНЬ МОДУЛЮ АНАЛІЗУ ПОДІЙ ІБ

2.1. Порівняльний аналіз підходів до кореляції подій ІБ

Основними завданнями методик кореляції є:

1) зниження обсягу вихідного потоку подій безпеки за рахунок групування взаємопов'язаних подій, що дозволяє знизити когнітивну навантаження на аналітика;

2) визначення взаємозв'язків між подіями від різнорідних джерел, що сприяє кращому розумінню розвитку атаки в інформаційній системі;

3) кореляція подій в контексті системи, що дозволяє краще зрозуміти сценарій атаки, її мети і завдання. Існують різні схеми класифікації розроблених методик кореляції подій, що незначно відрізняються один від одного. В цілому можна виділити три основні групи методик кореляції подій виходячи з їх особливостей реалізації і вирішення за допомогою завдань кореляції:

- на основі подібності (подібності) подій;
- на основі знань;
- імовірнісні (або статистичні)

Алгоритми кореляції подій ІБ на основі подібності

Алгоритми на основі подібності подій ІБ базуються на обчисленні показника подібності, що дозволяє виконати порівняння двох подій безпеки або однієї події з групою подій безпеки. якщо рівень подібності перевищує або дорівнює деякому заданому граничному значенню, то події об'єднуються в одне метаподія. Таким чином, метою цих алгоритмів є агрегування подій у часі або побудова узагальнюючих ієрархій подій. основною перевагою алгоритмів цієї групи є відсутність необхідності точного визначення типу атак, і кореляція події може бути виконана тільки на визначенні показника подібності для атрибутів подій безпеки.

Визначення подібності на основі правил

Основна ідея цього підходу полягає в застосуванні досить простих правил для опису взаємозв'язків між атрибутами подій, які можуть бути пов'язані між собою.

На рис. 2.1. представлена загальна схема кореляції подій ІБ на основі правил[12].

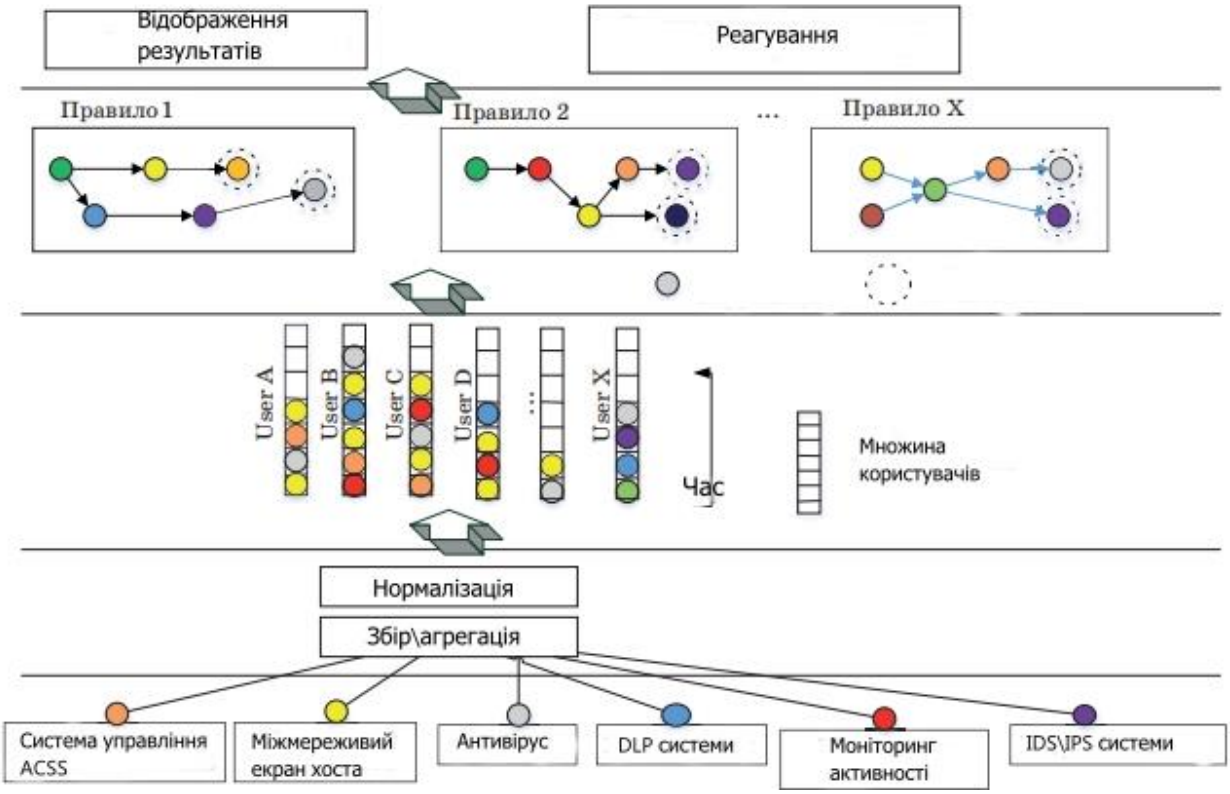

Рис. 2.1. Загальна схема кореляції подій безпеки на основі правил

На першому етапі формується безліч правил, які встановлюють зв'язки між подіями безпеки для користувачів в системі, має певну роль в інформаційній системі. Далі при появі контрольованих подій спрацьовують певні правила, які в свою чергу можуть ініціювати спрацьовування інших правил, на основі яких робиться висновок про виконання атаки певного типу. Кінцевий результат кореляції подій полягає в застосуванні контрзаходів, заданих для кожного типу атаки, і його уявленні в графічному вигляді адміністратору мережевої безпеки. Зазвичай описують правила взаємозв'язку між атрибутами даних трьох рівнів[13]:

- рівень даних (data level) - правила, що працюють безпосередньо з сирими даними;
- рівень знань (knowledge level) - правила, що описують специфіку предметної області і дозволяють працювати з метаподіями вищого рівня;
- рівень управління (control level) - правила, безпосередньо описують логічний

висновок на основі подій безпеки, т. е. це «Серце» модуля кореляції.

Очевидно, що ефективність таких систем залежить тільки від якості застосовуваних правил.

Визначення подібності на основі кодової книги

Клігер і ін.[14] представили систему, метою якої є локалізація виникаючих проблем в системі на основі вибору деякого відповідного підмножини подій-«симптомів», пов'язаних з цими проблемами. підмножина подій-«симптомів» і є змістом кодової книги. Для кожної проблеми створюється деякий двійковий вектор, який визначає, є деяка подія ознакою деякої проблеми чи ні, і записується в кодову книгу. Для виявлення проблем всі події, представлені в кодовій книзі, відстежуються в режимі реального часу. При настанні деякої події вектор події порівнюється з безліччю векторів з кодовою книги, вибирається вектор, у якого відстань

Хемминга між ним і вектором події є мінімальним. Завдяки такому рішенню система завжди видає деяке припущення про можливу проблему. очевидним недоліком даного підходу є неможливість врахувати час між настанням двох різних подій, що є важливим параметром при встановлення часових зв'язків між подіями.

Визначення подібності подій з використанням алгоритмів машинного навчання

В останню категорію увійшли алгоритми, в яких міра подібності між подіями визначається автоматично за допомогою алгоритмів машинного навчання. В основному для вирішення цього завдання використовуються алгоритми класифікації з роботи [15] і нейронні мережі[16]. Так, наприклад, для кластеризації подій використовуються дерева рішення[15]. Причому даний алгоритм може застосовуватися двічі - для визначення схожих подій безпеки і для визначення наслідків атаки. Для коректної побудови дерева рішень і, відповідно, коректного функціонування самого алгоритму кореляції подій потрібно навчальна вибірка значного розміру, що містить можливі сценарії атак.

Поява нових даних, що описують нові сценарії атак, вимагає перенавчання моделі аналізу, що значно знижує гнучкість і розширюваність алгоритму.

Класифікація подій безпеки здійснюється також за допомогою нейронної мережі і кластеризації, завдяки чому знижується число попереджень, які потребують ручної обробки. Однак відзначається, що через відсутність прозорості у функціонуванні та навчанні нейронних мереж вони не дуже популярні для побудови інструментів кореляції подій.

Алгоритми на основі знань

До цієї категорії відносяться алгоритми кореляції знань, які функціонують на за допомогою опису можливих сценаріїв атак або принципів функціонування контрольованої системи. Вхідні в цю категорію алгоритми можна розділити на дві підгрупи.

Алгоритми на базі передумов і наслідків

Алгоритми даної підкатегорії відстежують значення виникають подій безпеки, оцінюють стан мережі, після чого діагностують наявність вторгнення або будь-якої іншої проблеми. Для того щоб встановити взаємозв'язку між різними етапами атак, виразами у вигляді ланцюжка деяких подій безпеки, передбачається використання

1) бази даних, що описують топологію мережі, конфігурацію її вузлів;

2) бази знань, яка для кожної події безпеки містить опис його всіх можливих передумов і наслідків його настання[17-19].

Зазвичай в цих алгоритмах події моделюються за допомогою логіки першого порядку.

Результат їхньої роботи може бути представлений у вигляді графа можливих подій і зв'язків між ними. В роботі [9] запропоновано розширення даного підходу для визначення елементів атак, що не були діагностовані датчиками безпеки. Сценарій атаки (рис. 2.2.), побудований за допомогою системи, представлена в роботі [19] та відображається у вигляді зв'язного графа, вершинами якого є атакуючі дії зловмисника, ребра позначають можливу послідовність появи таких подій.

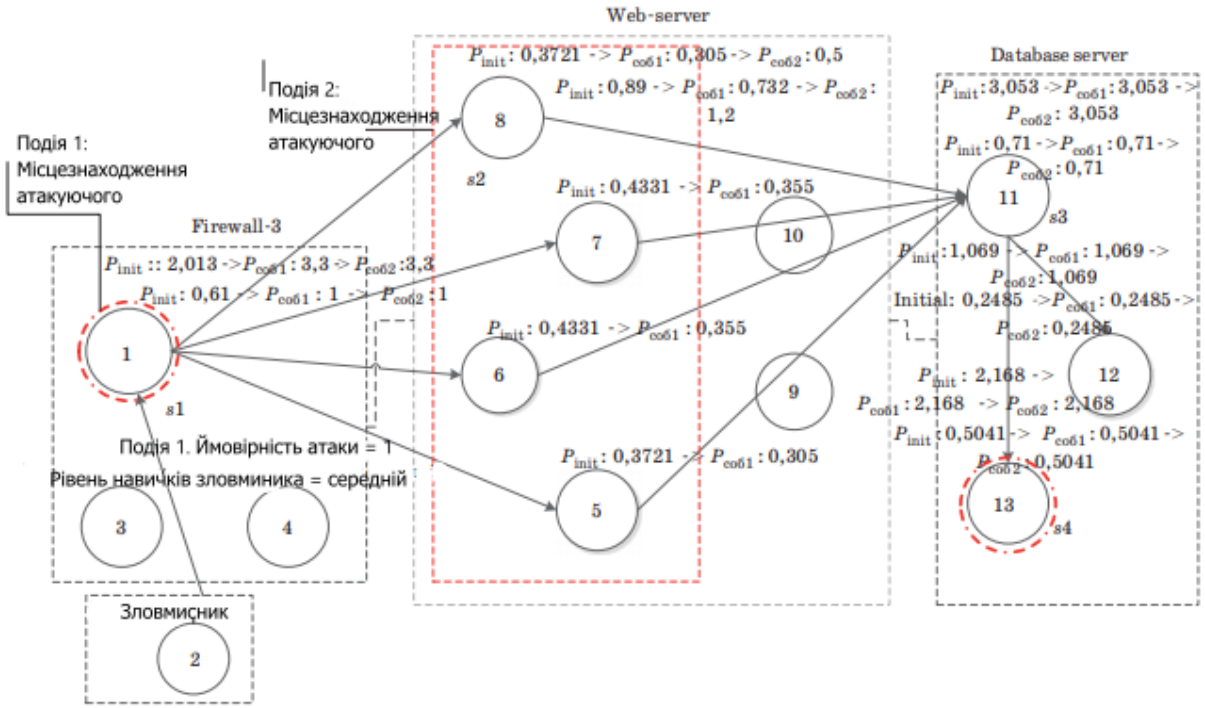

Рис. 2.2. Приклад сценарія атаки

Можливість виконання атакуючого дії пов'язана з наявністю вразливостей програмного забезпечення, складність експлуатації яких визначає вихідну ймовірність їх настання. Для кожної події вихідна ймовірність його настання вказана як Pinit. вершини графа (s1-s4) відображають послідовність дій зловмисника, який володіє середнім рівнем технічних навичок. ймовірності Pсоб1 і Pсоб2 позначають ймовірності настання подій після того, як в системі було зареєстровано події 1 і 2, що відбулися на міжмережевому екрані Firewall-3 і на веб-сервері Webserver відповідно. Штрихпунктирною лінією на графі виділені вузли, що позначають початкове положення зловмисника (вершина 1) і найбільш ймовірне кінцева подія в результаті атаки (вершина 13).

Алгоритми на основі сценаріїв атак

Основним призначенням цієї множини алгоритмів є визначення багатокрокових атак. Для опису сценаріїв атак запропоновані різні мови, проте їх загальна ідея полягає в описі етапу атаки, необхідних умов для його виконання та цілі. Алгоритми цієї групи оперують знаннями вищого рівня по порівняно з алгоритмами на основі передумов і наслідків, оскільки останні функціонують на рівні подій безпеки. Приклади таких алгоритмів представлені в роботах [20-23].

Імовірнісні алгоритми кореляції подій безпеки

В основі цієї групи алгоритмів лежить припущення, що атаки мають загальне статистичний розподіл атрибутів, і коректна класифікація атак може бути виконана на основі оцінки розподілу значень атрибутів в мережевому трафіку. У загальному випадку алгоритми цієї групи формують базу причинно-наслідкових зв'язків між різними подіями безпеки, аналізуючи частоту їх виникнення в контрольованій системі під час періоду навчання системи і ладу можливих сценаріїв атаки.

Непараметричні статистичні алгоритми не вимагають ніякої апріорної інформації про можливі сценарії атак. Алгоритми цієї групи

також можуть бути розділені на дві підгрупи.

Алгоритми першої підгрупи будують статистичну модель мережевого трафіку, а алгоритми другої підгрупи оцінюють причинно-наслідковий зв'язок між подіями безпеки.

Побудова статистичної моделі мережевого трафіку

Метою алгоритмів[24] є створення статистичної моделі мережевого трафіку, прогнозування і виключення передбачуваних ситуацій. Важливою особливістю даних алгоритмів є можливість діагностування подій, які виникають періодично через ймовірних некоректних мережевих налаштувань або політик безпеки. Алгоритми цієї групи зазвичай не вимагають даних про контрольованої системи, їх функціонування визначається статистичними даними, сформованими на етапі навчання.

У більшості випадків навчання здійснюється в режимі реального часу, тому підстроювання алгоритмів до змін в конфігурації інформаційної системи здійснюється досить просто і гнучко.

Представлений підхід[25] до кореляції подій безпеки, в основі якого лежать алгоритми пошуку асоціативних правил для виявлення подій безпеки, які зазвичай виникають спільно. Важливою особливістю даної підгрупи алгоритмів є визначення пріоритетів подій безпеки на основі того факту, чи є виявлена комбінація подій характерною для даної системи або являє собою новий патерн

атаки. Крім того, алгоритми пошуку асоціативних правил можуть бути використані для формування пов'язаних метаподій безпеки.

Оцінка причинно-наслідкового зв'язку між подіями

В основі алгоритмів цієї підгрупи лежить побудова можливої моделі, яка визначає кореляційні зв'язки між подіями безпеки. Наприклад в роботах [26-28], побудова причинно-наслідкових зв'язків між подіями здійснюється шляхом оцінювання впливу заданого події в процесі передбачення появи інших подій безпеки.

Особливість запропонованих методик полягає в тому, що для побудови моделей аналізу не потрібна додаткова інформація про контрольованої системи. Як і в попередній підгрупі алгоритмів, для отримання точної та надійної моделі потрібна велика кількість історичних даних зі сценаріями атак.

Для побудови імовірнісних моделей використовуються байєсовські мережі. Для визначення імовірнісних залежностей між подіями безпеки використовують приховані марковські ланцюги.

