DOI 10.15276/imms.v13.no3-4.282
УДК 004.056.53

Informatics and Mathematical Methods in Simulation
Vol.13 (2023), No. 3-4, pp. 282-292

РОЗРОБКА СИСТЕМИ МОНІТОРИНГУ ПОДІЙ ІНФОРМАЦІЙНОЇ БЕЗПЕКИ

П.Ю. Паталашко, Н.І. Кушніренко, Н.Г. Козаченко, Н.В. Бойко

Національний університет «Одеська політехніка»
1 Шевченка пр., Одеса, 65044, Україна
e-mail: infsec2011@gmail.com

У випадку, коли проблема виникає у мережі, IT-спеціалісти повинні бути впевнені, що будуть повідомлені про неї та отримають необхідну інформацію для її вирішення й виключення подібних ситуацій в майбутньому. Важливим елементом моніторингу мережі є сповіщення. Коли в мережі виникає аномальна поведінка, система повинна негайно повідомити IT-спеціалістів, щоб вони могли швидко реагувати. Саме тоді процес відновлення стану мережі стає найбільш ефективним. Моніторинг мережі в обов'язковому порядку повинен включати автоматичне отримання повідомлень про збої та події, які можуть впливати на продуктивність. У разі помилки, такі сповіщення мають бути надіслані через електронну пошту, SMS або інший зручний канал. Динамічні мережі вимагають систем, які могли б ідентифікувати вразливості в реальному часі. Тому деякі системи моніторингу мережі надають можливості для збору даних про безпеку, такі як мережеві журнали, журнали додатків та повідомлення про безпеку. Ці дані збираються та аналізуються для виявлення аномалій, і в разі появи загроз можливо надіслати сповіщення відповідному IT-спеціалісту. Таким чином, система моніторингу мережі дає можливість: забезпечувати неперервний робочий процес, відстежувати стан мережі та вчасно реагувати на можливі проблеми, поліпшувати ефективність роботи мережі, підвищувати надійність та безпеку мережі. Враховуючи це, можна зробити висновок, що для успішного функціонування мережі дуже важливо проводити їх моніторинг та вчасно реагувати на виникнення проблем. Для цього потрібно використовувати спеціалізовані системи моніторингу, які здатні швидко аналізувати стан мережі, визначати проблемні місця та повідомляти про них відповідних спеціалістів. Ця робота була зосереджена на аналізі наявних систем моніторингу мережі, висвітлено їх плюсів та мінусів. Досліджено технології та методики, що застосовуються для створення таких рішень. Розроблено нову систему моніторингу, спрямовану на оптимізацію процесу відстеження статусу мережевої безпеки. Збагачено функціонал системи відправкою сповіщень, а також впровадженням правил відстежування аномальної поведінки в мережі, що допоможе краще виділяти зловмисницькі дії та вчасно попереджувати спеціалістів.

Ключові слова: моніторинг, ELK-стек, OSSEC, ElastAlert, інцидент, логи, метрики.

Вступ. Коли підприємство розпочинає займатися питанням інформаційної безпеки, йому потрібно впровадити масу різноманітних систем. Це можуть бути антивірус, мережевий брандмауер, мережева та серверна системи виявлення вторгнень, міжсистемний фаєрвол, сканер уразливостей, система обліку цілісності, та багато іншого. Список інструментів, які можуть використовуватися в структурі, є досить широким. Оскільки безпека являє собою не лише стан системи (у традиційному розумінні), але й процеси, важливою складовою яких є моніторинг подій з інформаційної безпеки [1]. У будь-якому випадку виникає запитання про централізоване спостереження та аналіз журналів подій, які згенеровані перерахованими системами в різних масштабах.

На сьогоднішній день існує ряд рішень для організації моніторингу. Наприклад, вкрай популярним рішенням є Splunk. Цей інструмент збирає, індексує та співвідносить дані в реальному часі у сховищі з можливістю пошуку і виконання різноманітних запитів за заданими параметрами. За отриманими результатами можна створювати графіки, звіти, інформаційні панелі та різноманітні візуалізації [1]. Splunk має багато переваг, включаючи збір, відстеження, моніторинг та аналіз великих обсягів даних, які можна виконати в історичному режимі пошуку або в реальному часі. Проте він досить вартісний, особливо при великих обсягах даних, а його можливості створення власних правил кореляції обмежені у термінах складності та гнучкості.

Ще одне рішення для організації моніторингу, що було розглянуто – LogPoint [2]. Для даного інструменту можна відзначити вибір методу зберігання логів: залежно від цінності даних, ви можете вказати різні періоди зберігання, а також можливість створення складних кореляційних пошукових запитів. Крім того, можна налаштувати автоматичне сповіщення, використовуючи будь-який створений запит, що активує електронного листа при виявленні події. З недоліків LogPoint - недостатня інтуїтивна структура, яка ускладнює пошук деяких функцій. Хоча ціна на систему є доволі конкурентоспроможною, відсутність безкоштовної версії обмежує можливість її використання невеликими підприємствами. Спільнота користувачів LogPoint не є такою широкою і активною, як у інших пропозицій на ринку.

