УДК 004.056.5

**РОЗРОБКА СИСТЕМИ МОНІТОРИНГУ ПОДІЙ ІНФОРМАЦІЙНОЇ БЕЗПЕКИ**

**П.Ю. Паталашко, Н.І. Кушніренко**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

Національний університет «Одеська Політехніка», просп. Шевченка, 1, Одеса, 65044, Україна; e-mail: [infsec2011@gmail.com](mailto:infsec2011@gmail.com)

У випадку, коли потенційна проблема виникає у мережі, IT-спеціалісти можуть бути впевнені, що вони будуть повідомлені та отримають необхідну інформацію для вирішення й виключення подальших проблем. Важливим елементом моніторингу мережі є сповіщення. Коли в мережі виникає проблема, система повинна негайно повідомити IT-спеціалістів, щоб вони могли швидко реагувати. Саме тоді даний процес стає найбільш ефективним. Моніторинг мережі в обов'язковому порядку повинен включати автоматичне отримання повідомлень про збої та події, що можуть вражати продуктивність. У разі помилки, такі сповіщення мають бути надіслані через електронну пошту, SMS або інше зручне місце. 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, відслідковувати 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.

**Ключові слова:** моніторинг, ELK-стек, OSSEC, ElastAlert, інцидент, сповіщення, 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. 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 популярна система Splunk бере дані і дозволяє користувачам шукати інформацію і робити запити на те, що їм потрібно. Splunk збирає, індексує та співвідносить дані в реальному часі у сховищі з можливістю пошуку, з якого можна створювати графіки, звіти, інформаційні панелі та 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.

Іншим досить популярним рішенням є ELK-стек, що приваблює відкритим кодом, який дає можливість використовувати систему будь-кому, незалежно від розміру компанії [2]. Це сприяє легкому впровадженню і гнучкості у налаштуванні системи для конкретних потреб організації. Але без додаткових компонентів в системі відсутній вбудований механізм сповіщення про потенційні загрози, його потрібно встановлювати окремо [3]. Також, система поставляється без готових правил кореляції, отже користувачу доведеться створювати власні правила для належного аналізу даних [3].

Ще одна система моніторингу, що була розглянута – LogPoint [4]. можно відзначити використання різних джерел, таких як Active Directory, для збагачення журналів, а також можливість створення складних кореляційних пошукових запитів. Крім того, у будь-якому пошуку можна налаштувати автоматичне сповіщення, яке активує електронного листа при виявленні події або перевищенні встановленого порогу [5]. 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.

Підсумовуючи аналіз різних систем моніторингу, для даної конкретної задачі було обрано ELK-стек через наступні 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.

Недоліками даної системи є: відсутність механізму сповіщення адміністратора при виникненні потенційно небезпечних подій в мережі, відсутність правил кореляції і механізму створення правил для відстежування характеру логів [14]. Беручи до уваги вищеописані недоліки, було вирішено розглянути потенційні способи подолання і збагачення функціоналу ELK-стеку за допомогою додаткових рішень з відкритим програмним кодом: Elastalert для організації відстежування і відправки сповіщень, і 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 відкритому Інтернеті не було знайдено, що підвищує цінність розробки. В процесі виконання даної роботи необхідно розв’язати наступні задачі:

* дослідити предметну область, розібрати різні системи моніторингу і SIEM-системи;
* проаналізувати та виявити найбільш важливі метрики для збору, що відображають стан мережі підприємства, а також найчастіші можливі порушення безпеки на серверах;
* провести аналіз та порівняння сучасних систем для 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 [3]. Він об’єднує процеси спостереження та аналізу мережевих компонентів таких, як роутери, комутатори та брандмауери, а ще 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, свідомість 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 інших сторін. 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. Сам таким потрібнм функціоналом володіє система, що буде лежати в основі рішення для моніторингу мережі. Як з’ясовано раніше, ЕLK-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 [15]. Elastalert дозволяє створювати правила, які матимуть увазі будь-які нелегкі ситуації і сповіщати вас 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 ваші власні алгоритми 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 cповіщеннями. 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:

– «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.

A screenshot of a computer program

Description automatically generated

**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 [16]. 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 [17]. Приклад налаштування такого правила наведений на рис.2.

A screen shot of a computer code

Description automatically generated

**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, що буде шукатись 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 задовільняють 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.

A diagram of a company

Description automatically generated

**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 спеціалізованної системи моніторингу вигористовувались 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 частин:

A screenshot of a computer

Description automatically generated

**Рис.4.** Початковий інтерфейс системи

Частина 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 потенціально шкідливих процесів. Крім відображення у процентному відношенні, а також надається інформація в дійсних 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 їх існує десять рівнів, від системного повідомлення до детектування втручання в систему, або виявлення потенційної атаки. Ранжирування 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.

