Центральноукраїнський національний технічний університет
Механіко-технологічний факультет
Кафедра кібербезпеки та програмного забезпечення

”Допущено до захисту”
Завідувач кафедри кібербезпеки
та програмного забезпечення
д.т.н., професор
_____ Олексій СМІРНОВ
“____” _____ 2025 р.

ВИПУСКНА КВАЛІФІКАЦІЙНА РОБОТА
за другим (магістерським) рівнем вищої освіти
на тему
“Дослідження та програмна реалізація системи мережевого корпоративного центру управління інформаційною безпекою (SOC)”

Виконав здобувач вищої освіти
ІІ курсу, групи КІ-24М
ОПП «Комп’ютерна інженерія»
спеціальності 123 «Комп’ютерна інженерія»
_____ Іваненко М.О.
«____» _____ 2025 р.

Керівник проекту
кандидат технічних наук, доцент
_____ Буравченко К.О.
«____» _____ 2025 р.
Рецензент _____
_____

м. Кропивницький

АНОТАЦІЯ

Іваненко М.О. Дослідження та програмна реалізація системи мережевого корпоративного центру управління інформаційною безпекою (SOC). 123 Комп'ютерна інженерія. Центральноукраїнський національний технічний університет. Кропивницький. 2025.

В даній випускній кваліфікаційній роботі за другим (магістерським) рівнем вищої освіти розроблено програмне забезпечення, яке призначено для системи мережевого корпоративного центру управління інформаційною безпекою (SOC).

Метою розробки є дослідження та програмна реалізація системи мережевого корпоративного центру управління інформаційною безпекою (SOC).

Об'єктом дослідження є процес мережевого корпоративного центру управління інформаційною безпекою (SOC).

Предметом дослідження є методи мережевого корпоративного центру управління інформаційною безпекою (SOC).

Методи дослідження базуються на методах захисту інформації у мережі, методах математичної статистики, методах розробки програмного забезпечення.

Результат роботи – програмна реалізація системи мережевого корпоративного центру управління інформаційною безпекою (SOC).

В процесі роботи над програмною моделлю виконано аналіз існуючих апаратних та програмних засобів. В повній мірі описані всі компоненти розробленого програмного забезпечення.

Розроблено зручний інтерфейс користувача. Наведені інструкції по роботі з програмними засобами.

Програма може використовуватися на ПЕОМ з ОС Windows 10/11.

Програму розроблено в середовищі Python.

Ключові слова: комп'ютерна інженерія, мережевий корпоративний центр управління інформаційною безпекою (SOC)

ABSTRACT

Ivanenko M.O. Research and software implementation of the network corporate information security management center (SOC) system. 123 Computer Engineering. Central Ukrainian National Technical University. Kropyvnytskyi. 2025.

In this final qualification work for the second (master's) level of higher education, software has been developed, which is intended for the network corporate information security management center (SOC) system.

The purpose of the development is the research and software implementation of the network corporate information security management center (SOC) system.

The object of the research is the process of the network corporate information security management center (SOC).

The subject of the research is the methods of the network corporate information security management center (SOC).

The research methods are based on methods of information protection in the network, methods of mathematical statistics, methods of software development.

The result of the work is the software implementation of the network corporate information security management center (SOC).

In the process of working on the software model, an analysis of existing hardware and software was performed. All components of the developed software are fully described.

A convenient user interface has been developed. Instructions for working with the software are provided.

The program can be used on a PC with Windows 10/11.

The program was developed in the Python environment.