Іншим досить популярним рішенням є ELK-стек, що приваблює відкритим кодом, який дає можливість впровадити моніторинг будь-кому, незалежно від розміру компанії [3]. Це сприяє легкому впровадженню і гнучкості у налаштуванні системи для конкретних потреб організації. Недоліками даної системи є відсутність правил кореляції і механізму створення правил для відстежування характеру логів [4]. Також без додаткових компонентів в системі відсутній вбудований механізм сповіщення про потенційні загрози, його потрібно встановлювати окремо [5]. З іншого боку ELK-стек має суттєві переваги:

– ELK є безкоштовною системою (не зважаючи на витрати серверів). Попри вимоги витрат на налагодження та підтримку, вона забезпечує кращий баланс вартості і потужності;

– дозволяє збирати метрики, обробляти великі обсяги даних;

– відкритий код, що надає значну гнучкість у впровадженні;

– швидкість розгортування і легку масштабованість;

– має зручний API для створення запитів і можливості програмної інтеграції з іншими продуктами;

– добре розвинута спільнота користувачів, що забезпечує постійне оновлення та вдосконалення системи, а також швидке вирішення виникаючих проблем.

Беручи до уваги вищеописані недоліки і переваги, було вирішено розглянути потенційні способи подолання і збагачення функціоналу ELK-стеку для розробки власної системи моніторингу. В якості додаткових рішень з відкритим програмним кодом доцільно використати: Elastalert – для організації відстежування і відправки сповіщень, OSSEC – у якості елементу для конфігурації правил порушення безпеки.

Мета і задачі дослідження. *Метою роботи* є підвищення ефективності відстежування стану безпеки комп’ютерної мережі шляхом розробки системи моніторингу подій. За рахунок такого підходу підвищиться рівень безпеки організації, та значно знизиться час, що пройшов від моменту виникнення конкретних подій, до моменту їх нейтралізації. Альтернатив у відкритих джерелах

не було знайдено, що підвищує цінність розробки. В процесі виконання даної роботи необхідно розв'язати наступні задачі:

– проаналізувати та виявити найбільш важливі метрики для збору, що відображають стан мережі, а також найчастіші можливі порушення безпеки на серверах;

– обрати і описати складові частини для розробки системи моніторингу;

– розробити програмний продукт для візуалізації централізованого моніторингу комп'ютерної мережі.

Основна частина. Моніторинг мережі – процес відстежування дієздатності та стабільності мережі, її функціонування та продуктивності в рамках складних мережевих структур [6]. Він об'єднує процеси спостереження та аналізу мережевих компонентів таких, як роутери, комутатори та брандмауери, а також з'єднань між ними. Моніторинг мережі також охоплює керування різними рівнями даних, кінцевими мережевими вузлами та інтерфейсами.

Роутери, комутатори та вузли створюють сполучення між великою кількістю робочих станцій і ключовими програмними застосунками, розміщеними на численних серверах і в Інтернеті. Крім того, налаштовані безліч інструментів і застосунків безпеки та комунікацій, включаючи брандмауери, віртуальні приватні мережі (VPN), і антивіруси.

Перевірка роботи та продуктивності інтерфейсів стосовно їхніх потенційних збоїв сприяє діагностуванню, оптимізації і контролю різних мережевих ресурсів як локально, так і на відстані. За допомогою даних, представлених у вигляді таблиць, діаграм, графів, інформаційних панелей та звітів, моніторинг мережі дозволяє системним адміністраторам зменшити середній час відновлення (MTTR), а також розв'язати проблеми мережевої продуктивності в режимі реального часу. Коли подібні проблеми виявлені, система повідомляє системних адміністраторів безпосередньо або за допомогою підтримки, дозволяючи ним найшвидше розв'язати проблему.

Розуміння архітектури та складності мережі, обізнаність про роботу кожного її складового елемента у будь-який момент – все це важливі чинники, які сприяють успішному підтриманню стабільності та цілісності мережі компанії і її клієнтів. В мережі може бути тисячі точок даних для моніторингу, тому край важливим є доступ до значущої, точної та актуальної інформації в будь-який час. Системні адміністратори повинні постійно бути в курсі всього, що відбувається в кожному сегменті мережі.