Відмінною характеристикою запропонованого алгоритму є можливість оцінки ймовірності кожного сценарію атак і виконання кожного етапу атаки на основі попередніх кроків.

Ймовірності оцінюються на основі навчальної вибірки, наприклад, історичних даних про атаках на контрольовану систему, тому для навчання моделі аналізу потрібна велика кількість даних, що містять правильно категоризовані сценарії атак.

Порівняльний аналіз розглянутих підходів до кореляції подій безпеки

Щоб описати переваги і недоліки розглянутих підходів для застосування в системах управління інформаційною безпекою, були виділені наступні критерії оцінки:

- можливості, які полягають в здатності алгоритмів агрегувати схожі події, виявляти послідовність подій від різних сенсорів безпеки і мережевих пристроїв, що утворюють єдиний сценарій атаки;

- необхідність застосування бази знань, визначальною коректність функціонування системи виявлення атак;
- точність підходу, що полягає в здатності алгоритмів виявляти атаки і прогнозувати їх розвиток;
- гнучкість і розширюваність, яка оцінює рівень адаптованості модуля кореляції подій алгоритмів до появи нових видів атак і можливість налаштування параметрів кореляції користувачем;
- обчислювальна ефективність алгоритмів, що визначає потужність обчислювальних ресурсів, необхідних для виконання кореляції подій безпеки з метою виявлення атак.

Алгоритми на основі подібності подій безпеки в меншій мірі вимагають контекстної інформації про предметну область, про можливі сценарії атак зокрема, оскільки вони виконують кореляцію даних на основі аналізу подібності атрибутів події безпеки, що робить їх більш універсальними порівняно з методиками кореляції на основі знань, які припускають наявність даних про конфігурацію пристроїв мережі, її топології, встановлення залежностей між використовуваними мережевими сервісами. Очевидно, результативність алгоритмів як на основі знань, так і на основі правил сильно залежить від коректності опису використовуваних правил, семантичного навантаження подій безпеки, тому *їх розробка потребує безпосередньої участі експертів в ІБ.*

Виняток становлять методики кореляції на основі правил, в яких взаємозв'язку між атрибутами подій безпеки встановлюються за допомогою методик машинного навчання.

Однак в цьому випадку результати кореляції є «прозорими» для кінцевого користувача, і, як показали дослідження різних SIEM-систем, механізми валідації коректності функціонування моделей кореляції даних відсутні. Визначення існуючих сценаріїв атак, як і установка передумов і наслідків, є нетривіальним завданням, якість вирішення якої визначається в першу чергу повнотою вихідних даних про предметну область. Досліди показали, що складність графа атак, побудованого для комп'ютерної мережі, що складається з

n вузлів, тільки на основі даних про топології, конфігурації її вузлів без урахування існуючих залежностей між мережевими сервісами становить O (scn2), де c - це середнє число вразливостей для одного хоста, s - середнє число умов на хості, що забезпечують реалізацію атаки. Це визначає досить високі вимоги до обчислювальних потужностей пристроїв при виконанні кореляції подій в режимі реального часу. Разом з тим саме алгоритми на основі сценаріїв атак або графів атак здатні виявляти складні багатокрокові атаки, об'єднуючи безліч подій безпеки, зареєстрованих на різних вузлах комп'ютерної мережі в різні періоди часу, в єдину послідовність, що описує дії зловмисника.

Високою точністю - низьким рівнем помилок, виявленням хибнопозитивних спрацьовувань сенсорів безпеки - володіють алгоритми, що вимагають вихідних даних у вигляді експертних знань. В першу чергу до них відносяться алгоритми на основі сценаріїв атак і передумов-наслідків, в другу - алгоритми на основі подібності. Істотним недоліком цих двох підходів є їх нездатність виявляти нові типи атак, що використовують не відомі на момент розробки моделі кореляції даних вразливості програмного забезпечення, помилки налаштувань мережевих пристроїв.

Таким чином, для адаптації модуля кореляції до появи нових, вже виявлених атак необхідно здійснювати регулярне оновлення баз використовуваних правил, а в разі використання алгоритмів на основі сценаріїв атак і графів атак - оновлювати структуру графа атак при будь-якій зміні конфігурації комп'ютерної мережі, налаштувань мережевих пристроїв і виявлених вразливостей програмного забезпечення.

Здатністю виявляти нові сценарії атак мають імовірнісні методи кореляції, оскільки в їх основі лежить побудова статистичної моделі функціонування досліджуваної мережі, і будь-яке відхилення від неї може бути розцінено як потенційний вплив зловмисника. Крім того, вони мають досить високу ефективність. А ось загальна точність цієї групи алгоритмів кореляції невисока в зв'язку з мінливістю моделі «нормального» функціонування комп'ютерної мережі.

Узагальнена порівняльна характеристика підходів дана в табл. 2.1.

Таблиця 2.1.

Порівняльний аналіз підходів до кореляції подій ІБ

Характеристика	**Алгоритми кореляції подій ІБ**		
	на основі подібності	**на основі знань**	**імовірнісні**
Комбінація подій безпеки від різних датчиків безпеки	Так	Так	Так
Вимога попередніх знань (Навчання алгоритму)	Так	Так	Ні
Точність (виявлення помилкових подій безпеки)	Так	Так	Малоймовірно
Виявлення багатокрокових атак	Малоймовірно	Так	Малоймовірно
Виявлення нових атак	Ні	Ні	Так
Рівень помилок	Середній	Низький	Високий
Обчислювальна ефективність	Висока	Низька	Середня
Гнучкість і розширюваність	Висока	Висока	Низька

2.2. Порівняльний аналіз архітектур модуля кореляції подій ІБ

Слід зазначити, що на практиці в основному застосовуються методи кореляції подій на основі правил подібності і кодової книги, що пояснюється в першу чергу їх високою обчислювальної ефективністю і точністю, проблема виявлення нових сценаріїв атак вирішується шляхом регулярного оновлення баз знань. Автори вважають досить перспективними також імовірнісні методи кореляції, які потребують низьку обчислювальну ефективність, так і мають здатність до виявлення нових типів атак, що в поєднанні з методиками на основі подібності дозволить не допустити виникнення ситуацій пропуску атак в силу їх новизни.

У більшості випадків для побудови компонента кореляції подій використовується архітектура, керована подіями (event-driven architecture, EDA), яка є шаблоном архітектури програмного забезпечення. У науковій літературі представлено 2 основні підходи до побудови компонентів кореляції подій - централізована архітектура компонента[30,31] і розподілена архітектура[29],[32-34].

Централізована архітектура компонента кореляції подій. Ключовим елементом даного типу архітектури є центральний вузол кореляції подій, який аналізує події від різних датчиків безпеки[30,31]. На рис. 2.3. представлена централізована архітектура компонента безпеки.

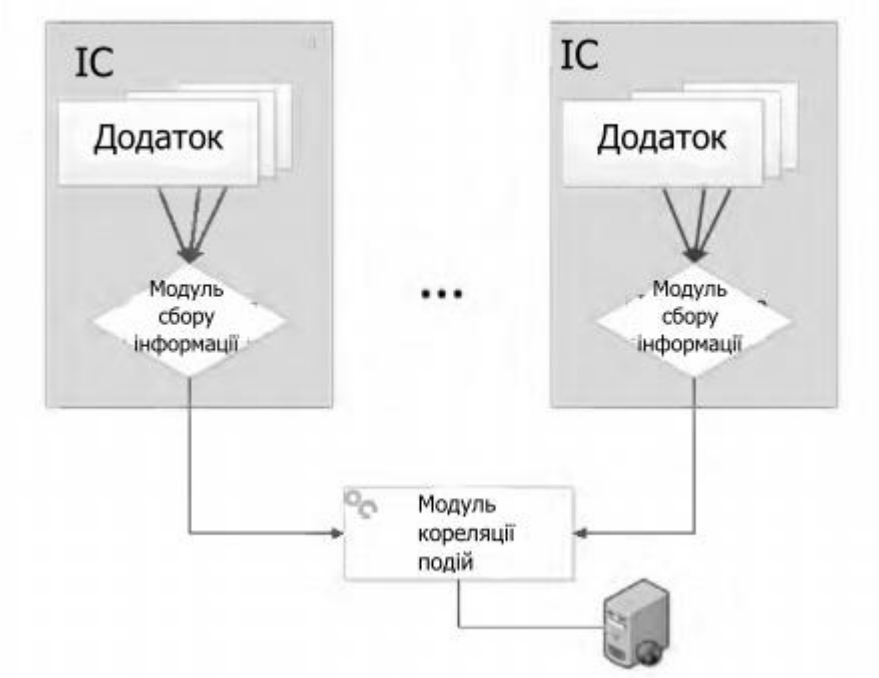

Рис. 2.3. Централізована архітектура компонента кореляції подій

Розподілена архітектура компонента кореляції подій. Певні недоліки централізованої архітектури компонента кореляції подій (навантаження на мережу, обчислювальні обмеження центрального компонента кореляції подій) можуть бути усунені при використанні розподіленої архітектури кореляції подій[32,33]. Окрім того, така архітектура здатна підвищити загальну ефективність функціонування системи кореляції подій[33]. В загальному випадку розподілена архітектура модуля кореляції подій має 2 типа:

- повністю розподілена архітектура модуля кореляції подій;

- ієрархічна архітектура модуля кореляції подій.

У першому випадку всі компоненти кореляції не залежать одне від одного, ієрархічні зв'язки між ними відсутні. Головний модуль кореляції може бути обраний випадковим чином, так як в деяких випадках потрібна координація взаємодії, наприклад для обміну інформацією про нові типи атак між модулями кореляції. Однак, якщо з якоїсь причини головний модуль виходить з ладу, його легко може замінити будь-який інший. Схема такої архітектури представлена на рис. 2.4.

Рис. 2.4. Повністю розподілена архітектура модуля кореляції подій

Ієрархічна архітектура компонента кореляції подій представлена на рис. 2.5.

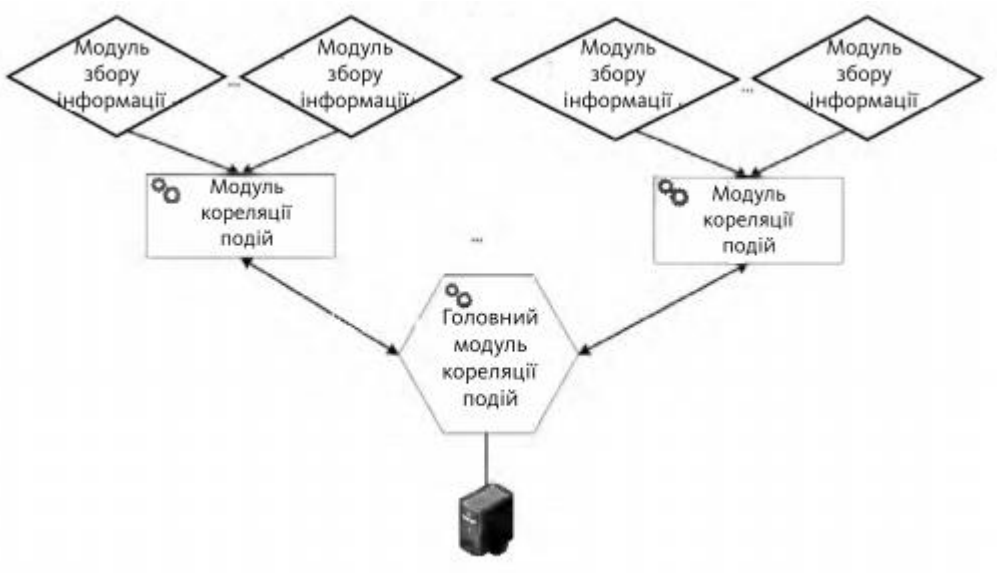

Рис. 2.5. Ієрархічна архітектура компонента кореляції подій представлена

Певні складнощі з використанням даного підходу до побудови модуля кореляції пов'язані з налаштуванням алгоритму кореляції, оскільки він виконується в 2 етапи - локально і централізовано (або глобально). Локальні модулі кореляції виконують кореляцію локальних подій безпеки від обмеженого числа датчиків безпеки, результати локальної кореляції подій надсилаються до модулів, що належить більш високих рівнів кореляції. Ці модулі здійснюють додатковий аналіз даних, виконують розрахунок статистик для користувача. Центральний модуль кореляції (кореневої вузол в ієрархії) будує граф кореляції на основі сформованих раніше результатів, він надає адміністратору безпеки звіт про виявлену атаку і її характеристики.

Порівняльний аналіз архітектур модуля кореляції подій представлений в табл. 2.2.

Таблиця 2.2.

Порівняльний аналіз архітектур модуля кореляції подій

Тип архітектури	**Основний елемент**	**Переваги**	**Недоліки**
Централізована	Центральний вузол кореляції подій	Відносна простота алгоритму кореляції подій	Всі записи передаються на центральний модуль кореляції, в результаті утворюється додаткове навантаження на пропускну

			здатність мережі. Обчислювальні здібності компонента кореляції обмежені, тому при великому потоці подій безпеки виникають тимчасові затримки в обробці подій. Зазвичай не існує яких-небудь механізмів захисту, встановлених між центральним модулем кореляції подій і датчиками безпеки. В разі недоступності модуля кореляції подій вся система кореляції подій виходить з ладу.
Розподілена	Може бути обраний випадковим чином	Взаємозамінність модулів кореляції подій	Необхідність розробки додаткового модуля кореляції, який би займався розстановкою пріоритетів проаналізованих подій і висновком цієї інформації користувачу
Ієрархічна	Головний модуль кореляції	Підвищення обчислювальної ефективності модуля кореляції	Складнощі при налаштуванні двоетапного алгоритму кореляції

2.3. Завдання модуля кореляції подій

До основних завдань модуля кореляції подій безпеки слід віднести:

- зниження рівня надлишкових подій безпеки;
- фільтрацію подій безпеки з низьким рівнем достовірності;
- визначення сценаріїв атак;
- фільтрацію і розстановку пріоритетів подій безпеки;
- передбачення наступного дії зловмисника;
- передбачення інших можливих атак.

Під подією в інформаційних системах розуміється створення деякої реєстраційного запису про відбувається в системі активності в відповідь на зміну стану ІС, її ресурсів, процесів і т. д. Зазвичай такий запис відображає 3 основні характеристики активності - тип (або форма) події, значимість події і зв'язок з іншими подіями[35].

Тип події позначає атрибути або компоненти інформаційної системи, залучені в певний процес. Зв'язок з іншими подіями (Процесами) в системі виражається в формі встановлення взаємовідносин між іншими подіями (або процесами) - вона може бути причинно-наслідкового (каузального) або об'єднує (якщо подія є абстракцією безлічі низькорівневих подій). Прикладом поняття «подія» в області ІБ можуть служити події безпеки / повідомлення від міжмережевих екранів, систем виявлення вторгнень, подій гіпервізора віртуальних машин і т. д.

Процес кореляції подій в загальному випадку незалежно від їх природи пов'язаний з двома основними поняттями[36,37]:

- обробкою подій;
- кореляцією подій.

Під обробкою подій розуміються обчислення і операції, визначені на множині подій, метою яких найчастіше є їх читання, створення, перетворення і видалення.

Кореляція подій - це спосіб вилучення високорівневих знань з інформації, представленої у вигляді безлічі подій. прикладом знань, які можуть бути отримані в результаті кореляції подій, є визначення аномальної ситуації, передбачення можливих майбутніх ситуацій або виявлення першопричини аномалії. Очевидно, що ці знання безпосередньо можуть бути використані при вирішенні завдання забезпечення ІБ. Для отримання таких знань кореляція подій виконується в 4 основних етапи[37](табл. 2.3.)

Таблиця 2.3.

Етапи процесу аналізу подій

Етап	Функція
Фільтрація подій	Видалення подій, які не є релевантними поставленій задачі і спостережуваним процесам в системі.
Агрегація подій	Злиття дублюючих даних для однієї події
Аналіз основних причин (Root cause analysis)	Аналіз залежностей між подіями, в більшості випадків на основі побудованої моделі навколишнього середовища і графа залежностей, для того щоб зрозуміти, чи можуть одні події пояснити інші. Це ключовий етап усього процесу кореляції подій, і він може бути розбитий на підетапи, характер яких залежить від безпосередньо поставленої задачі. Наприклад, при виявленні мережевих вторгнень даний етап може складатися з підетапів - відновлення ходу атаки, відновлення сесії атаки, визначення та хід атаки і т. д.
Маскування подій	Ігнорування подій, що надходять від систем, залежних від відмов роботи системи

Основні операції над подіями. Безпосередньо процес виявлення патернів в події (або аналіз складних подій) може бути описаний за допомогою безлічі операцій. Вибір операцій залежить від поставленого завдання і, відповідно, етапу процесу кореляції. За своїм змістом операції можуть бути розділені на 3 основні групи:

- операції фільтрації, які здійснюють відбір подій, що беруть участь в процесі кореляції;
- операції зіставлення, які виконують пошук патернів серед вхідних подій і створюють нові події, які задовольняють деякому шаблоном;
- операції виведення, які використовують вихідні дані операції зіставлення для створення нових подій або налаштування параметрів подій.