Keywords: computer engineering, network corporate information security control center (SOC)

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ І ТЕРМІНІВ

ГБЗ	–	глобальна база знань
ДФ	–	дестабілізуючі фактори
ІС	–	інформаційна система
КФ	–	керуючі фактори
ЛОМ	–	локальна обчислювальна мережа
НКК	–	нечіткі когнітивні карти
ММЕ	–	міжмережний екран
ПФ	–	проміжні фактори
СЗДДДТ	–	систем запобігання деструктивним діям дестабілізуючого трафіку
ЦФ	–	цільові фактори
SSDT	–	таблиця дескрипторів системних сервісів

ВСТУП

Актуальність теми. Центр операцій безпеки (SOC) – це централізований підрозділ, який займається питаннями безпеки на організаційному та технічному рівні. SOC оснащений командою аналітиків та інженерів з безпеки, а також сучасними технологіями виявлення та запобігання для моніторингу, аналізу та реагування на інциденти кібербезпеки.

Головною метою SOC є виявлення, оцінка, пом'якшення та звітування про кіберзагрози, забезпечення запобігання або раннього виявлення потенційних порушень безпеки та своєчасного реагування на них. Це включає постійний спостереження за IT-інфраструктурою організації, включаючи її мережі, пристрої, програми та дані, для захисту від загроз безпеці, починаючи від атак шкідливого програмного забезпечення до складного кібершпигунства.

Мета й завдання дослідження. Метою роботи є дослідження та програмна реалізація системи мережевого корпоративного центру управління інформаційною безпекою (SOC).

Для досягнення поставленої мети визначена програма дослідження, що складається з наступних завдань:

– Огляд існуючих систем мережевого корпоративного центру управління інформаційною безпекою (SOC).

– Дослідження системи мережевого корпоративного центру управління інформаційною безпекою (SOC).

– Програмна реалізація системи мережевого корпоративного центру управління інформаційною безпекою (SOC).

Об’єктом дослідження є процес мережевого корпоративного центру управління інформаційною безпекою (SOC).

Предметом дослідження є методи мережевого корпоративного центру управління інформаційною безпекою (SOC).

Методи дослідження базуються на методах захисту інформації у мережі, методах математичної статистики, методах розробки програмного забезпечення.

Наукова новизна отриманих результатів. У процесі рішення завдань, обумовлених цілями дослідження, отримані наступні результати:

– Удосконалено метод мережевого корпоративного центру управління інформаційною безпекою (SOC).

– Розроблено вітчизняний продукт мережевого корпоративного центру управління інформаційною безпекою (SOC), який має більш широкі можливості, на відміну від існуючих аналогів.

Практична цінність отриманих результатів полягає в тому, що розроблені алгоритми дозволяють успішно вирішувати задачі мережевого корпоративного центру управління інформаційною безпекою (SOC).

Достовірність наукових результатів підтверджена теоретичними викладеннями, даними комп'ютерного моделювання, коректними дослідженнями параметрів на функціонуючій обчислювальній мережі, а також відповідністю отриманих результатів окремим результатам, наведеним у науковій літературі.

Робота апробована на LVII Науково-технічній конференції здобувачів вищої освіти LV науково-технічної конференції «Наука в ЦНТУ: основні досягнення та перспективи розвитку» (2025 р.), основні положення випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти надруковані у статті збірника праць молодих науковців ЦНТУ, випуск №15.

Таким чином, виходячи з вищеперерахованого, дослідження та програмна реалізація системи мережевого корпоративного центру управління інформаційною безпекою (SOC), є актуальною задачею, яка потребує вирішення у даній випускній кваліфікаційній роботі за другим (магістерським) рівнем вищої освіти.

1 ПРИЗНАЧЕННЯ ТА ОБЛАСТЬ ВИКОРИСТАННЯ

1.1 Призначення системи

Центр мережевих операцій (NOC) служить нервовим центром для моніторингу стану, безпеки та пропускної здатності мережі організації, забезпечуючи високу доступність та продуктивність. NOC відповідає за постійний нагляд за мережею, забезпечуючи централізоване місце для усунення несправностей мережі та управління мережевими операціями.