Мережа, як правило, має внутрішніх та зовнішніх користувачів, включаючи співробітників, клієнтів, партнерів та інші сторони. Відмова мережі може мати різний ефект на бізнес, в залежності від типу користувача. Наприклад, якщо працівники не можуть отримати доступ до потрібної інформації для виконання роботи, це може призвести до зниження продуктивності, фінансових втрат і, ймовірно, шкоди репутації компанії в майбутньому.

Кожний компонент мережі є потенційною точкою відмови. Тому надзвичайно важливим є розроблення стратегії, що мінімізує можливість збою. Таким чином, якщо один сервер або роутер зазнає збою, інший може автоматично під'єднатися до мережі для зменшення ефекту від відмови головного обладнання. Не всі проблеми можуть бути прогнозовані й розв'язані до моменту, коли реальні загрози стануть очевидними. Але якщо здійснювати активний контроль мережі в режимі реального часу, можливо виявити та розв'язати проблеми до того, як вони набудуть глобальних обсягів. Наприклад, перевантажений сервер може бути замінений, перш ніж він зазнає збою, але це можливо лише при своєчасному отриманні цієї інформації.

Система моніторингу мережі може стати важливим інструментом для подальшого розвитку та планування мережі. Завдяки своїй здатності інформувати IT-спеціалістів про використання окремих елементів мережі та передбачати потенційні виклики, що можуть призвести до перевантаження, така система може сприяти ефективній адаптації мережі до швидкого зростання бізнесу або збільшення числа користувачів.

Інструменти моніторингу мережі забезпечують системному адміністратору постійний доступ до актуальної інформації про стан мережі, що дає можливість оперативно реагувати на виникнення проблем та вирішувати їх вчасно. Саме таким функціоналом володіє система, що буде лежати в основі рішення для моніторингу мережі. Як з'ясовано раніше, ELK-стек немає можливості відправки сповіщень і налаштування правил для аналізу записів у системних журналах. Тому далі буде детально описані рішення, що будуть використовуватися для доповнення функціоналу системи.

У якості рішення для відправки сповіщень було обрано легковісний ElastAlert. ElastAlert – це простий фреймворк для сповіщення про аномалії, сплески та інші патерни з даних в Elasticsearch [7]. Elastalert дозволяє створювати правила, які будуть описувати будь-які цільові ситуації і сповіщати про них. Є можливість налаштувати різні типи правил, такі як зміни в частоті подій, різке збільшення або зменшення кількості подій, або навіть кастомізовані правила, що використовують власні алгоритми користувача для виявлення аномалій. Це налаштовується набором правил, кожне з яких визначає запит, тип правила і набір оповіщень.

Фреймворк працює, поєднуючи Elasticsearch з двома типами компонентів, типами правил і сповіщеннями. На Elasticsearch періодично відправляється запит і дані з запиту (логи) передаються до типу правила, який визначає, чи знайдено збіг. Коли відбувається збіг, він передається одному або декільком правилам сповіщення, які вживають заходів на основі цього збігу.

Кожне правило визначає запит, який потрібно виконати, параметри, за якими спрацьовує збіг, і список сповіщень, які потрібно запустити для кожного збігу. Кожне правило являє собою окремий YAML-файл, який має містити наступні обов'язкові поля [7]:

– «es_host» і «es_port» повинні вказувати на кластер Elasticsearch, до якого ми робимо запит;

– «name»: унікальне ім'я для правила. ElastAlert не спрацює, якщо два правила мають однакову назву;

– «type»: кожне правило має свій тип, який може приймати різні параметри. Тип «frequency» означає «Сповіщати, коли відбувається більше ніж «num_events» протягом часового інтервалу»;

– «index»: назва індексу(ів) для запиту. Якщо використовується Logstash, за замовчуванням індекси будуть відповідати "logstash-*";

– «num_events»: параметр є специфічним для типу frequency і є пороговим значенням для спрацьовування оповіщення;

– «timeframe»: період часу, за який має відбутися num_events;

– «filter»: список фільтрів Elasticsearch, які використовуються для фільтрації результатів. Тут ми маємо фільтр за одним терміном для документів, у яких «деяке_поле» збігається з «деяким_значенням». Якщо фільтри не потрібні, слід вказати порожній список: filter: [];

– «alert»: список цілей, яким будуть вислані сповіщення. Сповіщення електронною поштою потребує SMTP-сервера для надсилання пошти. За замовчуванням він намагатиметься використовувати localhost. Це можна змінити за допомогою параметра smtp_host. Ще однією популярною ціллю є «telegram», для якого потрібен токен і канал;

– «email»: це список адрес, на які будуть надіслані сповіщення.