A screenshot of a computer

Description automatically generated

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

A screenshot of a computer

Description automatically generated

**Рис.6.** Прикад листу з сповіщенням на електронну пошту

На рис.7 відображена створена система моніторингу в процесі роботи, відстежуючи доступність серверів, збираючи метрики і інформацію про події на сервері від OSSEC. Таким чином, система дозволяє отримувати дані про стан мережі у реальному часі.

A screenshot of a computer

Description automatically generated

**Рис.7.** Система моніторингу в процесі відстежування стану мережі

**Висновки**

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

Розібрані вимоги до системи, її функції, різні типи архітектур журналювання. Проаналізовано існуючі рішення для моніторингу мережі, здійснено порівняльний аналіз таких систем, виявлено сильні і слабкі сторони. Наведено детальне порівняння за такими критеріями, як формат логуваня, використана база даних, процес збору і менеджменту логів, можливий термін їх зберігання, можливості масштабування, швидкість розгортування і фінансові витрати. У ході аналізу обраним рішенням став ELK-стек через її переваги у коштах, зручності користування, масштабованості і 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.

Для подальшого покращення системи рекомендується додати більше панелей для відстежування подій безпосередньо у програмному застосунку.

**Список літератури**

1. Організація комп’ютерних мереж, URL:

https://ela.kpi.ua/bitstream/123456789/22890/1/Organizacia\_komputernyh\_merezh\_Konspekt\_lekciy.pdf

1. Тип атаки «Людина по середині», URL:

https://csrc.nist.gov/glossary/term/man\_in\_the\_middle\_attack

1. OWASP Session hijacking threat, URL:

https://owasp.org/www-community/attacks/Session\_hijacking\_attack

1. R. Fisli, “Secure Corporate Communications over VPN-Based WANs,” Master’s Thesis in Computer Science at the School of Computer Science and engineering, Royal Institute of Technology, Sweden (2005), P.61-83
2. Khan, M.T., DeBlasio, J., Voelker, G.M., Snoeren, A.C., Kanich, C., & Vallina-Rodriguez, N. (2018). An Empirical Analysis of the Commercial VPN Ecosystem. IMC. P.170-186
3. Donenfeld, J. (2018). WireGuard: Next Generation Kernel Network Tunnel [Ebook] (1st ed.). Retrieved from http://www.wireguard.com/papers/wireguard.pdf
4. Впровадження Wireguard до ядра Linux, URL:

https://lists.zx2c4.com/pipermail/wireguard/2020-March/005220.html

1. Оцінка і порівняння роботи основних VPN-протоколів, URL:

https://www.wireguard.com/performance/

1. Розподіл ринку в розрізі хмарних постачальників на 2022, URL:

https://www.channele2e.com/news/cloud-market-share-amazon-aws-microsoft-azure-google/

1. Огляд найпопулярніших сервісів Amazon Web Services, URL:

https://www.clickittech.com/aws/aws-services-list/

**AUTOMATION OF CONFIGURATING SECURE CONNECTION TO CORPORATE NETWORKS**

P. Patalashko, N. Kushnirenko

Odessа Polytechnic National University,

1, Shevchenko Ave., Odesa, 65044, Ukraine; e-mail: [infsec2011@gmail.com](mailto:infsec2011@gmail.com)

With the rapid development of information technology, there is a increasing need to protect information and securely transmit it over the global network. The purpose of work is the problem of information security, its systematization, identification of sources of information threats, indicators, criteria and standards. The problem of information security at the stage of transportation via the Internet has a high priority for research and the development of various solutions. Virtual private network (VPN) technology was created because of the need to connect computer networks or their segments to each other using a secure communication channel over a network with less trust. From point №1 to point №2, the data must be transmitted in such a way that it cannot be accessed by third parties and only the intended reciever has the ability to obtain and use it. Quite a real and practical problem, the relevance of which grows with each year of development of information technology, which is aimed at the technology of virtual private networks. Another important reason for using VPN technology is the rapid growth of cloud services. The problem of secure access of employees and connection of two or more networks to each other via the Internet must be solved, without increasing the level of threat due to the inevitable intersection of information through open network. The proposed software product can be recommended for use in practice with the real needs of companies that use Amazon Web Services, for automated organization of secure connection to internal networks. Also, this development can serve as a basis for the implementation of similar solutions that will satisfy the conditions of each user individually, as the code base will be same for any provider.

Keywords: virtual private networks, data transmission, organization of secure connection, tunneling protocols, cloud services, AWS.