До групи фільтрації можна віднести наступні операції:

- Стиснення (або видалення дублікатів, агрегація подій, compression).
- Операція заміни безлічі ідентичних подій, які відрізняються тільки часом генерації, на одну-єдину подію. Результуюча подія містить в ідеальному випадку число вихідних подій і час першої і останньої події.

- Агрегація подій (aggregation).Під агрегацією подій розуміється об'єднання безлічі подій в одну нову, при цьому не вимагається виконання умови ідентичності вихідних подій, як у випадку операції стиснення.
- Фільтрація подій (filtering). Зазвичай виділяють 2 режима фільтрації подій - без запам'ятовування стану (stateless filtering) мова йде про видалення подій, що володіють певними властивостями, з вхідного потоку. При цьому не береться до уваги внутрішній стан модуля кореляції, враховуються тільки атрибути безпосередньо самої події.

  - із запам'ятовуванням стану (stateful filtering) операція фільтрації здійснює придушення певних подій в залежності від контексту (стану) модуля кореляції подій. В цьому випадку оператор фільтрації часто називають оператором придушення подій. Окремим випадком операції придушення є операція маскування подій, яка також називається топологічним маскуванням подій. Метою цієї операції є приховування подій від вузлів, які вже повідомили про можливі проблеми.
- Завдання порога спрацьовування (thresholding). Операція, в результаті якої створюється подія, якщо частота певної події перевищує задане граничне значення або, навпаки, падає нижче його.
- Композиція (composition). Операція обробляє групу подій, що надходять від різних джерел даних, вибираючи підмножина за допомогою деякої функції зіставлення, і створює нові події на основі відібраних подій.

Групу співвідношення складають наступні операції:

- Логічні операції. Під логічними операціями розуміється об'єднання подій за допомогою операторів булевої логіки.
- Встановлення тимчасових зв'язків (temporal relationship). Дана операція встановлює взаємозв'язки в часі між подіями або порядком їх появи. Наприклад, інтервал часу між подіями A і B не перевищує 10 хв.
- Перетворення (modification або enrich). Операція перетворює атрибути вихідного події згідно деякої заданої функції, наприклад за результатами його порівняння з іншим вже існуючим подією.

- Виявлення шаблону (або кластеризація (Clustering)). Під кластеризацией розуміється операція, в результаті якої створюється нова подія на основі деякого складноструктурованого паттерна подій, побудованого з використанням раніше перерахованих операцій. Наприклад, створити подія С, якщо частота подій А перевищує поріг 5 подій в хвилину, жодної події В не було зареєстровано в останню годину.

  До операцій виведення можна віднести наступні:

- Зміна атрибута події (escalation, modification). В результаті цієї операції змінюється значення певного параметра події (частіше всього - його пріоритет).
- Узагальнення (Generalization). Під узагальненням розуміється співвіднесення події в деякий клас більш загальних подій.Незважаючи на те, що нової інформації в даному випадку не створюється, процес виявлення «причини-наслідку» зазнає суттєвого спрощення формуванням деяких узагальнюючих конструкцій.
- Уточнення (Specialization). Операція за своїм змістом є протилежною операції узагальнення. В результаті її виконання деякий більш загальна подія замінюється на подію, що відноситься до підкласу класу вихідної події. Наприклад, якщо приходить повідомлення про те, що деякий заданий хост недоступний, операція «уточнення» може автоматично створити події, повідомляють про те, що служби, запущені на заданому хості, недоступні. Можливо, нові повідомлення не містять додаткової інформації про причини недоступності хоста, однак вони можуть послужити тригером для спрацьовування деяких правил, що обробляють недоступні сервіси.

Для забезпечення масштабованості і еластичності безпосередньо модуля кореляції визначені дві операції - семантичний роутер (SemanticRouter) і злиття подій (EventMerger). Операція семантичний роутер забезпечує паралельність виконання процесу кореляції, зокрема він виконує маршрутизацію, т. е. визначає модуль-одержувач вхідних подій, отриманих від різних потоків-обробників черг

подій, які виконуються паралельно. Операція злиття подій функціонує навпаки: все вхідні потоки подій від різних потоків об'єднуються в один потік даних, упорядкований в часі. на рис. 2.6. представлений принцип спільного функціонування цих двох операцій.

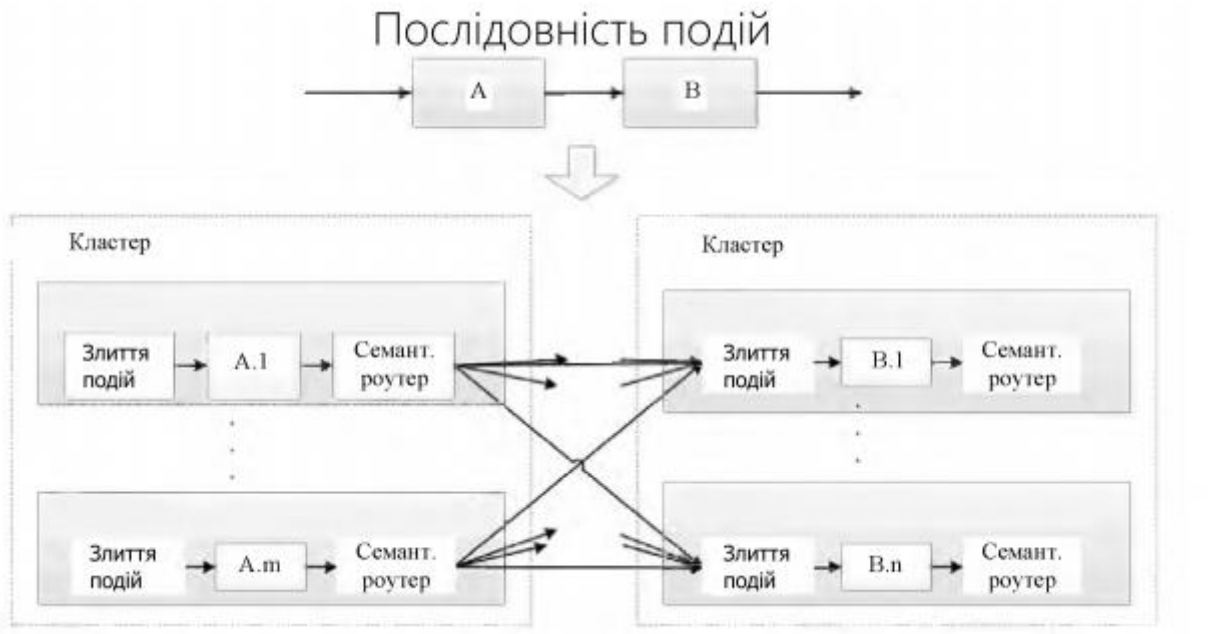

Рис. 2.6. Операція злиття подій

2.4. Висновки до розділу

Очевидно, результативність алгоритмів як на основі знань, так і на основі правил сильно залежить від коректності опису використовуваних правил, семантичного навантаження подій безпеки, тому *їх розробка потребує безпосередньої участі експертів в ІБ.*

Визначення існуючих сценаріїв атак, як і установка передумов і наслідків, є нетривіальним завданням, якість вирішення якої визначається в першу чергу повнотою вихідних даних про предметну область. Визначає досить високі вимоги до обчислювальних потужностей пристроїв при виконанні кореляції подій в режимі реального часу. Разом з тим саме алгоритми на основі сценаріїв атак або графів атак здатні виявляти складні багатокрокові атаки, об'єднуючи безліч подій безпеки, зареєстрованих на різних вузлах комп'ютерної мережі в різні періоди часу, в єдину послідовність, що описує дії зловмисника.

Таким чином, для адаптації модуля кореляції до появи нових, вже виявлених атак необхідно здійснювати регулярне оновлення баз використовуваних правил, а в разі використання алгоритмів на основі сценаріїв атак і графів атак - оновлювати структуру графа атак при будь-якій зміні конфігурації комп'ютерної мережі, налаштувань мережевих пристроїв і виявлених вразливостей програмного забезпечення.

РОЗДІЛ 3. РОЗРОБКА ПРОГРАМНОГО МОДУЛЮ АНАЛІЗУ ПОДІЙ ДЛЯ СТЕКУ ELK

3.1. Розгортання стеку ELK

Для реалізації абревіатура сервісів ELK передбачає наступні завдання:

1) Обробку даних, що надходять і доставку їх в Elasticsearch - за це відповідає сервіс Logstash

2) Пошуковий движок і інтерфейс доступу до даних - за це відповідають сам Elasticsearch і Kibana

Logstash не повинен відповідати за доставку даних, оскільки краще доставку даних делегувати четвертому сервісу - Filebeat.

Загальна схема роботи виглядає наступним чином на рис. 3.1.:

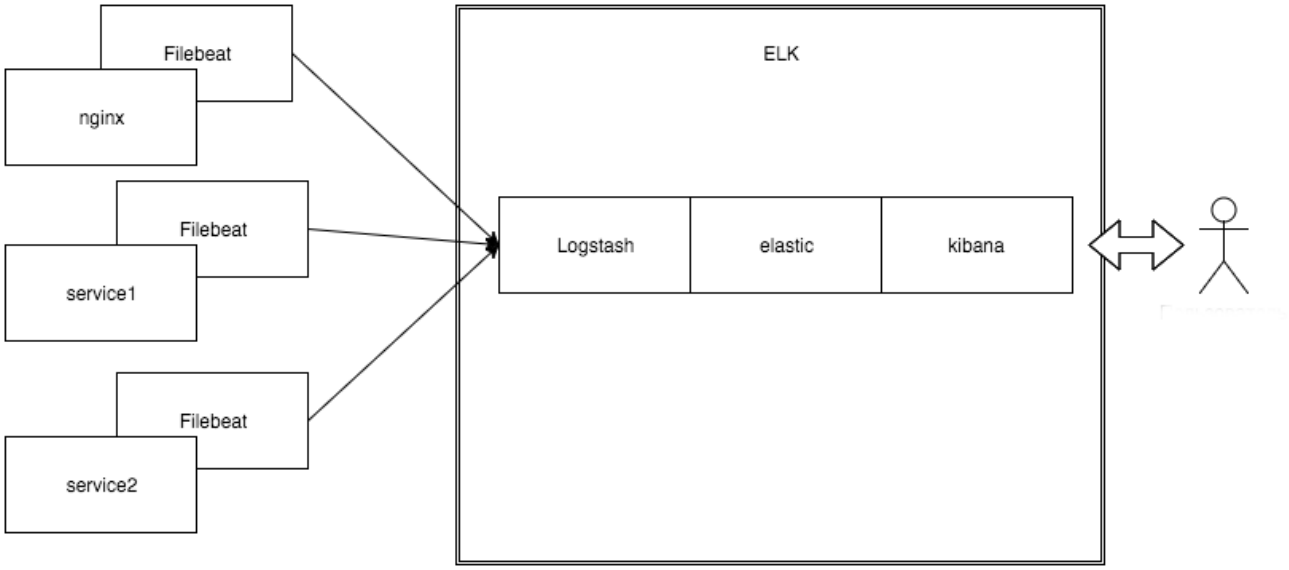

Рис. 3.1. Схема роботи стеку ELK

Стек ELK (Elasticsearch, Logstash та Kibana) може бути встановлений на безліч різних операційних систем. Хоча найпоширенішою установкою встановлення є Linux та інші системи, засновані на Unix, для проведення розробки ПЗ зручним є використання Docker.

Я використовую Dockerized ELK Stack, в результаті чого встановлюються три контейнери Docker, що працюють паралельно: Elasticsearch, Logstash та Kibana, налаштована переадресація портів та обсяг даних для збереження даних Elasticsearch.

Спочатку потрібно завантажити репозиторій:

git clone https://github.com/deviantony/docker-elk.git

Для запуску ELK стеку знадобиться лише одна команда:

docker-compose up

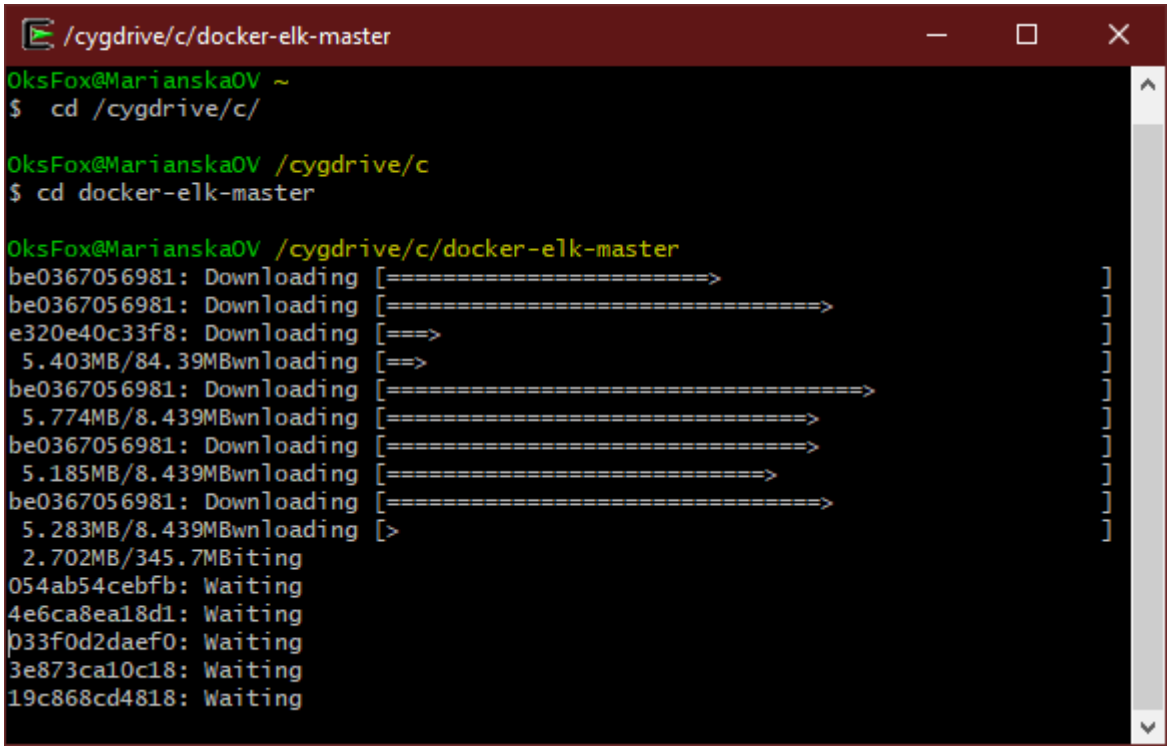

Рис. 3.2. Процес завантаження контейнерів Docker

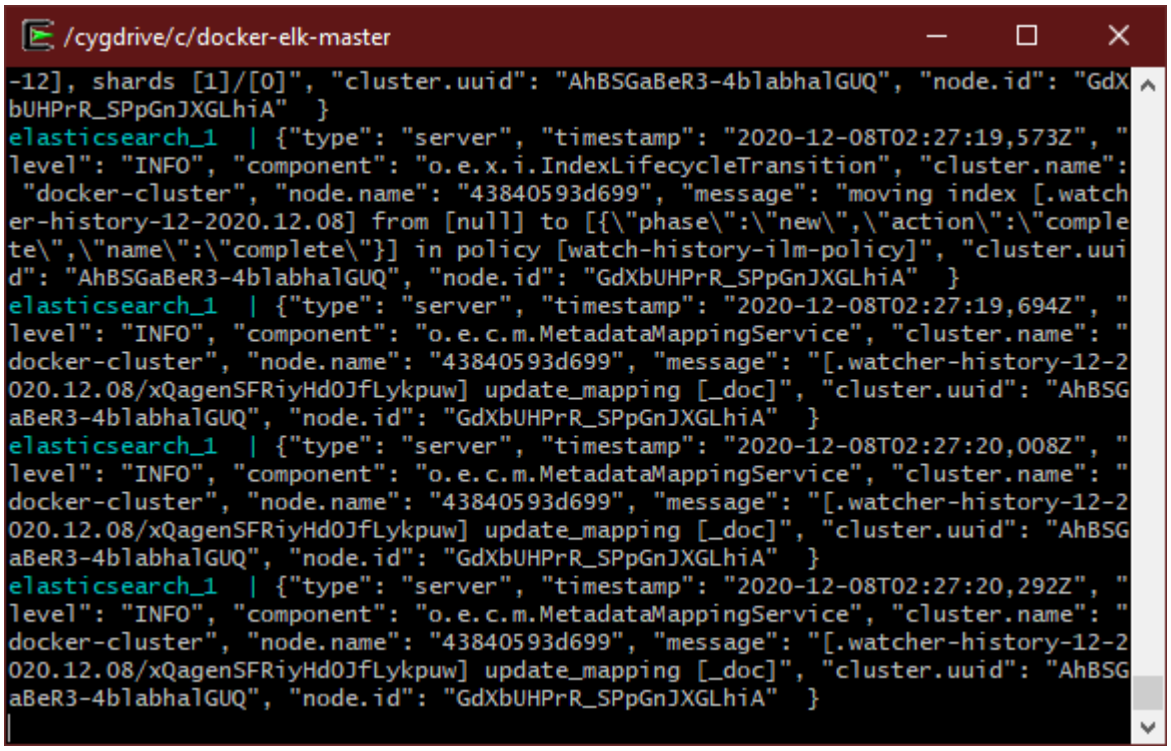

Рис. 3.3. Відображення процесу роботи контейнерів Docker

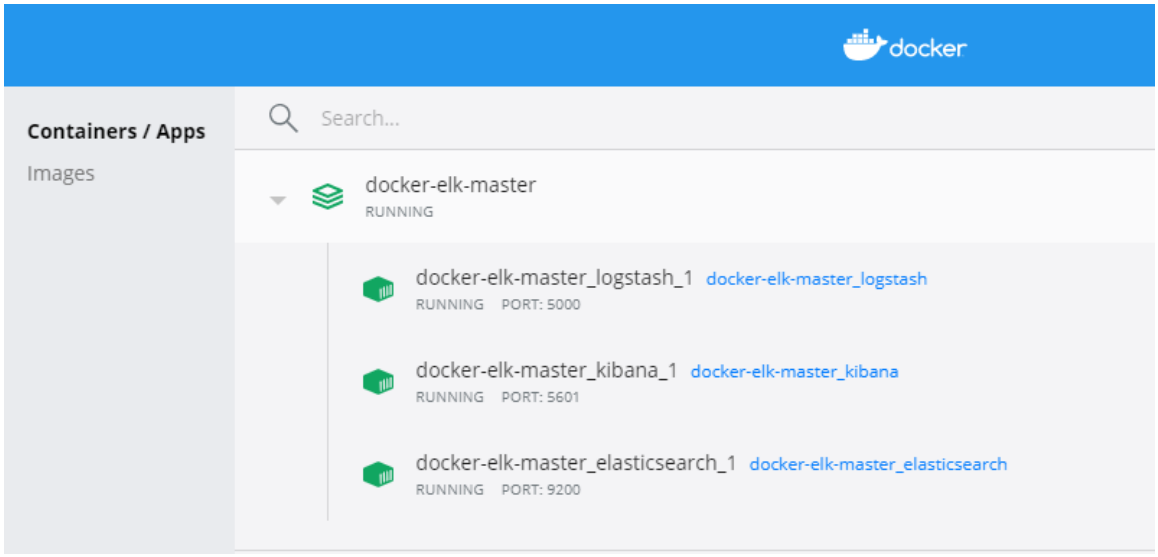

Рис. 3.4. Запущений ELK стек за допомогою Docker

Тепер продемонструю вигляд інтерфейсу ELK без розробленого модулю аналізу подій:

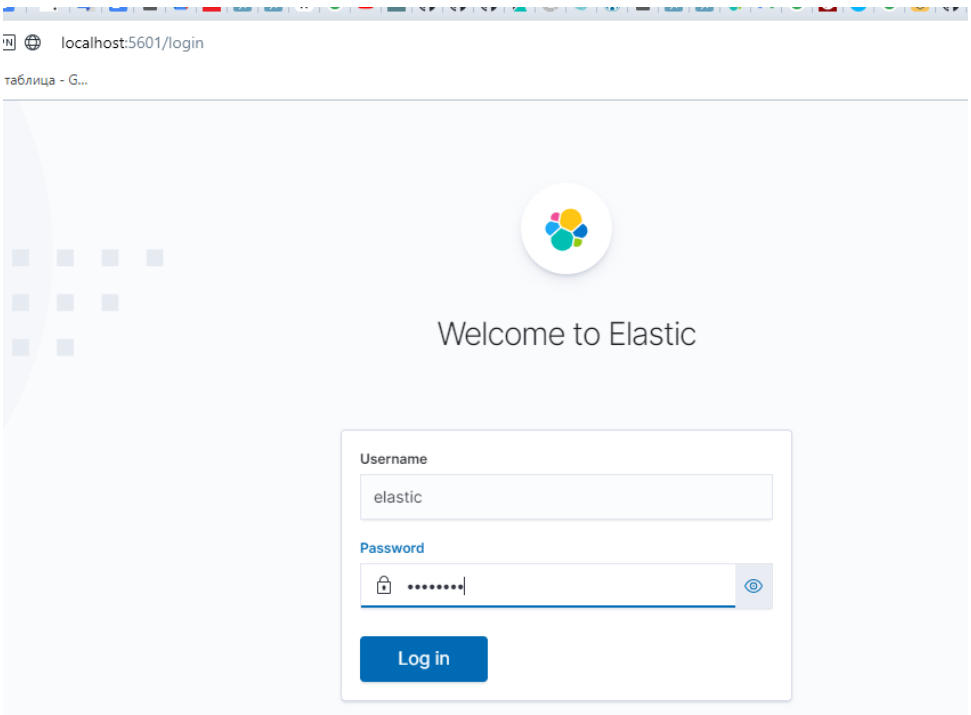

Рис.3.5. Авторизація до систему стеку ELK

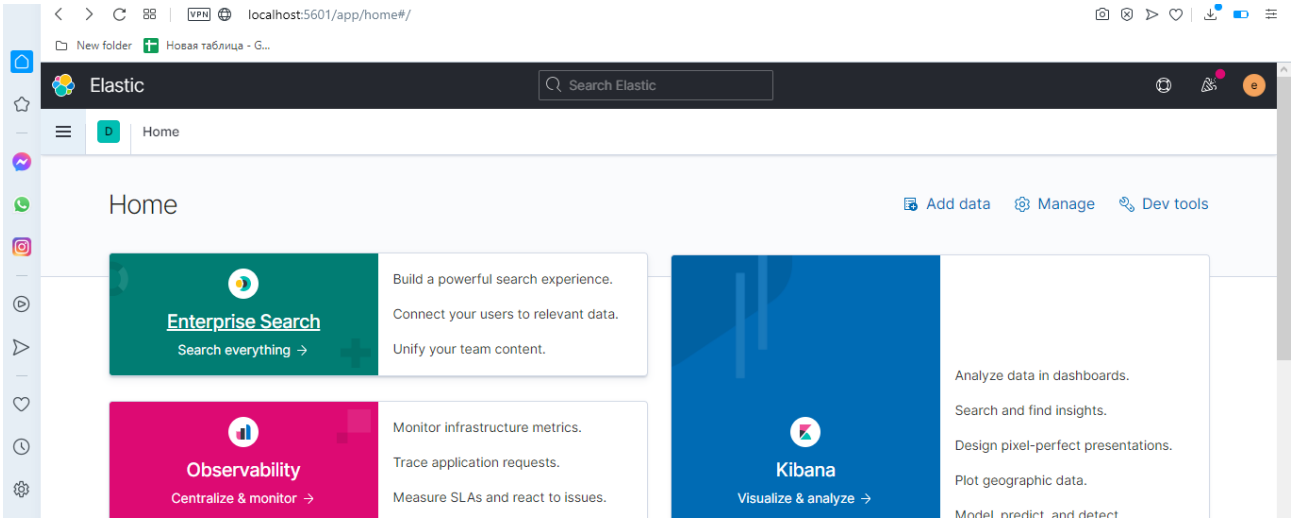

Рис.3.6. Веб інтерфейс стеку ELK

3.2. Схема роботи модулю аналізу подій ІБ на основі стеку ELK

Схема роботи модулю аналізу подій ІБ в загальному випадку представлена на рис. 3.7.

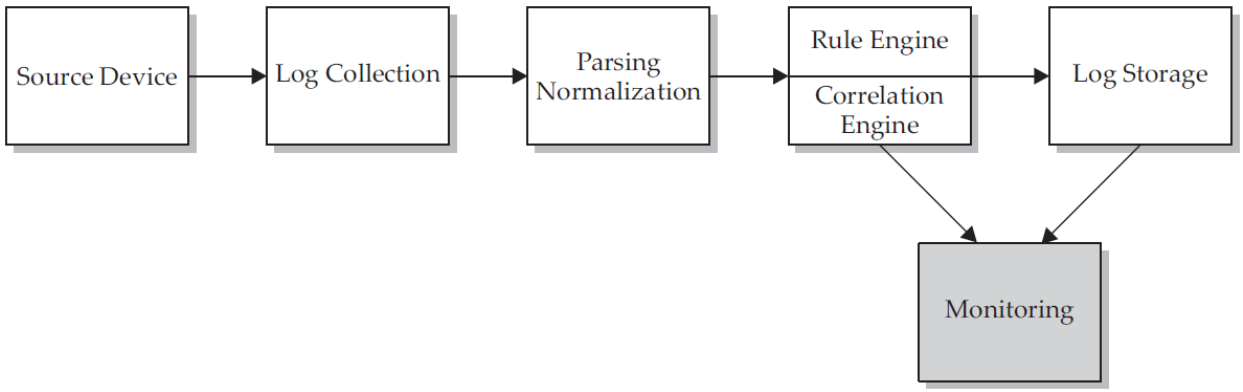

Рис. 3.7. Схема роботи модулю аналізу подій ІБ в загальному випадку

Побудована схема роботи модулю аналізу подій ІБ (Module of event analysis) на основі стеку ELK представлена на рис. 3.3.

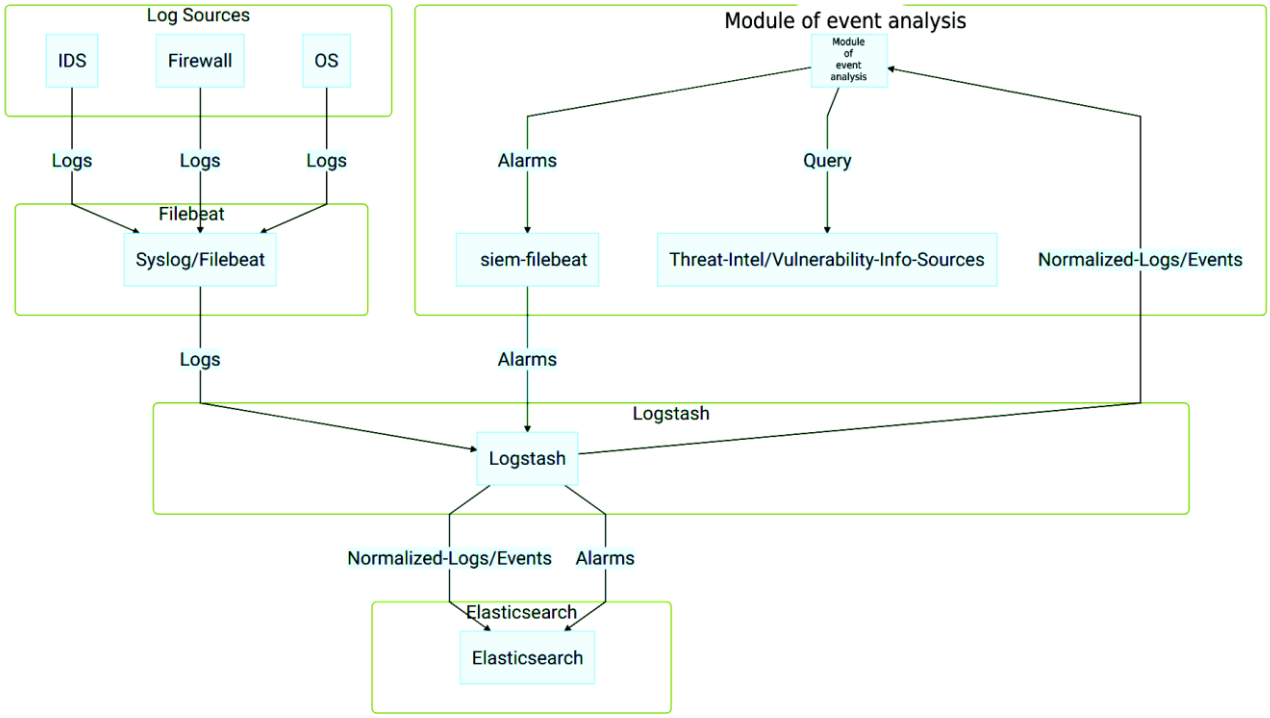

Рис. 3.8. Схема роботи модулю аналізу подій з ELK стеком

На діаграмі вище:

1. Джерела журналів надсилають свої журнали до Syslog / Filebeat, який потім надсилає їх до Logstash з унікальним ідентифікаційним полем. Потім Logstash аналізує журнали за допомогою різних фільтрів на основі типу джерел журналів і надсилає результати до Elasticsearch, зазвичай створюючи єдиний шаблон індексу для кожного типу журналу (наприклад, ids- * для журналів, отриманих від IDS, ssh- * для журналів SSH тощо).
2. Модуль аналізу подій ІБ використовує спеціальний конфігураційний файл logstash для клонування вхідної події з джерел журналу, відразу після того, як logstash зробить її аналіз. Через той самий конфігураційний файл нова клонована подія використовується (незалежно від початкової події) для збору необхідних для модуля аналізу подій ІБ полів, таких як Title, Source IP, Destination IP тощо.
3. Вихідні дані цього кроку називаються нормалізованою подією, оскільки вони представляють журнали з декількох різних джерел в одному форматі, який має набір загальних полів. Потім ці події надсилаються до модулю

аналізу подій ІБ через вихідний плагін Logstash HTTP та до Elasticsearch під шаблоном імені індексу siem_events- *.

4. Модуль аналізу подій ІБ корелює вхідні нормалізовані події на основі налаштованих правил кореляції подій, виконує пошук інформації про загрози та вразливості, а потім генерує сигнал загрози в Alarm, відповідно до умов правил. Потім сигнал загрози записується в локальний файл журналу, який збирається локальним Filebeat, налаштованим на відправлення його вмісту в Logstash.
5. У кінці logstash є ще один спеціальний конфігураційний файл модуль аналізу подій ІБ, який зчитує подані сигнали загроз ІБ та надсилає їх до остаточного індексу загроз ІБ в Elasticsearch.

3.3. Побудова виявлення сценарію експлуатації вразливості CVE-2014-6271 («Bash»)

У 2020 році і досі існують компанії, які не поспішають оновлювати ПЗ інфраструктури та застосування застарілого програмного забезпечення, що не має останніх патчів і оновлень, на жаль, є частим випадком. Винятком не стали операційні системи Linux, такі як: Debian, CentOS.

Вразливість CVE-2014-6271 - це легко експлуатована вразливість, використання якої може призвести до дуже серйозних наслідків. Експлуатуючи цю вразливість, атакуюча сторона отримує можливість виконувати команди системного рівня з такими ж привілеями, як і атакуємі сервіси. Це стосується не тільки веб-серверів, але і будь-якого ПО, яке використовує інтерпретатор bash і зчитує дані, які ви можете контролювати.

Варто зазначити, що інформація про наявність критичних вразливостей CVE-2014-6271 одержало назву «Shellshock» в Bash та відома ще з 2014 року. Вразливість Shellshock (bashdoor) відноситься до програми bash (розробляється в рамках проекту GNU), який використовується в безлічі Unix-подібних операційних систем і дистрибутивів як інтерпретатора командного рядка і для

виконання командних скриптів. Часто встановлюється в якості системного інтерпретатора за замовчуванням.