Приклад конфігурації правила зображено на рис.1.

```
name: Load average for 1 minutes over 2
type: any
index: logstash-*
num_events: 2
timeframe:
  minutes: 2
filter:
  - query:
      query_string:
        query: "system.load.1:>1"
alert:
  - "telegram"
telegram_bot_token: 6973162738:AAFyWuCEOVhu2hKlPACvj
telegram_room_id: "@elk_alert_scream"
```

Рис.1. Приклад конфігурації правила ElastAlert

У якості системи для відстеження порушень в системі було обрано OSSEC. OSSEC – це система виявлення вторгнень з відкритим вихідним кодом. Вона виконує аналіз журналів, перевірку цілісності, моніторинг реєстру Windows, виявлення руткітів, оповіщення в реальному часі та активне реагування [8]. Вона працює на більшості операційних систем, включаючи Linux, OpenBSD, FreeBSD, Mac OS X, Solaris і Windows. Вона поєднує в собі всі аспекти HIDS (Host Intrusion Detection System – виявлення вторгнень на основі хостів), моніторингу журналів та управління інцидентами безпеки (SIM)/управління інформацією та подіями безпеки (SIEM) в одному простому, потужному рішенні з відкритим вихідним кодом. Основні функції:

– перевірка цілісності файлів. Будь-яка атака супроводжується зміною системи. Мета перевірки цілісності файлів (File Integration Monitoring) – виявити ці зміни і попередити, коли вони відбудуться. Це може бути атака, зловживання з боку співробітника або навіть друкарська помилка адміністратора, про будь-яку зміну файлу, каталогу або реєстру вам буде повідомлено;

– моніторинг журналів. Кожна операційна система, додаток і пристрій у мережі створюють журнали подій, щоб повідомити вам про поточний стан системи. OSSEC збирає, аналізує та опрацьовує ці журнали, щоб повідомити вам, якщо відбувається щось підозріле (атака, зловживання, помилки тощо). Наприклад, на клієнтському комп'ютері була встановлена програма, або були внесені зміни правил у вашому брандмауері чи фаєрволі.

Обробка журналів виконується всередині OSSEC процесами logcollector і analysisd. Перший збирає події, а другий аналізує (розшифровує, фільтрує і класифікує) їх. Це робиться в режимі реального часу, тому як тільки подія записується, OSSEC обробляє її. OSSEC може читати події з внутрішніх файлів журналів, з журналу подій Windows, а також отримувати їх безпосередньо через віддалений syslog.

OSSEC постачається з набором вбудованих правил, які визначають типову активність, яку слід відстежувати в системах. За замовчуванням ці правила включають широкий спектр активностей, таких як неуспішні спроби входу, отримання доступу до важливих файлів та зміну конфігурації системи. Також користувачі можуть створювати власні правила з урахуванням специфічних вимог до їхнього середовища [9]. Приклад налаштування такого правила наведений на рис.2.

```
<rule id="5700" level="0" noalert="1">
  <decoded_as>sshd</decoded_as>
  <description>SSHD messages grouped.</description>
</rule>

  <rule id="5710" level="5">
  <if_sid>5700</if_sid>
  <pcre2>illegal user|invalid user</pcre2>
  <description>Attempt to login using a non-existent user</description>
  <group>invalid_login,authentication_failed,</group>
</rule>
```

Рис.2. Приклад OSSEC для виявлення несанкціонованого входу до SSH

Існує головний елемент, що групує правила по типу «sshd», тобто ті, що належать до взаємодії з протоколом SSH. Поле «decoded_as» зберігає це значення, про що говорить опис «description». Дочірнє правило є більш специфічним, і реагує на конкретну подію. Воно має свій ID, дескриптор «if_sid», значенням якого є головна група. «pcre2» містить вираз, який потрібно знайти у журналах, події якого будуть відстежуватись. Одразу як таке повідомлення має збіг, OSSEC реагує моментально. Такий функціонал є дуже потужним і дозволяє створювати свої фільтри та налаштовувати критичність подій. Це допоможе більш точно виявляти серйозні інциденти.

Як було з'ясовано, наразі є велика кількість рішень, що задовольняють потреби адміністратору мережі у відстежуванні її стану, але для нашої конкретної системи основою слугуватиме ELK-стек. Для закриття недоліків ELK-стеку у вигляді відсутності механізму сповіщення і наявності правил порушення безпеки було обрано Elastalert та OSSEC HIDS через їх легке впровадження та потужний функціонал. Таким чином, проаналізувавши взаємодію серверів в мережі, необхідне ПЗ, клієнт-серверний підхід, була розроблена схема роботи майбутньої системи моніторингу, зображена на рис.3.