Основні функції NOC включають постійний моніторинг мережевої та серверної інфраструктури, управління комунікаціями (електронними листами, заявками, телефонними дзвінками) щодо мережевих подій, реагування на інциденти та їх вирішення, а також контрольоване внесення змін до мережі. Завдяки проактивному виявленню та вирішенню мережевих проблем, NOC допомагає запобігти простоям та підтримувати продуктивність мережі.

1.2 Область застосування

SOC (система кібербезпеки) в першу чергу спрямована на захист від кіберзагроз та управління реагуванням на інциденти. Вона зосереджена на моніторингу, виявленні та аналізі кіберзагроз у всій IT-інфраструктурі організації.

NOC зосереджується на підтримці оптимальної продуктивності та доступності мережевої інфраструктури. Його основна увага приділяється моніторингу мережі, управлінню нею та забезпеченню безперебійної підтримки мережею програм та послуг організації.

Функції SOC зосереджені на розвідці загроз, управлінні інцидентами та аналізі подій безпеки. SOC відповідають за збір, оцінку та поширення інформації

про поточні та нові загрози. Вони аналізують сповіщення про безпеку, керують інцидентами та складають звіти про загрози, порушення та рекомендації щодо безпеки. Результати SOC включають звіти про розвідку загроз, результати реагування на інциденти та аудити відповідності.

Функції NOC зосереджені на моніторингу продуктивності мережі, вирішенні проблем та управлінні змінами. NOC постійно контролюють стан мережі, трафік та продуктивність, щоб забезпечити безперебійну роботу та ефективність. Вони усувають та вирішують проблеми з мережею, керують змінами в мережі та координують свою діяльність з постачальниками для отримання підтримки. Результати роботи NOC включають звіти про продуктивність мережі, документацію щодо вирішення інцидентів та журнали управління змінами.

Центри безпеки використовують такі інструменти, як системи управління інформацією та подіями безпеки (SIEM), системи виявлення вторгнень (IDS), рішення для виявлення та реагування на кінцеві точки (EDR) та платформи розвідки загроз. Ці інструменти дозволяють центрам безпеки агрегувати та аналізувати дані по всьому цифровому сліду організації, сприяючи своєчасному виявленню кіберзагроз та реагуванню на них.

Центри мережевого контролю (NOC) використовують інструменти моніторингу мережі, аналізатори продуктивності мережі та бази даних керування конфігурацією (CMDB) для забезпечення справності та ефективності мережі. Ці інструменти дозволяють NOC контролювати мережевий трафік, виявляти вузькі місця, керувати конфігураціями мережі та автоматизувати реагування на поширені проблеми мережі.

Фахівці SOC зазвичай володіють навичками кібербезпеки, аналізу загроз, реагування на інциденти та знаннями нормативних актів щодо дотримання вимог. Вони повинні вміти використовувати інструменти управління інформацією та подіями безпеки (SIEM), розуміти найновіші загрози кібербезпеці та впроваджувати заходи безпеки.

Персонал NOC потребує глибоких знань у сфері мережевого адміністрування, системної інженерії, інструментів мережевого моніторингу та методів усунення несправностей. Їм необхідно розуміти мережеві протоколи, проектування інфраструктури та стратегії оптимізації продуктивності.

Кар'єрні шляхи SOC зазвичай починаються з початкових посад, таких як аналітик безпеки, і продовжуються до таких ролей, як менеджер SOC або спеціаліст з реагування на інциденти. Розширені посади можуть включати аналітика розвідки загроз або архітектора безпеки, що зосереджуються на стратегічному плануванні безпеки та розширеному аналізі загроз. Фахівці SOC можуть додатково спеціалізуватися в таких галузях, як судово-медичний аналіз або посади у сфері дотримання вимог та аудиту.

У мережевому центрі (NOC) кар'єрний ріст часто починається з посади мережевого техніка або мережевого аналітика, а потім ступінь – мережевого інженера або менеджера NOC. З досвідом роботи фахівці можуть просуватися до таких посад, як мережевий архітектор або системний інженер, що спеціалізуються на проектуванні, впровадженні та оптимізації мереж. Спеціалізації включають хмарні мережі та автоматизацію.