В Unix-подібних і інших підтримуваних bash операційних системах кожна програма має список пар ім'я-значення, званий змінними середовища (англ. Environment variable). Коли одна програма запускає іншу, то також передається початковий список змінних середовища. Крім змінних середовища, bash також підтримує внутрішній список функцій - іменованих скриптів, які можуть викликатися з виконуваного скрипта bash. При запуску нових екземплярів bash з існуючого bash можлива передача (експортування, export) значень існуючих змінних оточення і визначень функцій в породжений процес. Визначення функцій експортуються шляхом кодування їх у вигляді нових змінних оточення спеціального формату, який починається з порожніх дужок «()», за якими слід визначення функції у вигляді рядка. Нові екземпляри bash при своєму запуску сканують всі змінні середовища, детектуючи даний формат і перетворюючи його назад в визначення внутрішньої функції. Дане перетворення проводиться шляхом створення фрагмента bash-коду на базі значення змінної середовища і його виконання, тобто «на льоту» ('on-the-fly'). Схильні вразливості версії bash не роблять перевірок, що виконуваний фрагмент містить лише визначення функції. Таким чином, якщо зловмисник має можливість подати довільну змінну середовища в запуск bash, то з'являється можливість виконання довільних команд.

До складу системи захисту мережі від несанкціонованого доступу , як правило, включено підсистеми, що виявляють відомі вразливості(OSSEC, Suricata). Програми цих підсистем, аналізують, який трафік надходить в мережу.

Suricata(IDS) має розвинені засоби інспектування HTTP-трафіку на основі бібліотеки HTP, створеної Іваном тиками (Ivan Ristic), автором ModSecurity. Підтримується вилучення та перевірка переданих по HTTP файлів, розбір стисненого контенту, можливість ідентифікації по URI, cookie, заголовкам, user-agent, тілу запиту і відповіді. Цю можливість Suricata, до речі, в деяких мережах використовують для протоколювання HTTP-трафіку без детектування. Контент

в потоці можна виділяти за маскою і за допомогою регулярних виразів, ідентифікація файлів можлива по імені, типу або контрольної MD5-сумою.

Досліджуваним матеріалом є «сирий» трафік, що збирається за допомогою ELK Stack. Дані таких полів будемо використовувати:

- протокол -protocol;
- IP-адреса джерела - src_ip;
- IP-адреса одержувача - dst_ip;
- порт джерела - src_port;
- порт одержувача - dst_port.

Попередньо в систему закладають інформацію про активні пристрої досліджуваної мережі. Заздалегідь формується безліч адрес мережевих пристроїв, які є легальними, активними джерелами і одержувачами даних. Наприклад, такими пристроями є dns-, proxy- і web-сервери. Ця інформація дозволяє прискорити роботу системи і скоротити число помилкових результатів.

Приклад реалізації вразливості: [http request> cgi script> bash = remote code exploration]. У прикладі використовується обробка CGI скриптів за допомогою оболонки bash. Потенційні вектори експлуатації: DHCP, SSH, Telnet.

Наприклад, у змінній оточення:

VAR = () {ignored; };

/ Bin / id буде виконаний / bin / id при імпорті оточення в процес bash.

Для перевірки можна запустити такий код:

env x = '() {:;}; echo vulnerable 'bash -c "echo this is a test"

Таким чином, змінну оточення з довільним ім'ям можна використовувати як носія для доставки на атакуєму систему потрібних команд. В даний момент найбільш небезпечним методом експлуатації вразливості вважають HTTP-запити до скриптів CGI, які ми і будемо використовувати.

Для сповіщення про експлуатації вразливості на ресурсах нашої системи потрібно створити директиву кореляції подій ІБ від тих джерел, які нам необхідні.

Директива кореляції - набір логічних правил, які дозволяють здійснювати порівняння параметрів подій ІБ, а також їх кількості і частоти з заданими показниками для виявлення інцидентів інформаційної безпеки.

Підтримка методики шаблонів поведінки необхідна будь-якій системі: пакети директив кореляцій відображають можливий ланцюг подій (аномалій), який відповідає моделі реальної атаки(APT).

Створена директива сценарію експлуатації вразливості “CVE-2014-6271”:

root":{1 item
"directives":[1 item
0:{7 items
"name":"Successful Shellshock vulnerability exploitation"
"kingdom":"System Compromise"
"category":"Web Attack"
"all_rules_always_active":true
"id":10
"priority":4
"rules":[3 items
0:{13 items
"name":"Possible shellshock exploit"
"type":"PluginRule"
"stage":1
"plugin_id":1001
"plugin_sid":[1 item
0:2019232
]
"occurrence":1
"from":"ANY"
"to":"HOME_NET"
"port_from":"ANY"
"port_to":"ANY"

"protocol":"HTTP"
"reliability":1
"timeout":0
}
1:{13 items
"name":"OS file system changed"
"type":"PluginRule"
"stage":2
"plugin_id":50001
"plugin_sid":[4 items
0:550
1:551
2:552
3:554
]
"occurrence":1
"from":"ANY"
"to":":1"
"port_from":"ANY"
"port_to":"ANY"
"protocol":"TCP/IP"
"reliability":6
"timeout":600
}
2:{13 items
"name":"OS file system changed"
"type":"PluginRule"
"stage":3
"plugin_id":50001
"plugin_sid":[4 items

0:550
1:551
2:552
3:554
]
"occurrence":3
"from":"ANY"
"to":":1"
"port_from":"ANY"
"port_to":"ANY"
"protocol":"TCP/IP"
"reliability":10
"timeout":3600
}
]
}
]

Під кожну компанію аналізується та проектується кастомізований набір директив кореляції.

3.4. Експлуатація програмного засобу

Docker являє собою систему управління контейнерами. Вона дозволяє «упакувати» додаток або веб-сайт з усім його оточенням і залежностями в контейнер, яким в подальшому можна легко і просто керувати: переносити на інший сервер, масштабувати, оновлювати.

Образи контейнерів Docker містять в собі всі необхідні додатком бібліотеки, тому конфліктів з іншим ПЗ не буде.

Список контейнерів, що нам необхідні для дослідження роботи програмного модулю аналізу подій:

- elasticsearch: швидкий, горизонтально масштабований і безкоштовний гібрид NoSQL бази даних і пошук для неї. Він використовує HTTP API і через нього вже отримує на вхід JSON документи для індексації і зберігання.
- logstash: дозволяє збирати, аналізувати, фільтрувати, нормалізувати дані, має більше 200 плагінів які дозволяють підключати велике число різних типів джерел або потоків даних.
- kibana: програмна панель візуалізації даних, у процесі використання програми інформація, проіндексована в кластері Elasticsearch, представляється у вигляді діаграм різних видів
- siem: розроблений модуль аналізу подій ІБ
- siem-demo-frontend: розроблений контейнер веб інтерфейсу для модулю аналізу подій;
- filebeat: виконує пересилання даних журналів подій, встановлений у якості агента на ваших серверах, контролює вказані вами файли журналів або розташування, збирає події журналу та передає їх до Elasticsearch або Logstash для індексації.
- auditbeat: передає події в реальному часі стеку ELK (Elasticsearch, Logstash і Kibana) для подальшого аналізу;
- suricata: програмне забезпечення для реалізації IDS (система виявлення вторгнень), IPS (система запобігання вторгнень) і NSM (моніторинг мережевої безпеки). Продукт був розроблений організацією Open Information Security Foundation (OISF) як альтернатива Snort. Для зручності роботи, правила останнього сумісні з suricata;
- ossec: це хостова система виявлення вторгнень (HIDS) з відкритим вихідним кодом. Якщо у вас з'явилася задача перевірки контролю цілісності файлів на ваших серверах, логування різних дій на серверах, отримання подій безпеки з ваших серверів (а також будь-яких інших) і сповіщень про ці події, виведення різних звітів і багато іншого, то HIDS OSSEC - відмінне рішення під ці завдання;

Ось так виглядає модуль аналізу подій до подання журналів подій:

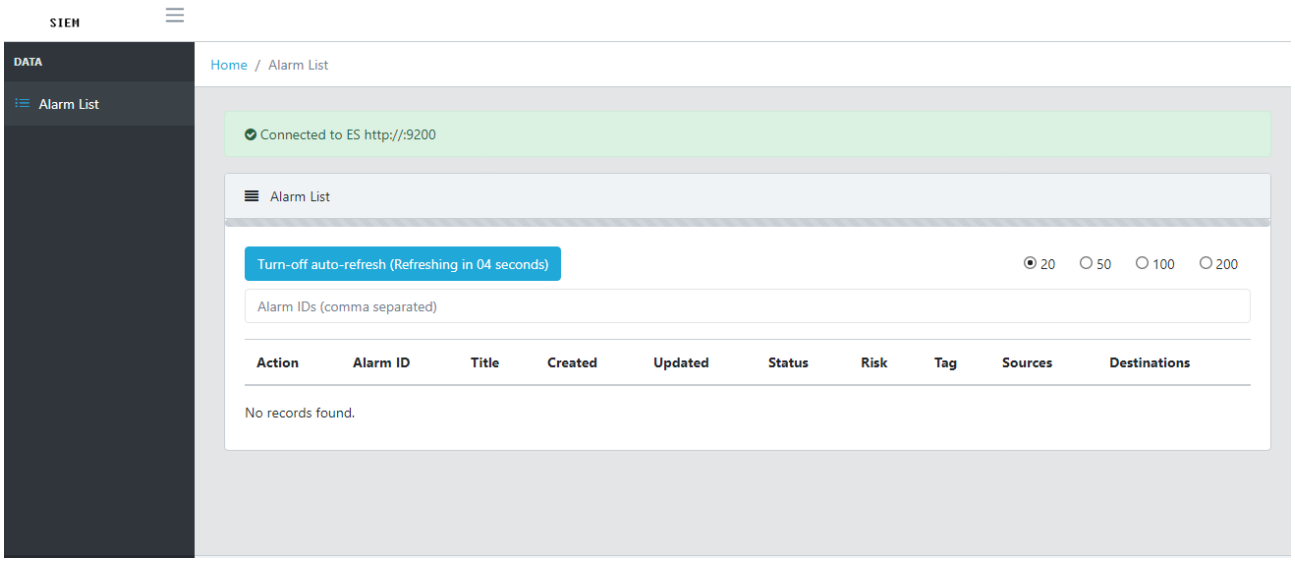

Рис. 3.9. Веб інтерфейс розробленого модулю аналізу подій

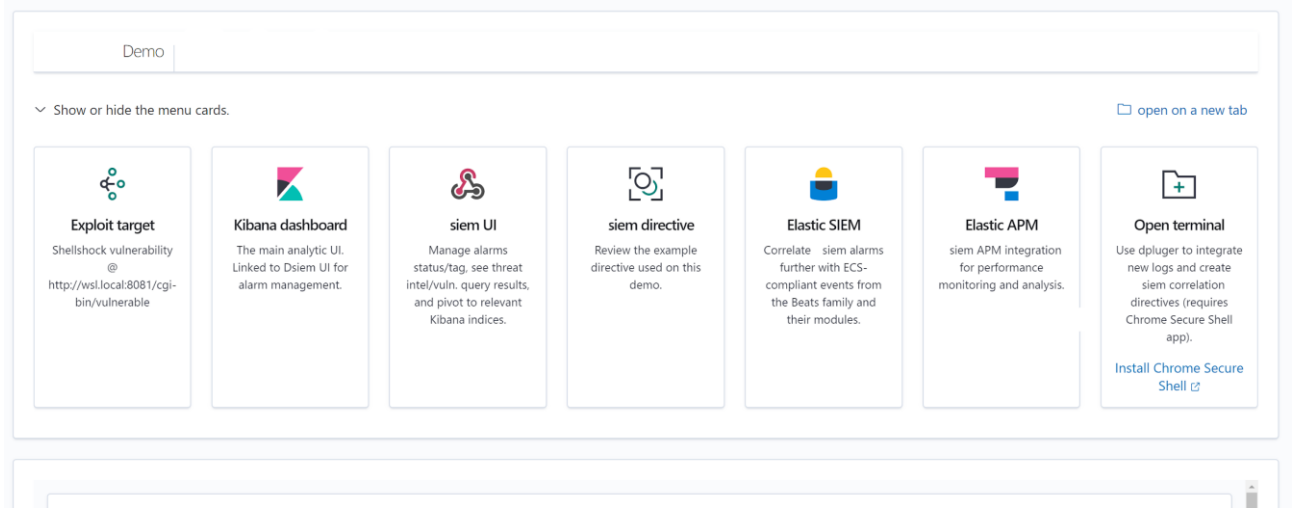

Рис. 3.10. Веб інтерфейс меню системи на основі стеку ELK

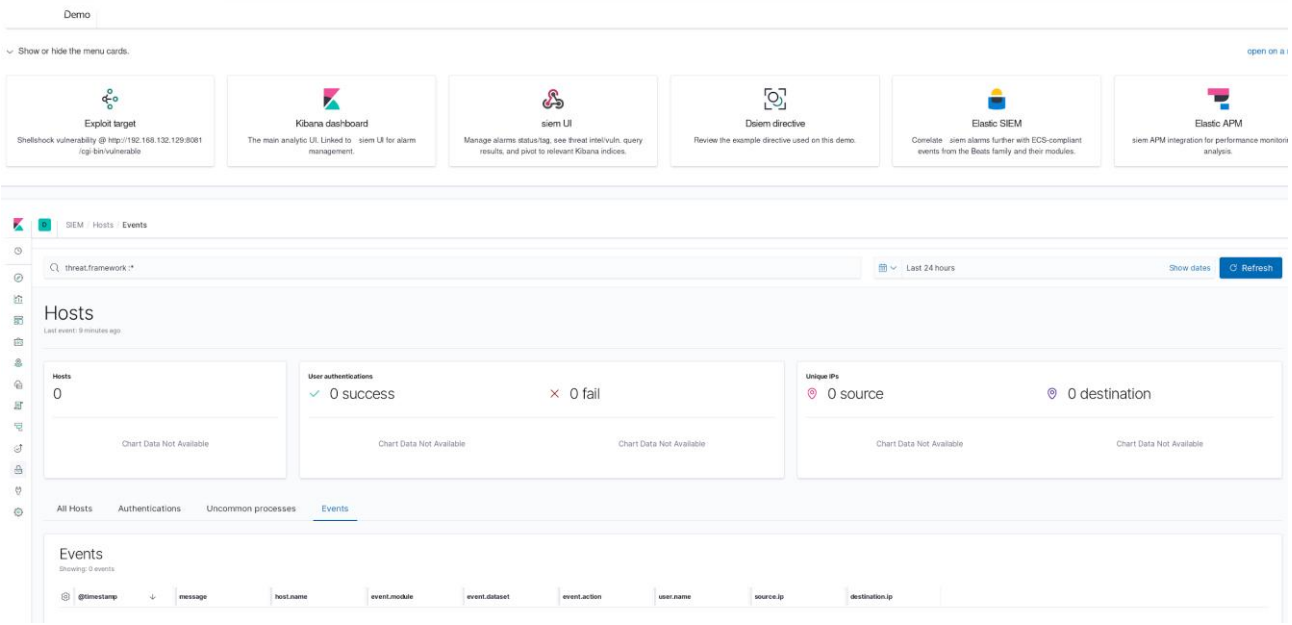

Рис. 3.11. Веб інтерфейс системи відображення хостів на основі стеку ELK

3.5. Експериментальне дослідження системи

Для дослідження роботи розробленого модулю аналізу подій ІБ, інтегрованого в стек ELK, встановлюю віртуальну хостову машину:

- CentOS 7

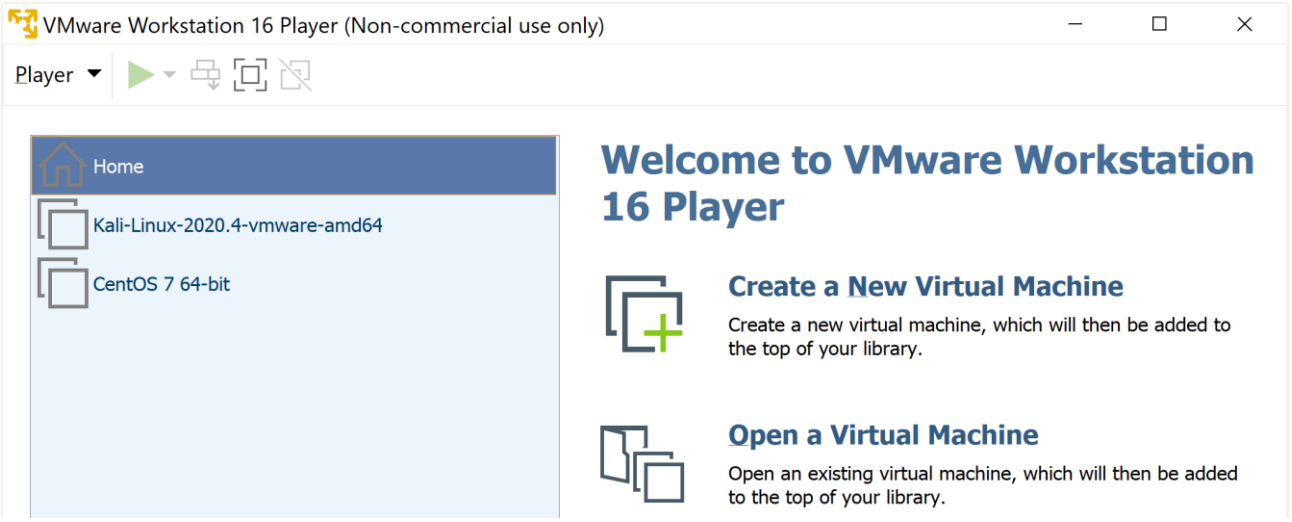

Рис. 3.12 Розгортання віртуальної операційної системи для дослідження

Вимоги до хостової системи:

- Docker та Docker Compose;
- Bash та загальні утиліти Unix (grep, awk тощо) - протестовано лише на Linux та WSL);

- Принаймні 4 ядра процесора та 8 ГБ оперативної пам'яті на хостовій системі;
- Місце на диску до 20 ГБ для зберігання всіх зображень докера та створених журналів / файлів.