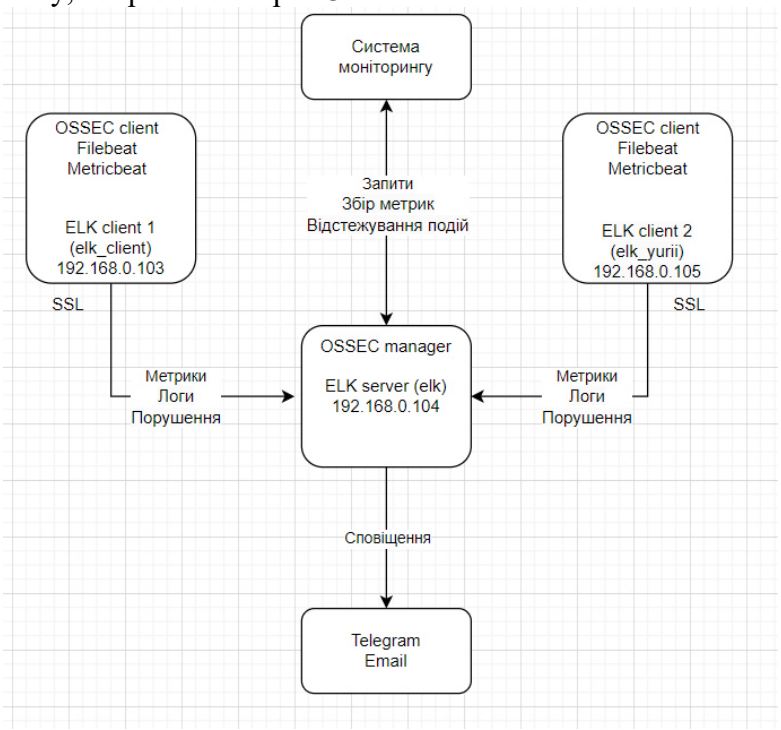

Рис.3. Схема роботи системи моніторингу в мережі

Для низького порогу входження і візуалізації у ELK-стеку бракує зручного інтерфейсу – той, що пропонує Kibana, є досить надлишковим. Тому було вирішено створити систему моніторингу, яка надавала б можливість без зайвих зусиль

відстежувати сервери, налаштовувати і додавати правила для сповіщень у популярні цільові сервіси для підвищення швидкості реакції на інциденти. Для розробки спеціалізованої системи моніторингу використовувались наступні інструменти і складові:

– мова програмування Python і модулі tkinter з Elasticsearch;
– Filebeat і Metricbeat;
– ELK-стек.

Після запуску системи моніторингу перед користувачем відкриється наступний інтерфейс, зображений на рис.4, що складається з 4 секцій:

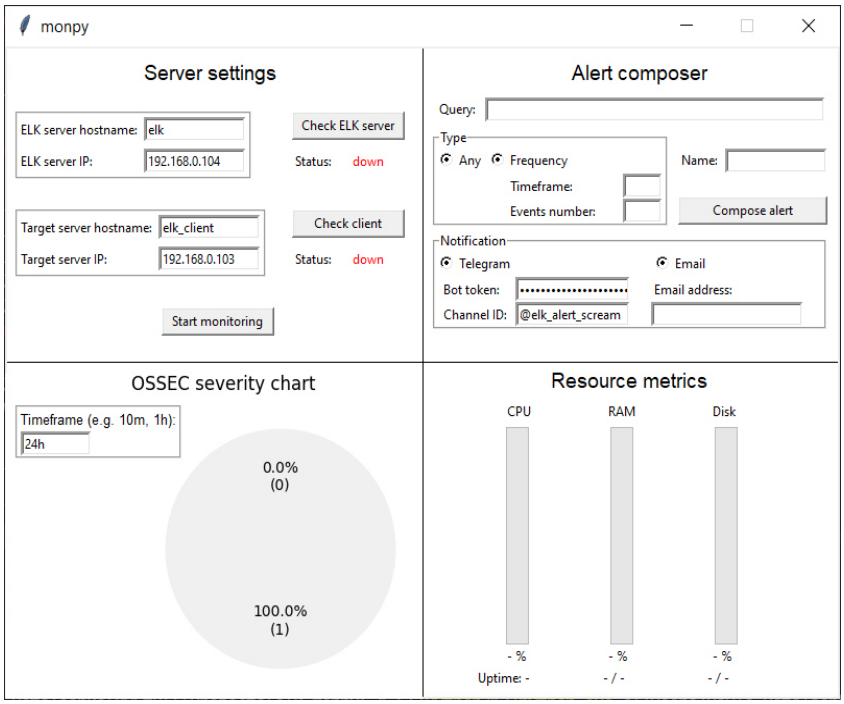

Рис.4. Початковий інтерфейс системи

Секція Server settings відповідає за внесення інформації про адресу серверу з ELK-стеком, з якого будуть збиратися події і до якого робляться усі подальші запити. Секція Resource metrics відображає поточну завантаженість цільового серверу. Додані три стовпці, які покажуть у процентному відношенні скільки ресурсів процесору було використано (стовпець CPU), об'єм оперативної пам'яті (стовпець RAM), а також зайняте місце на диску (стовпець Disk). За допомогою цих трьох показників можна досить швидко визначити стан серверу. Слідкування за ресурсами не тільки повідомить нам про необхідність розширення, наприклад, постійного накопичувача, але і повідомить про надмірну завантаженість процесору, що може свідчити про активність потенційно шкідливих процесів. Крім відображення у процентному відношенні, також надається інформація в числах, скільки усього ресурсу мається на сервері і яка кількість використовується. Для зручності додана стрічка з часом роботи машини з моменту включення (стрічка Uptime).