SOC та NOC: ключові виклики

Існує кілька проблем, що впливають як на команди SOC, так і на команди NOC.

Попередження про втому

Команди SOC часто стикаються з проблемою виснаження від сповіщень через величезну кількість сповіщень безпеки, що генеруються інструментами моніторингу. Розрізнення хибнопозитивних результатів та справжніх загроз може бути складним завданням, що призводить до пропущених або ігнорованих сповіщень.

У контексті NOC, «втома» сповіщень може виникати, коли інструменти моніторингу генерують надмірну кількість некритичних сповіщень, що потенційно призводить до того, що серйозні проблеми мережі не будуть

проігноровані. Впровадження кращих механізмів фільтрації та стратегій пріоритезації є важливим для ефективного управління обсягами сповіщень.

Аналітика спостереження та безпеки

Для команд SOC складність та обсяг даних, які вони повинні аналізувати, можуть бути приголомшливими. SOC потребують розширених інструментів спостереження та аналітики, щоб забезпечити глибоке розуміння поведінки мережі, активності користувачів та потенційних загроз безпеці. Ці інструменти повинні просіювати величезні обсяги даних, виявляючи аномалії та закономірності, які можуть свідчити про порушення безпеки.

Спостережуваність у мережевому контексті передбачає розуміння стану мережі та її компонентів у режимі реального часу, що є критично важливим для забезпечення високої доступності та продуктивності. Досягнення такого рівня спостережуваності вимагає комплексних інструментів моніторингу, які можуть аналізувати потоки трафіку, стан пристроїв та зміни топології мережі.

Розчинення периметра мережі

Розмивання мережевого периметра, з впровадженням хмарних сервісів, периферійних обчислень та політик «Принеси свій власний пристрій» (BYOD), створює проблеми як для безпеки, так і для управління мережею.

Центри мережевого контролю (SOC) повинні розширити свої можливості моніторингу та управління безпекою за межі традиційних мережевих меж, забезпечуючи безпечне розгортання хмарних технологій, моніторинг периферійних пристроїв та управління політиками безпеки для персональних пристроїв на робочому місці. Центри мережевого контролю (NOC) стикаються з проблемою підтримки продуктивності та надійності мережі в розширеному та децентралізованому середовищі.

SOC проти NOC: що підходить для моєї організації?

Вирішення питання про те, чи потрібен вашій організації Центр операцій безпеки (SOC), Центр операцій мережі (NOC) чи обидва, залежить від кількох факторів, включаючи розмір вашої організації, складність вашої IT-

інфраструктури та конкретні потреби безпеки та експлуатації. Ось ключові міркування, які слід враховувати при прийнятті рішення.

Розуміння основних потреб:

– Якщо вашою основною турботою є кібербезпека та захист ваших активів від кіберзагроз, SOC є надзвичайно важливим. Організації з конфіденційними даними, вимогами до відповідності та високим ризиком кібератак отримають вигоду від спеціалізованої уваги SOC до безпеки.

– Якщо забезпечення доступності, продуктивності та надійності вашої IT-інфраструктури є вашим пріоритетом, NOC відіграє вирішальну роль. Це особливо важливо для організацій, які значною мірою залежать від своєї мережі для щоденної роботи та надання послуг.

Бюджет та ресурси:

– Створення та функціонування SOC або NOC вимагає значних інвестицій у технології, інструменти та кваліфікований персонал. Оцініть свій бюджет і подумайте, який центр запропонує найбільшу цінність, виходячи з конкретних ризиків та операційних вимог вашої організації.

– Малі та середні підприємства (МСП) з обмеженими ресурсами можуть розглянути можливість аутсорсингу послуг SOC або NOC або впровадження гібридної моделі, яка поєднує внутрішні та зовнішні можливості.