Далі розгортаємо середовище з контейнерами, які необхідні для тестування.

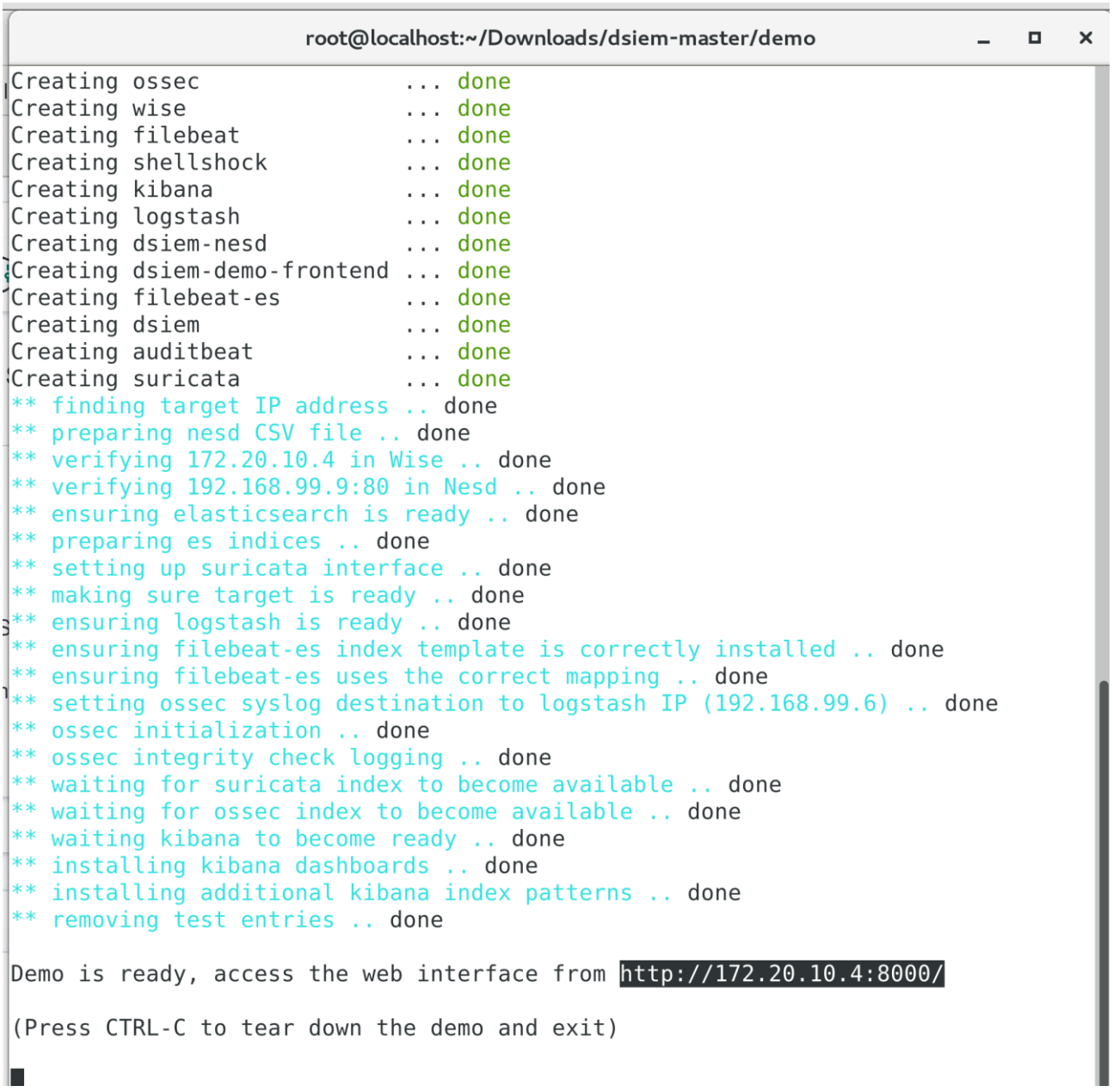

Рис. 3.13. Розгортання контейнерів на Docker

Запускаю систему з інтегрованим розробленим модулем аналізу подій на основі стеку ELK:

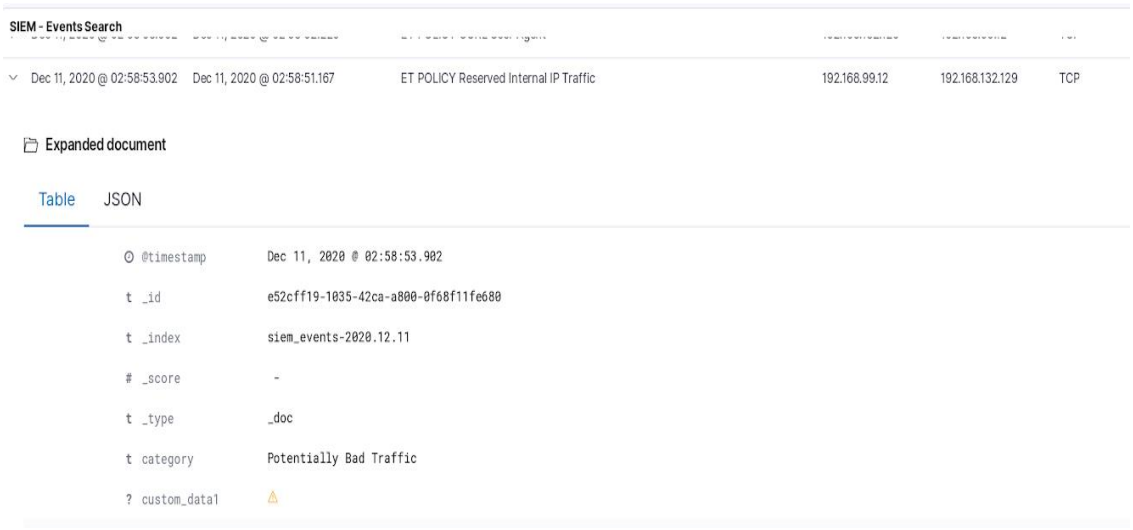

Рис. 3.14. Відображення про подію в розгорнутому вигляді

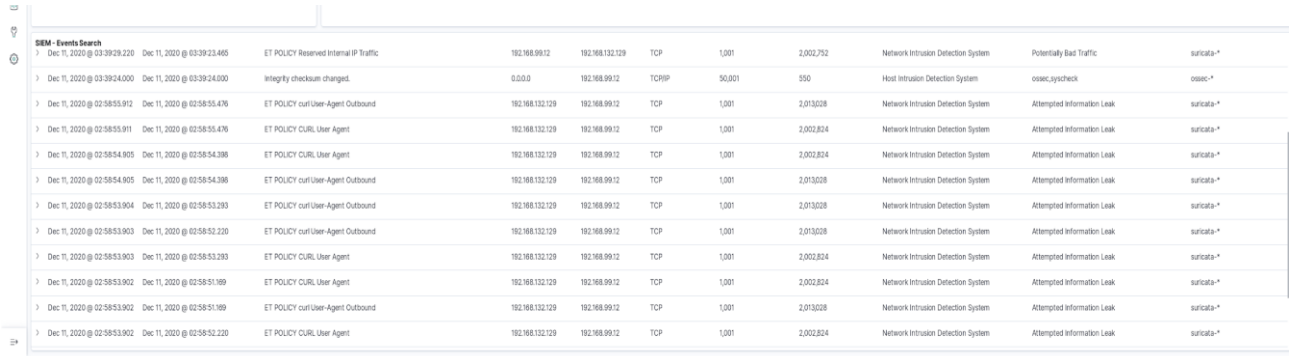

Рис. 3.15. Події, отримані від HIDS та NIDS

Для того, щоб провести аудит безпеки і перевірити, чи може впроваджена SIEM виявляти атаки(наприклад DoS-атаки), перш за все треба навчитися її проводити власноруч. Найпростіший спосіб досягнення цієї мети базується на використанні дистрибутива Kali Linux, а в цьому випадку точніше - програми hping3, популярного TCP-інструменту тестування проникнення, включеного в набір Kali Linux.

Користувачі Linux в якості альтернативної можливості можуть встановити інструментарій hping3 в свій існуючий дистрибутив Linux(CentOS 7):

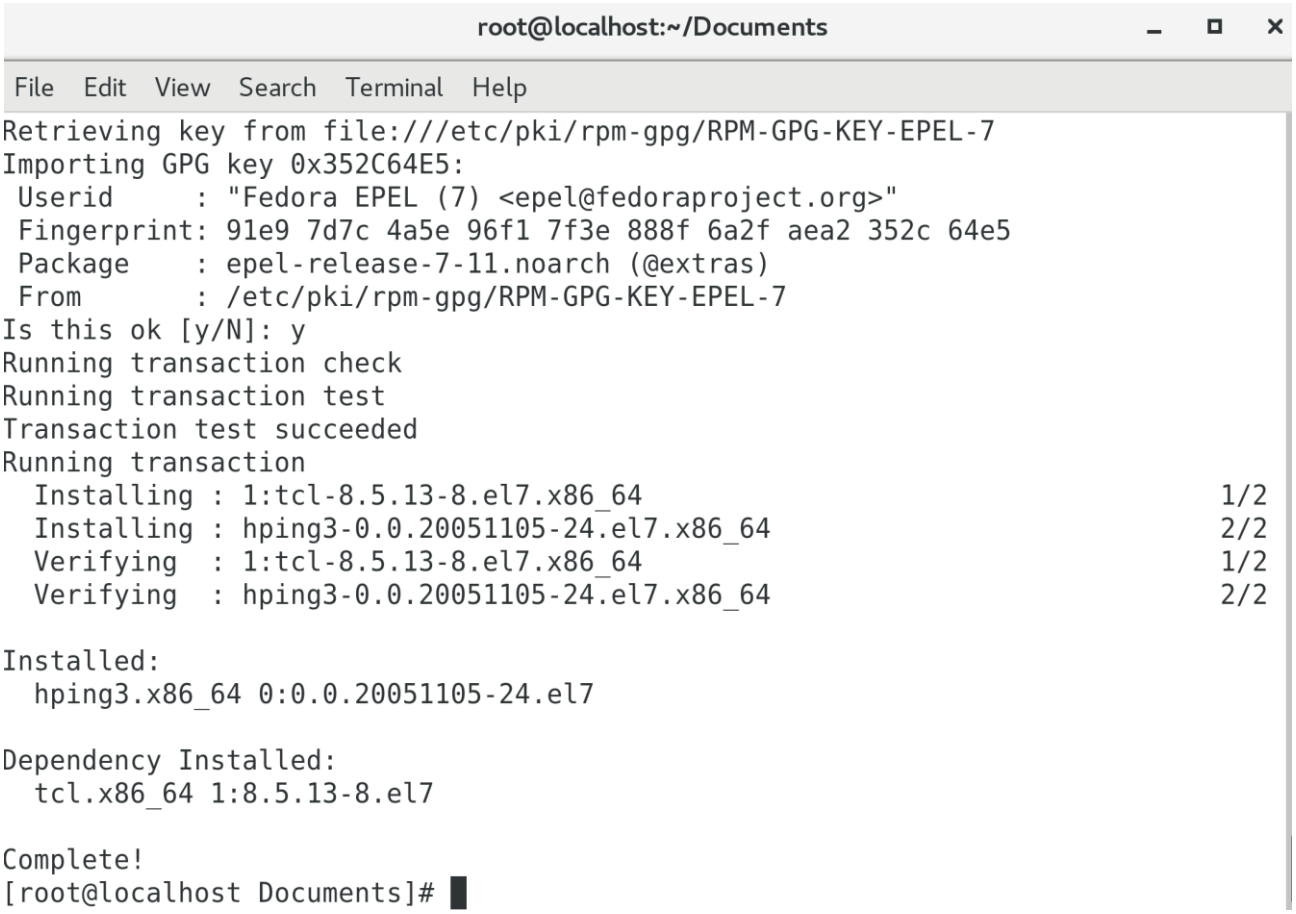

Рис. 3.16. Встановлення утиліти hping на CentOS 7

Так як в більшості випадків зловмисники будуть використовувати генератор пакетів hping або інший інструментарій з подібною функціональністю для підміни реальних IP-адрес випадковими, ми також звернемо фокус нашої уваги на цей момент. Наступний рядок дозволяє нам почати і направити атаку TCP SYN Flood на нашу ціль (192.168.99.14):

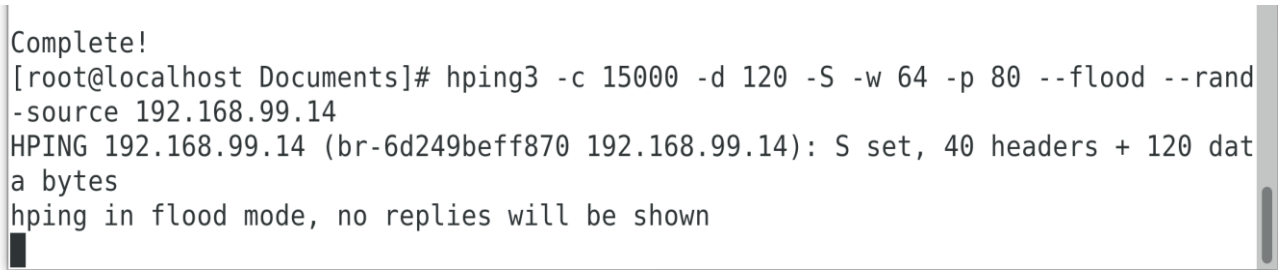

Рис. 3.17. Змодельований трафік атаки TCP SYN Flood

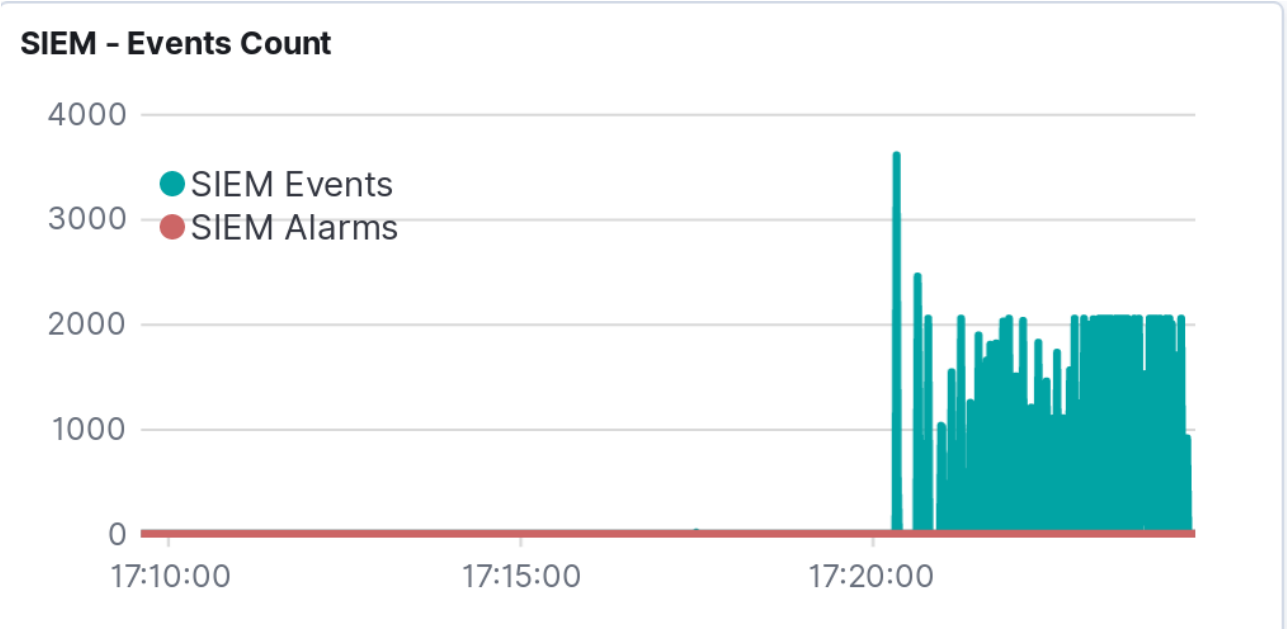

Рис. 3.18. Відображення дашборду подій ІБ

Переходимо в систему та бачимо відображення подій “Suricata stream packet with broken ack” від HIDS.