Секція OSSEC Severity chart являє собою діаграму критичності подій. Утиліта відслідковує і аналізує логи в системних журналах, а також стежить за процесами в цільовій системі. Кожна подія має свій «рівень небезпеки». Усього існує десять рівнів, від системного повідомлення до детектування втручання в систему, або виявлення потенційної атаки. Ранжування подій згідно з OSSEC відбувається наступним чином:

– рівні 1 - 2 вирішено опустити з причини малої цінності інформації про події, пов'язані з ними. Це можуть бути звичайні події, наприклад, перезавантаження служби операційної системи;

– низька критичність у подій рівня 3 - 4;

– середня: рівні 5 - 6;

– висока: рівні 7 - 8;

– критична: рівні 9 - 10.

В секції Alert composer ми можемо створювати файли правил сповіщень для ElastAlert. Також вона відповідає за відправлення правила на центральний сервер. На виході ми отримуємо YAML-файл, згідно нашого вводу, готовий до використання. Для створення і використання правил необхідно заповнити наступні поля:

– Query – запит, по якому будуть фільтруватись логи журналів;

– Type – панель вибору Any або Frequency. При виборі Frequency нам доступні поля Timeframe і Events number, які відповідають за проміжок часу і кількість подій, що відбулися за нього;

– Name – ім'я правила. Повинне бути унікальним для коректної роботи ElastAlert;

– Notification – панель вибору Telegram або Email. Якщо обрана позиція Telegram, необхідно надати Bot token (токен боту, що буде відправляти сповіщення) і Channel ID (ідентифікатор каналу для повідомлень). У випадку з Email, надати адресу отримувача листа зі сповіщенням.

Створимо правило для відправки сповіщення до Telegram-каналу, у якості запиту використаємо текст повідомлення про атаку перебором на SSH. Оскільки у самому правилі OSSEC вже є поля для відстеження частоти, встановимо тип Any. Введемо ім'я правила і натиснемо на кнопку Compose alert. На сервері з'явиться файл з усією відповідною конфігурацією. Приклад налаштування в секції Alert composer і результуючий YAML-файл з правилом зображено на рис.5. На рис.6 зображений приклад повідомлення на Email.