Відповідність нормативним вимогам та галузевим стандартам:

– У деяких галузях діють суворі регуляторні вимоги, які вимагають наявності SOC для дотримання стандартів, пов’язаних із захистом даних та конфіденційністю (таких як GDPR, HIPAA або PCI-DSS). Визначте, чи підпадає ваша організація під такі правила.

– Навіть якщо це не вимагається законом, дотримання найкращих практик в управлінні мережею та кібербезпеці може суттєво покращити репутацію вашої організації та довіру клієнтів.

Інтеграція з існуючою IT-інфраструктурою:

– Подумайте, як SOC або NOC інтегруватиметься з вашими поточними IT-операціями. Зосередження SOC на безпеці може вимагати змін у архітектурі вашої мережі для підтримки розширених можливостей моніторингу та реагування на інциденти.

– Акцент NOC на продуктивності мережі може вимагати оновлення або змін у вашій мережевій інфраструктурі для підтримки розширених інструментів моніторингу та управління.

Майбутнє зростання та масштабованість:

– Передбачте майбутнє зростання вашої організації та масштабованість вашої IT-інфраструктури. Центр безпеки (SOC) може допомогти керувати підвищеними ризиками безпеки, пов'язаними з розширенням, тоді як NOC може забезпечити ефективне масштабування вашої мережевої інфраструктури.

– Для організацій, що швидко розвиваються, створення як SOC, так і NOC може бути вигідним у довгостроковій перспективі, забезпечуючи комплексне охоплення як безпеки, так і продуктивності мережі.

SIEM у майбутньому SOC

Центр операцій безпеки переживає захопливу трансформацію. Він інтегрується з операційними відділами та відділами розробки, а також отримує потужні нові технології, зберігаючи при цьому свою традиційну командну структуру та ролі щодо виявлення та реагування на критичні інциденти безпеки.

– Вплив SIEM наступного покоління на SOC може бути значним. Він може:

– Зменште втому від тривог за допомогою аналітики поведінки користувачів та об'єктів (UEBA), яка виходить за рамки правил кореляції, допомагає зменшити кількість хибнопозитивних результатів та виявляти приховані загрози.

– Покращте показник MTTD, допомагаючи аналітикам швидше виявляти інциденти та збирати всі відповідні дані.

– Покращте MTTR (середній час очікування) шляхом інтеграції з системами безпеки та використання технології оркестрації, автоматизації та реагування на безпеку (SOAR).

– Забезпечте пошук загроз, надаючи аналітикам швидкий і легкий доступ і потужне дослідження необмежених обсягів даних безпеки.

Таким чином, виходячи з вищеперерахованого, дослідження та програмна реалізація системи мережевого корпоративного центру управління інформаційною безпекою (SOC), є актуальною задачею, яка потребує вирішення у даній випускній кваліфікаційній роботі за другим (магістерським) рівнем вищої освіти.

2 ПЕРЕГЛЯД АНАЛОГІЧНИХ ІСНУЮЧИХ СИСТЕМ

2.1 Огляд існуючих систем, технологій, архітектур, програмних рішень за профілем теми випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти

У даному розділі зрівняли популярні SIEM-рішення для створення SOC Ринок SIEM-систем зложився досить давно. На ньому представлені технічні рішення різних виробників. Всі вони відрізняються по архітектурі, можливостям масштабування, повноті функціонала, спектру розв'язуваних прикладних ІБ-завдань. Більша частина цих рішень пішла по шляху розвитку систем збору й централізованого зберігання подій з різних пристроїв (Log Management). Згодом з'явилися кореляція подій і виявлення інцидентів. На даний момент кожний виробник намагається доповнити функціонал SIEM допоміжними функціями, такими як керування уразливостями, ризиками, пошук аномалій у мережних протоколах, формування списків заражених IP-адрес і т.д.