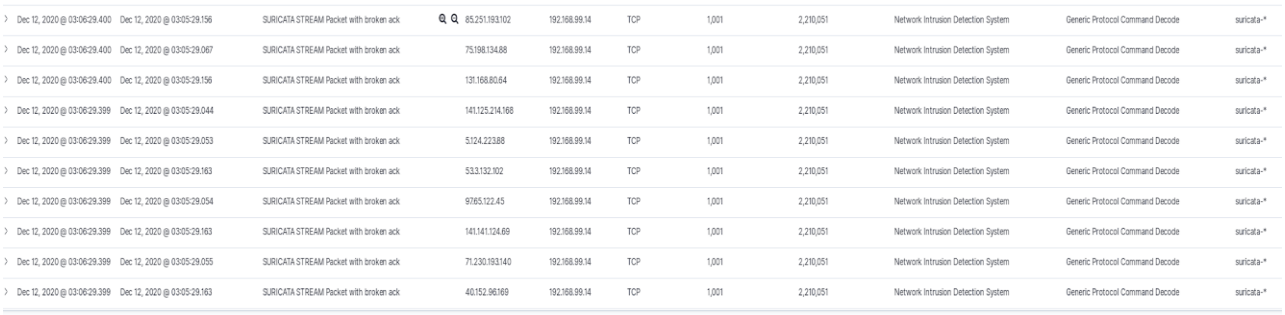

Рис. 3.19 Відображення подій від HIDS при моделюванні атаки TCP SYN Flood

Перевірка спостереження змодельованих подій в реальному часі пройшла вдало. Тепер перевіримо як працює виявлення сценарію на експлуатування вразливості CVE-2014-6271 («Bash»).

Для проведення експлуатації та виявлення вразливості “CVE-2014-6271 («Bash»)” додамо контейнер “ShellShock”: веб-сервер apache із середовищем інтерфейсу, використовуваного для зв'язку зовнішньої програми з веб-сервером, вразливим до Shellshock.

На рис. 3.20. представлений весь процес реалізації сценарію виявлення експлуатації вразливості CVE-2014-6271 («Bash») з використанням HIDS та NIDS.

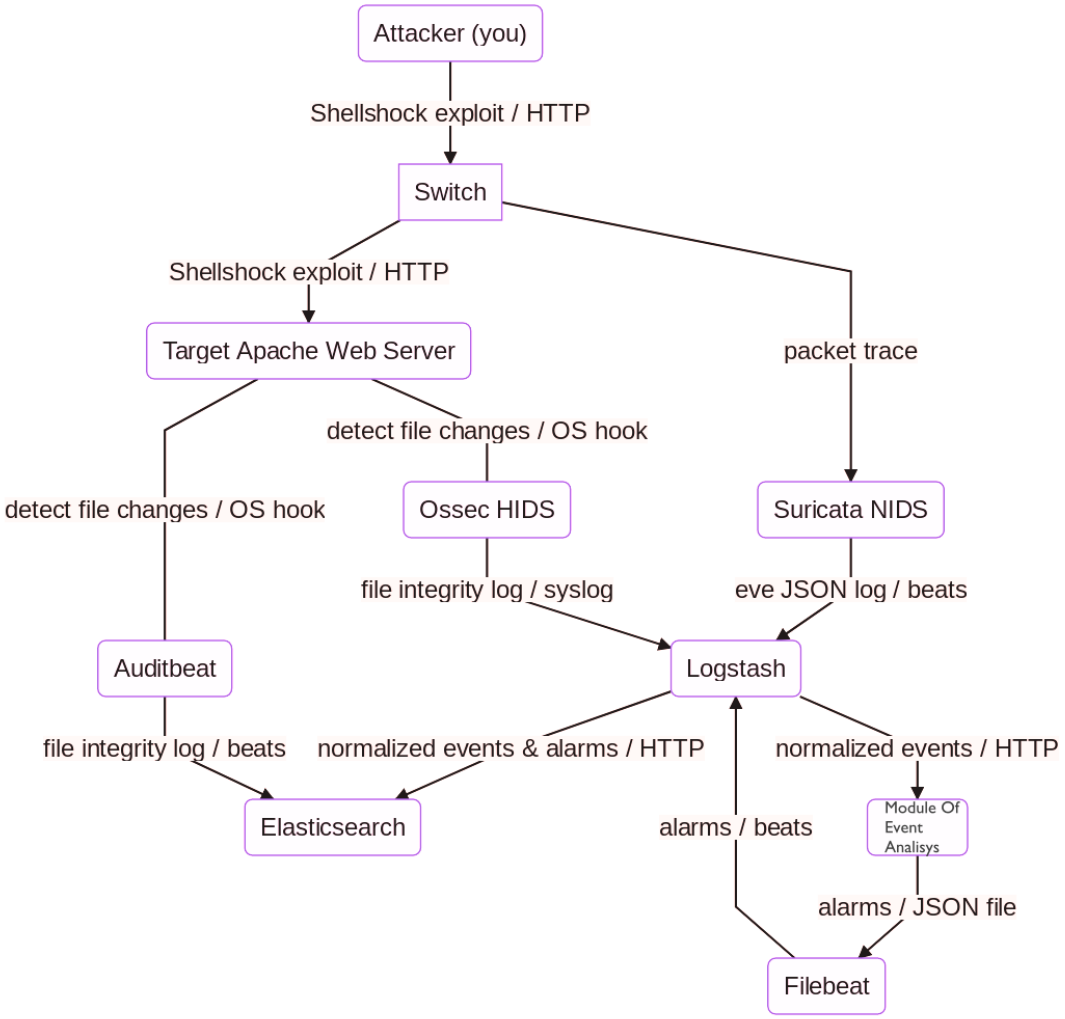

Рис. 3.20. Схема сценарію виявлення експлуатації вразливості CVE-2014-6271 («Bash»)

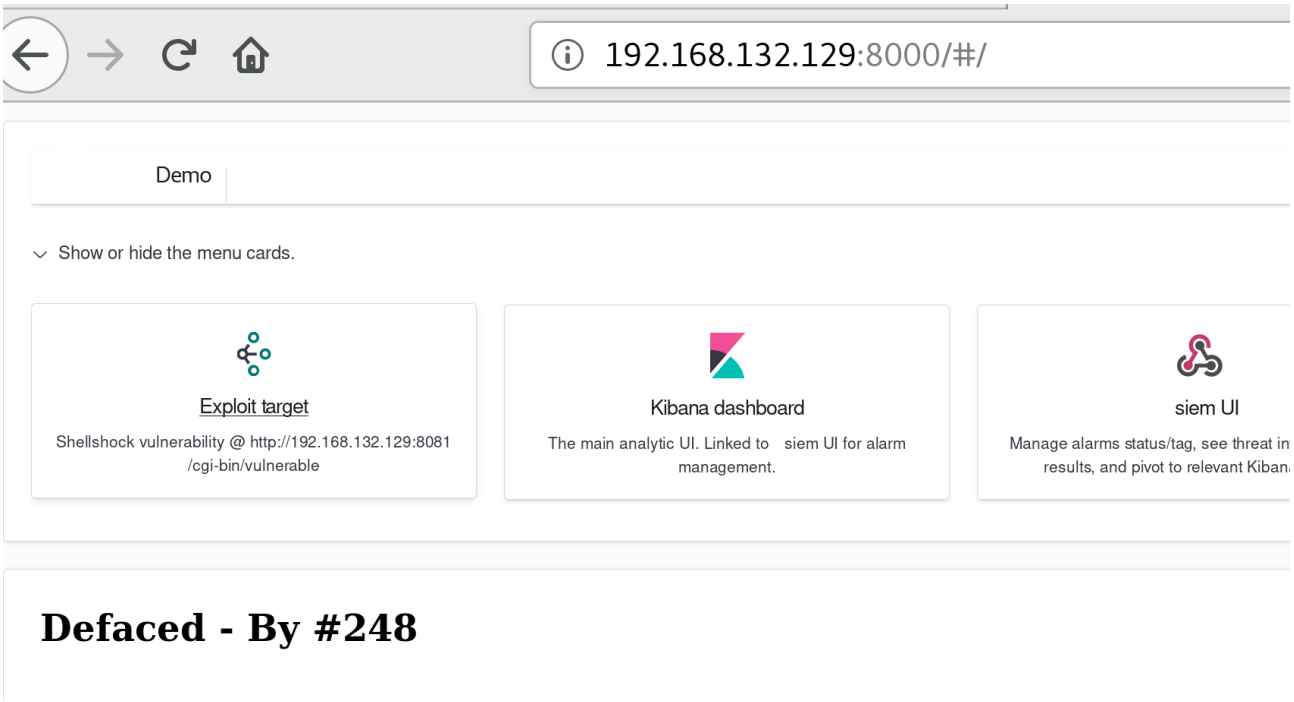

Рис. 3.21. Запуск скрипта з меню для виконання сценарію “Експлуатація вразливості Shellshock”

На рис. 3.22. зображена директива кореляції подій ІБ для виявлення загрози ІБ “Експлуатація вразливості Shellshock”.

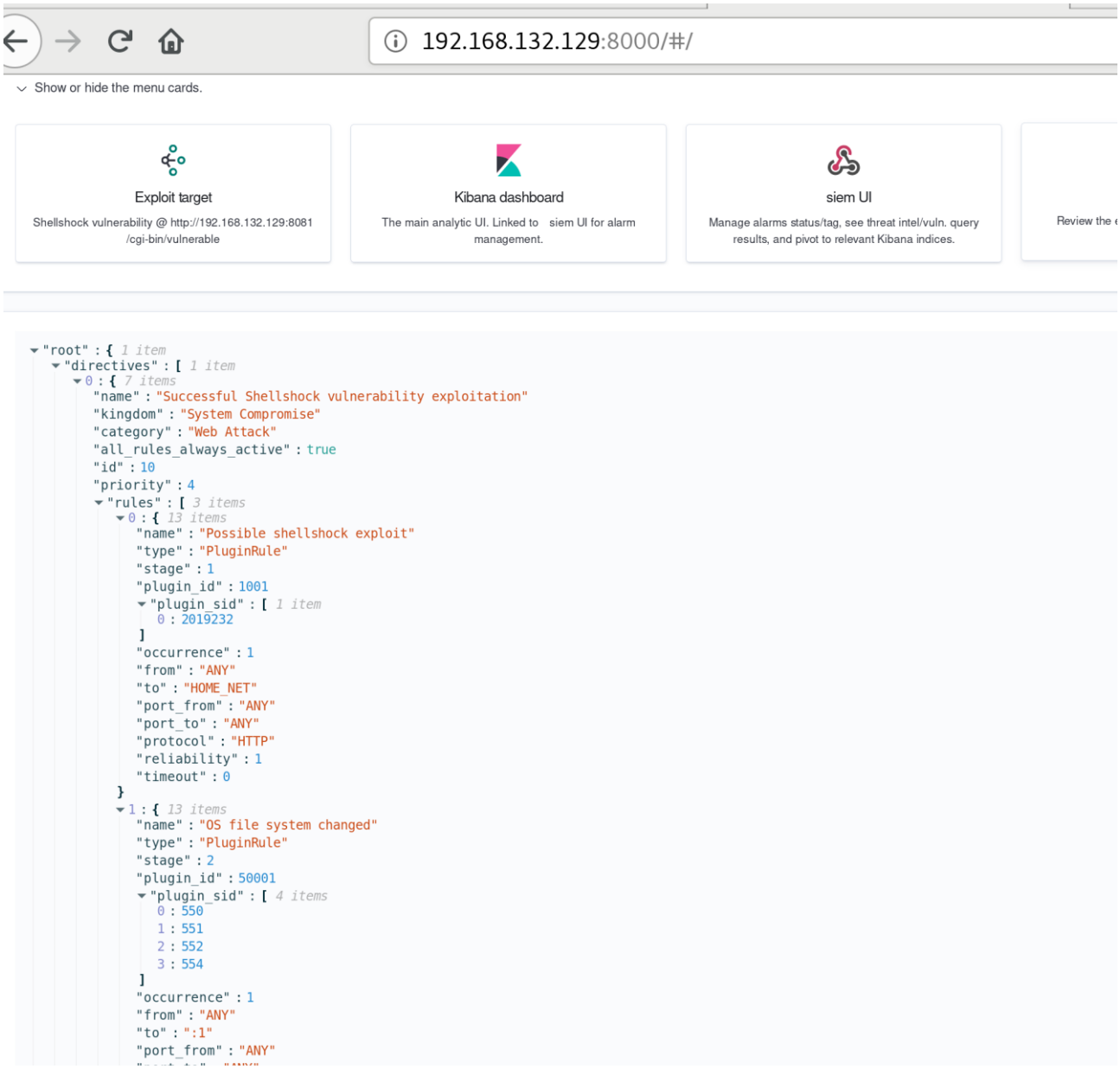

Рис. 3.22. Відображення в системі директиви кореляції подій ІБ для виявлення загрози ІБ “Експлуатація вразливості Shellshock”