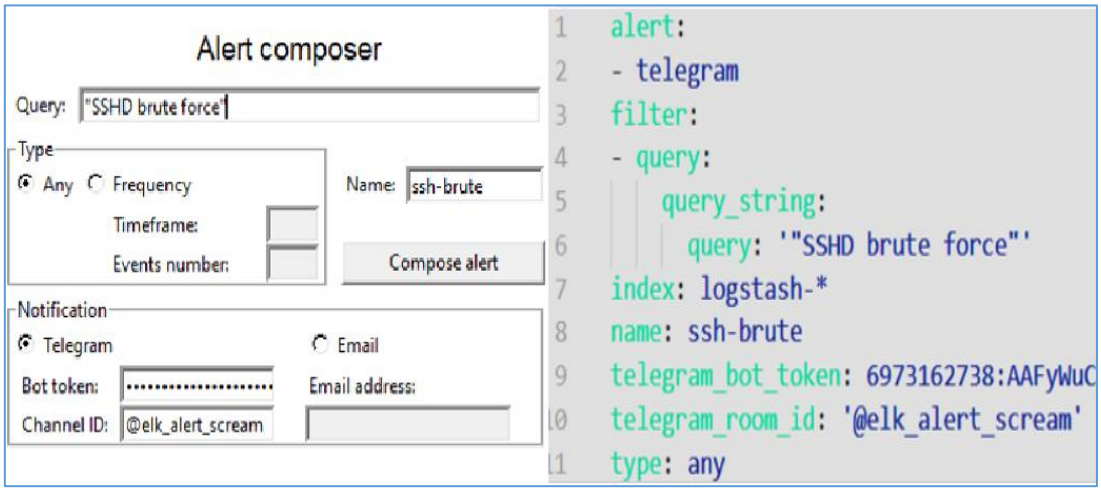

Рис.5. Налаштування Alert composer і результуючий файл з правилом

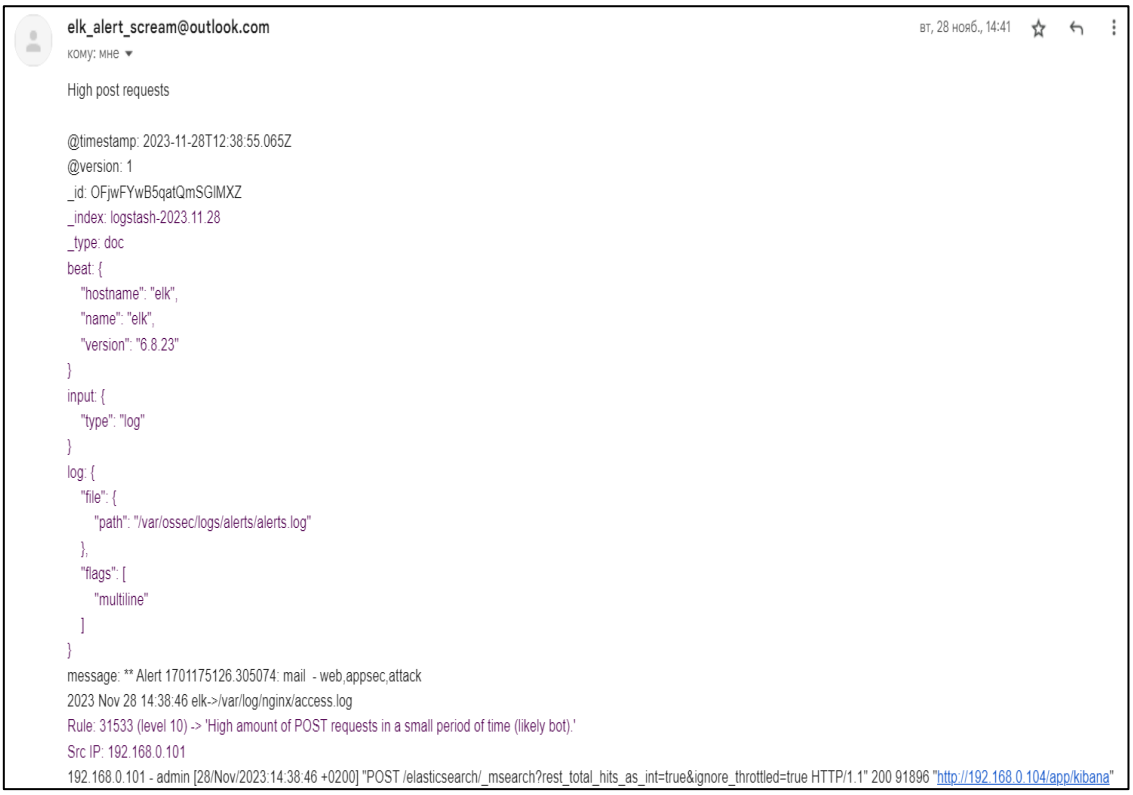

Рис.6. Приклад листа з сповіщенням на електронну пошту

На рис.7 відображена створена система моніторингу в процесі роботи, у момент відстеження доступності серверів, збору метрик і інформації про події на сервері від OSSEC. Таким чином, система дозволяє отримувати дані про стан мережі у реальному часі.

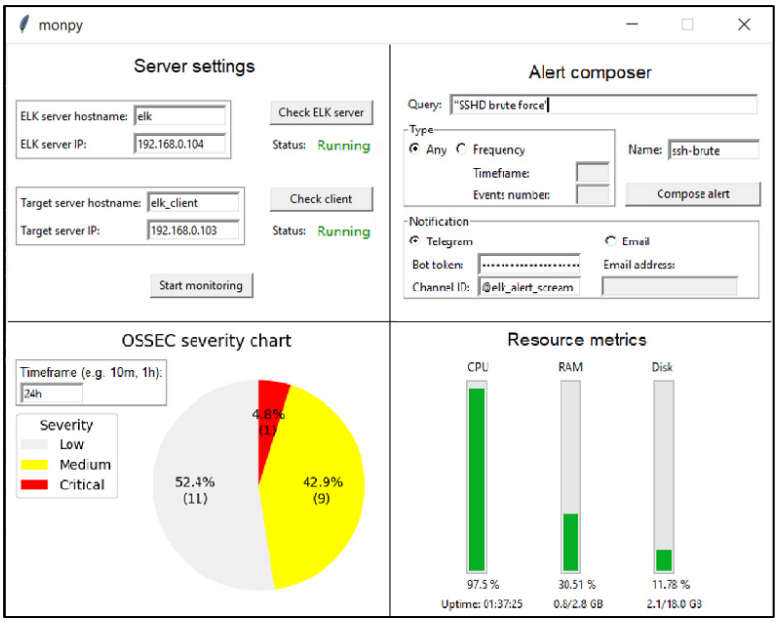

Рис.7. Система моніторингу в процесі відстежування стану мережі

Висновки. В роботі було досліджено і описано технології і способи, що використовуються для створення рішень для моніторингу, а також розроблено нову таку систему для підвищення ефективності відстежування стану безпеки комп'ютерної мережі.