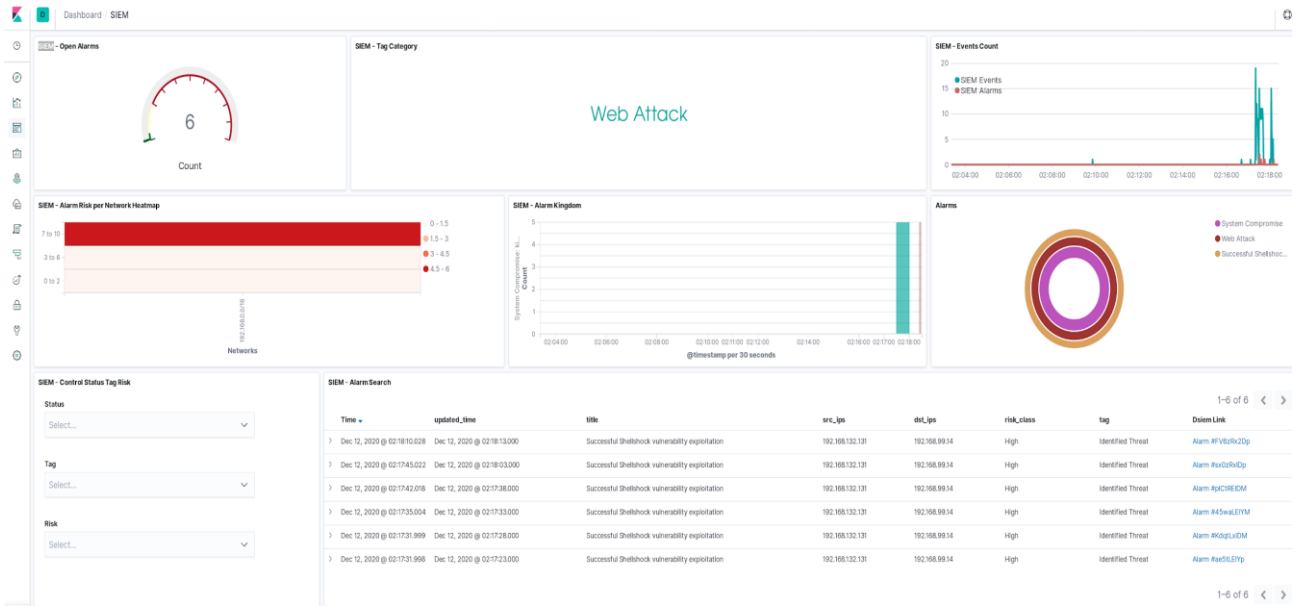

Рис. 3.23. Головне вікно дашбордів

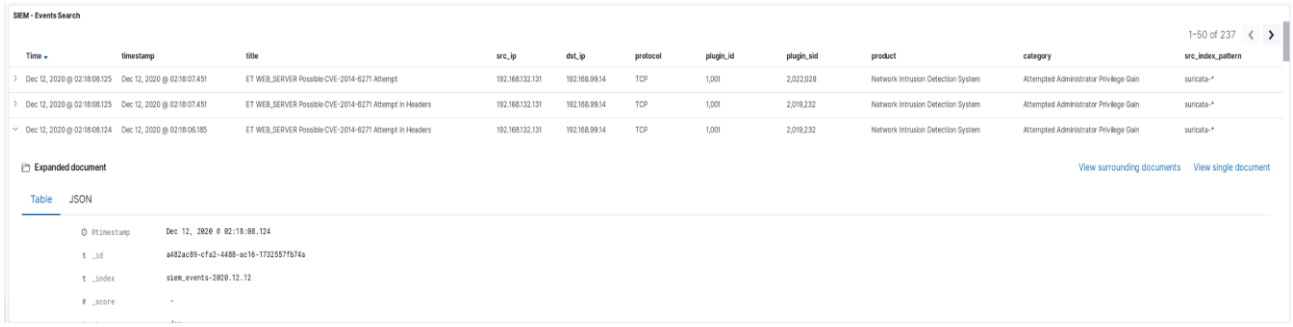

Рис. 3.24. Відображення подій під час APT

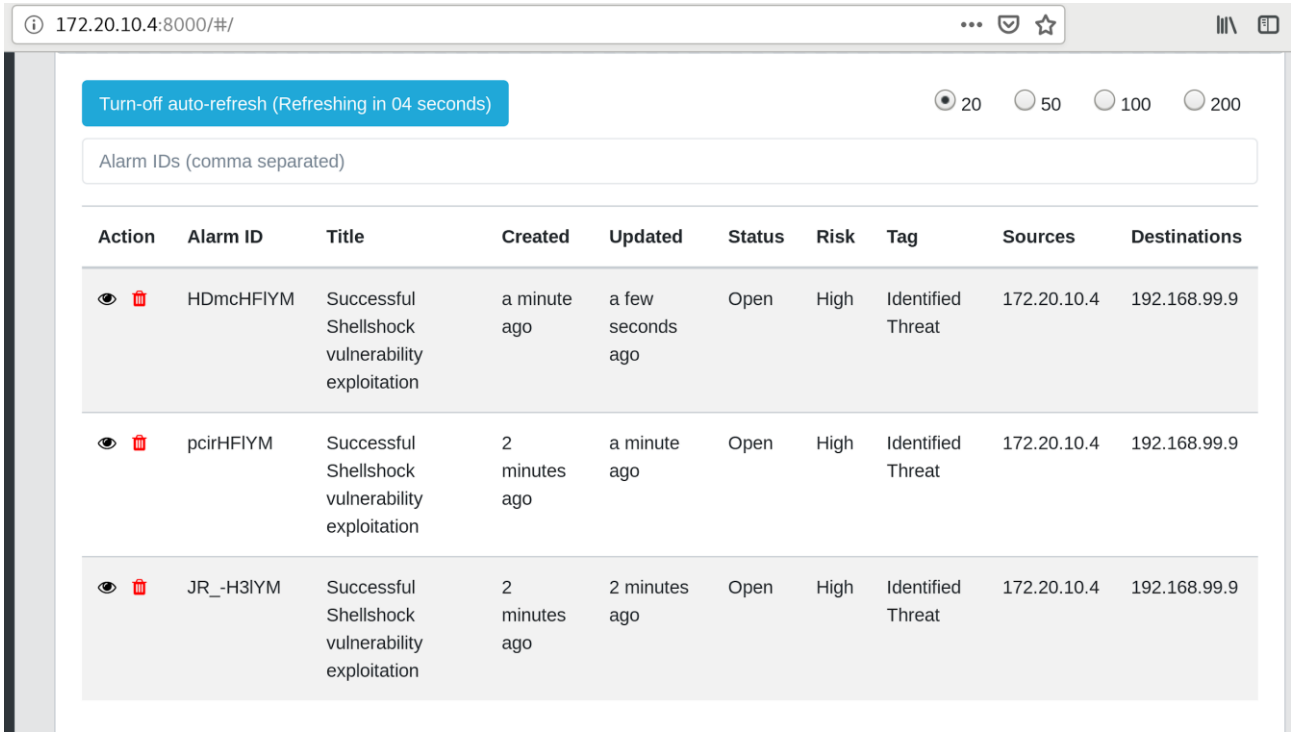

Рис. 3.25. Відображення сповіщень про загрозу ІБ “Експлуатація вразливості Shellshock”

3.6. Висновки до розділу

Для розробки програмного модулю аналізу подій ІБ побудована схема роботи модулю аналізу подій ІБ (Module of event analysis) на основі стеку ELK. Розгорнуто стек ELK і надалі всю роботу системи в контейнерах програмного забезпечення Docker.

До складу системи захисту мережі від несанкціонованого доступу включено підсистеми, що виявляють відомі вразливості(OSSEC, Suricata) та надсилають ці події для подальшої кореляції за допомогою директив. Програми цих підсистем, аналізують, який трафік надходить в мережу.

Для сповіщення про експлуатації вразливості на ресурсах нашої системи створюються директиви кореляції подій ІБ від тих джерел, які нам необхідні.

Директива кореляції - набір логічних правил, які дозволяють здійснювати порівняння параметрів подій ІБ, а також їх кількості і частоти з заданими показниками для виявлення інцидентів інформаційної безпеки. Під кожну компанію аналізується та проектується кастомізований набір директив кореляції.

Розроблено директиву кореляції подій ІБ для виявлення сценарію експлуатації вразливості CVE-2014-6271 («Bash»).

Підтримка методики шаблонів поведінки необхідна будь-якій системі: пакети директив кореляцій відображають можливий ланцюг подій (аномалій), який відповідає моделі реальної цільової атаки(APT).

ВИСНОВКИ

Результатом виконаної роботи є розробка модулю аналізу подій ІБ для стеку ELK.

У процесі виконання роботи отримані наступні результати:

1. Проведено порівняння існуючих програмних рішень класу SIEM:IBM QRadar, HP Arcsight, Splunk та виявлено їх переваги та недоліки, що дало основу для подальших досліджень в даній області.

2. Проведено дослідження та проаналізовано підходи до кореляції подій, архітектуру та завдання модулю аналізу подій ІБ, розроблено модель системи з інтегрованим модулем на основі стеку технологій ELK та модуль аналізу подій ІБ.

3. Розроблено та протестовано програмний модуль аналізу подій для ELK Stack з використанням директиви кореляції подій ІБ для виявлення сценарію експлуатації вразливості CVE-2014-6271 («Bash»), модулів HIDS, NIDS на основі яких реалізована, побудована система управління подіями і інцидентами інформаційної безпеки.

4. Проведено інтеграцію розробленого модулю до стеку ELK. Дослідження системи було проведено шляхом багатократного аналізу логів на виявлення сценарію експлуатації вразливості CVE-2014-6271 («Bash») за умови значень вхідних параметрів переданих журналів подій ІБ, що відповідають умовам правил кореляційного аналізу подій ІБ. Використання даної системи для відслідковування позитивних спрацювань на загрози інформаційній безпеці організації спрямовано на підвищення рівня захищеності ІС, можливості подальшого розслідування інцидентів ІБ.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. SIEM – 30.01.2020. – [Електронний ресурс]. – Режим доступу: World Wide Web. – URL: https://ru.wikipedia.org/wiki/SIEM
2. Розвинена стала загроза – 18.05.2020. – [Електронний ресурс]. – Режим доступу: World Wide Web. – URL: https://uk.wikipedia.org/wiki/Розвинена_стала_загроза
3. Методы и средства, применяемые в SIEM-системах при мониторинге информационной безопасности. Введение. Часть 1 – 30.07.2018. – [Електронний ресурс]. – Режим доступу: World Wide Web. – URL: https://www.securitylab.ru/blog/company/gamma/344431.php
4. IBM QRadar SIEM – [Електронний ресурс]. – Режим доступу: World Wide Web. – URL:https://www.ibm.com/ru-ru/products/qradar-siem
5. Продуктовая линейка HP Arcsight. – [Електронний ресурс]. – Режим доступу: World Wide Web. – URL: http://arcsight-russia.ru/products-hp-arcsight/products-hp-arcsight
6. K. Kavanagh, T. Bussa, G. Sadowski. Magic Quadrant for Security Information and Event Management. Gartner, 3 December 2018
7. Splunk - Overview. – [Електронний ресурс]. – Режим доступу: World Wide Web. – URL: https://www.tutorialspoint.com/splunk/splunk_overview.htm
8. 3 товарища в поиске и аналитике Big Data: Elasticsearch, Logstash и Kibana. – 08.06.2020. – [Електронний ресурс]. – Режим доступу: World Wide Web. – URL: https://www.bigdataschool.ru/blog/what-is-elk.html .
9. H. Karlzen, „An Analysis of Security Information and Event Management Systems: The Use of SIEMs for Log Collection, Management, and Analysis.,“ p. 45, January 2009.
10. Rafał Kuć, Marek Rogoziński, “Mastering ElasticSearch”, October 2013.
11. 2016 SIEM Efficiency Survey Report.– 2016. – – [Електронний ресурс]. – Режим доступу: World Wide Web. – URL: https://www.netwrix.com/2016_siem_efficiency_survey_report.html

12. Kang D., Na J. A Rule Based Event Correlation Aproach for Physical and Logical Security Convergence // IJCSNS Intern. Journal of Computer Science and Network Security. 2012. Vol. 12. N 1. P. 28–32.
13. Elshoush H. T., Osman I. M. Intrusion Alert Correlation Framework: An Innovative Approach. IAENG Transactions on Engineering Technologies, 2013, vol. 229, pp. 405–420.
14. Klinger S., Yemini S., Yemini Y., Ohsie D., Stolfo S. A Coding Approach to Event Correlation. Proc. of the Fourth Intern. Symp. on Integrated Network Management IV, Adarshpal S. Sethi, Yves Raynaud, and Fabienne Faure-Vincent(Eds.), 1995, pp. 266–277.
15. Dwivedi N., Tripathi A. Event Correlation for Intrusion Detection Systems. IEEE Intern. Conf. “Computational Intelligence & Communication Technology” (CICT), 2015, pp. 133–139.
16. Kidmose E., Stevanovic M., Pedersen J. M. Correlating Intrusion Detection Alerts on Bot Malware Infections using Neural Network. Intern. Conf. “Cyber Security and Protection of Digital Services (Cyber Security)”, 2016, pp. 1–8.
17. Xuewei F., et al. An Approach of Discovering Causal Knowledge for Alert Correlating Based on Data Mining //Dependable, Autonomic and Secure Computing(DASC): 2014 IEEE 12th Intern. Conf. IEEE, 2014. P. 57–62.
18. Wang C., Chiou Y. Alert Correlation System with Automatic Extraction of Attack Strategies by Using Dynamic Feature Weights // Intern. Journal of Computer and Communication Engineering. 2016. Vol. 5. N 1.P. 1–10.
19. Kotenko I., Doynikova E. Evaluation of Computer Network Security based on Attack Graphs and Security Event Processing // Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications. 2014. Vol. 5. N 3. P. 14–29.
20. Katz G., Elovici Y., Shapira B. CoBAn: A Context Based Model for Data Leakage Prevention // Information Sciences. 2014. Vol. 262. P. 137–158.

21. Morin B., Me L., Debar H., Ducasse M. A Logic-Based Model to Support Alert Correlation in Intrusion Detection // Information Fusion. 2009. Vol. 10. N 4. P. 285–299.
22. Eckmann S. T., Vigna G., Kemmerer R. A. An Attack Language for State-Based Intrusion Detection //Journal of Computer Security. 2002. N 10 (1–2). P. 71.
23. Totel E., Vivinis B. A Language Driven Intrusion Detection System for Event and Alert Correlation // Security and Protection in Information Processing Systems. 2004. Vol. 147. P. 208–224.
24. Viinikka J., Debar H., Me L., Lehikoinen A., Tarvainen M. Processing Intrusion Detection Alert Aggregates with Time Series Modelling // Information Fusion. 2009. Vol. 10. N 4. P. 312–324.
25. Treinen J., Thurimella R. A Framework for the Application of Association Rule Mining in Large Intrusion Detection Infrastructures // Recent Advances in Intrusion Detection (RAID). 2006. Vol. 4219. P. 1–18.
26. Jakobson G. Mission Resilience // Cyber Defense and Situational Awareness. — Springer International Publishing, 2014. — P. 297–322.
27. Gao J., Jiang G., Chen H., and Han J. Modeling Probabilistic Measurement Correlations for Problem Determination in Large-Scale Distributed Systems // Proc. IEEE, Montreal, Canada, June 22–26, 2009. P. 623–630.
28. Naukudkar K. B., Ambawade D. D., Bakal J. W. Enhancing Performance of Security Log Analysis using Correlation-Prediction Technique // Proc. of Intern. Conf. on ICT for Sustainable Development, Singapore, Feb. 26, 2016. Springer, 2016. Vol. 409. P. 635–643.
29. Vianello V. A Scalable SIEM Correlation Engine and Its Application to the Olympic Games IT Infrastructure // Intern. Conf. on Availability, Reliability and Security, Regensburg, 2013. P. 625-629.
30. Butun I., Morgera S. D., Sankar R. A survey of intrusion detection systems in wireless sensor networks // IEEE Communications Surveys & Tutorials. 2014. Vol. 16, № 1. P. 266-282.

31. Runtime correlation engine for system monitoring and testing / V. Holub, T. Parsons, P. O'Sullivan, J. Murphy // In ICAC'09: Proc. of the 6th Intern. Conf. on Autonomic computing, New York, USA, 2009. P. 43-44.
32. Donghai T., Changzhen H., Qi, Y., Jianqiao W. Hierarchical Distributed Alert Correlation Model // In Proc. of the 2009 Fifth Intern. Conf. on Information Assurance and Security. Vol. 02 (IAS'09). Vol. 2. IEEE Computer Society, Washington, DC, USA. P. 765-768.
33. Scalable Run-Time Correlation Engine for Monitoring in a Cloud Computing Environment / M. Wang, V. Holub, T. Parsons, J. Murphy, P. O'Sullivan // In Proc. of the 2010 17th IEEE Intern. Conf. and Workshops on the Engineering of Computer-Based Systems (ECBS '10). IEEE Computer Society, Washington, DC, USA. P. 29-38.
34. Streamcloud: an elastic and scalable data streaming system / V. Gulisano, R. Jimenez-Peris, M. Patino-Martinez, C. Soriente, P. Valduriez // ERCIM NEWS. 2012. Vol. 89. P. 32-33
35. Luckham D. The power of events: An introduction to complex event processing in distributed enterprise systems // Intern. Workshop on Rules and Rule Markup Languages for the Semantic Web. Springer Berlin Heidelberg, 2008.
36. Etzion O., Niblett P. Event Processing in Action. № ISBN: 9781935182214. Manning Publications Co., 2010.
37. MGller A. Event Correlation Engine // Computer engineering and networks laboratory / TIK Institut fGr Technische Informatik und Kommunikationsnetze, 2009.