Проаналізовано існуючі рішення для моніторингу мережі, здійснено аналіз таких систем, виявлено їх переваги і недоліки. У ході аналізу обраним рішенням став ELK-стек через його переваги у вартості, зручності користування, масштабованості і наявності повноцінного API.

ELK-стек не рекомендується використовувати в базовій комплектації для повноцінного моніторингу попри його потужний функціонал. Через відсутність вбудованих можливостей сповіщення адміністратора та правил кореляції ELK-стек не в змозі довершити повний набір інструментів, необхідний аналітику з безпеки. Тож він може бути доповнений іншими платформами, розширеннями і сервісами.

Наведено перелік інструментів, що були використані при розробці системи моніторингу. Покращено і доповнено функціонал ELK-стеку за допомогою побудованої на основі нього системи моніторингу. Впроваджені самостійно розроблені сповіщення з ElastAlert і правила кореляції від OSSEC для повноцінного і усебічного моніторингу стану мережі. Створено зручний інтерфейс, за допомогою якого легко і швидко можна дізнатися інформацію про стан мережі, а також створювати нові правила для сповіщень про інциденти. Подальшими кроками для покращення системи стане додавання панелей для відстежування подій безпосередньо у програмному застосунку.

Список літератури

1. Рішення Splunk. URL: https://www.splunk.com
2. LogPoint: Award winning SIEM software. URL: https://www.logpoint.com
3. Elasticsearch, Kibana, Beats & Logstash. URL: https://www.elastic.co/elastic-stack/
4. Threat Hunting Using Elastic Stack: An Evaluation. URL: https://www.researchgate.net/publication/357818741_Threat_Hunting_Using_Elastic_Stack_An_Evaluation
5. Al-Mahbashi I.Y.M., Potdar M.B., Chauhan P. Network security enhancement through effective log analysis using ELK. *International Conference on Computing Methodologies and Communicatio ICCMC.* 2017. P. 566-570. DOI:10.1109/ICCMC.2017.8282530
6. Моніторинг комп'ютерної мережі. URL: https://businessyield.com/uk/technology/network-monitoring
7. ElastAlert – Easy & Flexible Alerting With Elasticsearch. URL: https://elastalert.readthedocs.io/en/latest/elastalert.html
8. OSSEC HIDS. URL: https://www.ossec.net/docs/docs/manual/non-technical-overview.html
9. OSSEC rules composition. URL: https://www.ossec.net/docs/docs/manual/rules-decoders/create-custom.html

П.Ю. Паталашко, Н.І. Кушніренко, Н.Г. Козаченко, Н.В. Бойко

DEVELOPMENT OF INFORMATION SECURITY EVENTS MONITORING SYSTEM

P. Patalashko, N. Kushnirenko, N. Kozachenko, N. Boiko

National Odesa Polytechnic University
1, Shevchenko Ave., Odesa, 65044, Ukraine
e-mail: infsec2011@gmail.com

If a problem occurs on the network, IT professionals need to be sure that they will be notified and receive the necessary information to resolve it and prevent similar situations in the future. Notifications are an important element of network monitoring. When an abnormal behavior occurs on the network, the system must immediately notify IT professionals so that they can respond quickly. This is when this process becomes most effective. Network monitoring must include automatic notifications of failures and events that may affect performance. In the event of an error, such notifications should be sent via email, SMS, or other convenient channel. Dynamic networks require systems that can identify vulnerabilities in real time. Therefore, some network monitoring systems provide capabilities for collecting security data, such as network logs, application logs, and security messages. This data is collected and analyzed to detect anomalies, and in the event of threats, alerts can be sent to the appropriate IT professional. Thus, a network monitoring system allows you to: ensure a continuous workflow, monitor the status of the network and respond to possible problems in a timely manner, improve network efficiency, and increase network reliability and security. Given this, we can conclude that it is crucial to monitor networks and respond to problems in a timely manner for their successful operation. This requires the use of specialized monitoring systems that can quickly analyze the state of the network, identify problem areas, and notify the appropriate specialists. This work is focused on analyzing existing network monitoring systems, highlighting their pros and cons. The technologies and methodologies used to create such solutions were studied. A new monitoring system aimed at optimizing the process of tracking the status of network security was developed. The functionality of the system is enriched by sending notifications and implementing rules for tracking abnormal behavior in the network, which will help to better identify malicious actions and warn specialists in a timely manner.

Keywords: monitoring, ELK-stack, OSSEC, ElastAlert, incident, logs, metrics.