У даному розділі вирішили зрівняти функціонал ряду SIEM-систем, щоб дати узагальнене подання про їхні можливості. При підготовці списку порівнюваних рішень у тому числі опиралися на звіти міжнародних аналітичних агентств, у першу чергу – компанії Gartner, а також на популярність рішень на українському ринку. (Для аналізу «здатності реалізації» аналітики Gartner беруть до уваги сім критеріїв, пов'язаних з досвідом клієнта при використанні продукту або сервісу. Ця оцінка включає простоту установки, використання, адміністрування й масштабування, стабільність роботи й наступну сервісну підтримку. Для збору інформації аналітики проводять інтерв'ю зі споживачами послуг постачальників, а також збирають відкликання клієнтів Gartner, які користувалися SIEM-системами. По шкалі «повнота бачення» оцінюється здатність організації розуміти потреби клієнта й втілювати це у своїх продуктах і сервісах.

Основними функціями SIEM є кореляція подій і виявлення інцидентів інформаційної безпеки. Через малу поширеність на ринку України таких продуктів, як Splunk і LogRhythm, вирішили не розглядати ці платформи в даному розділі.

На нашу думку, серед вітчизняних рішень найбільш перспективної є платформа Positive Technologies. Незважаючи на те, що платформа заснована на Open Source, компанія здатна розвивати своє рішення самостійно. Таким чином, у список порівнюваних рішень потрапили:

- HP ArcSight.
- IBM QRadar.
- Intel Security.
- McAfee ESM.
- RSA Security Analytics.
- Positive Technologies MaxPatrol SIEM.

Тому що всі компанії різні, очікування від SIEM-систем у них різні, відповідно, відрізняються й пріоритети в частині функціональних можливостей. Тому вводимо вагову модель для визначення оптимальної платформи для компаній, де значення 5 – це максимально важливий критерій для організації, а 1 – відповідно, навпаки.

1. Підтримка джерел подій.
2. Збір подій.
3. Кореляція.
4. Пошук даних і аналітика.
5. Візуалізація й звітність.
6. Оповіщення й пріоритизація.
7. Загальні налаштування й передвстановлений функціонал.
8. Масштабованість, відказостійкість і зберігання.
9. Моніторинг компонентів системи й внутрішній аудит.
10. Зручність використання.

11. Наявність сертифікатів відповідності.

12. Додаткові модулі системи.

Для оцінки ступеня виконання критеріїв порівняння ввели наступні види оцінок:

0 – критерій не виконується;

1 – критерій виконується частково;

2 – критерій повністю виконується.

Далі розбираємо кожний блок

Підтримка джерел подій

PT MaxPatrol SIEM

Кількість підтримуваних джерел постійно росте. Відомі підтримувані джерела – Syslog, Windows Event Log, Windows File log, Windows WMI log, NetFlow, ODBC Log, Checkpoint LEA, SNMP Traps, SSH File Log, Telnet File Log. Max Patrol SIEM – новий продукт на ринку SIEM, він не дотягає до гігантів індустрії по кількості підтримуваних систем, але дуже динамічно розвивається. Українських корінь вендору можуть дати продукту підтримку вітчизняних джерел подій, відсутніх у всіх його західних конкурентах. Для розробки правил нормалізації використовується SDK, власна мова програмування дозволяє гнучко нормалізувати практично будь-яка подія. Автознаходження джерел подій у системі присутня, база джерел періодично поповнюється.

IBM QRadar

Платформою підтримується більше 300 стандартних джерел подій. Повноцінна категоризація визначена для більшої частини основних подій аудита, але досить часто зустрічаються й події без категорії. Якість парсінгу обумовлена й використовуваною схемою зберігання подій, що включає невелику кількість найбільш критичних полів. Відновлення коннекторів/парсерів подій випускаються вендором у міру виходу виправлень і доповнень. Є присутнім автооновлення парсерів подій. Для підключення нестандартних джерел можуть застосовуватися часто використовувані транспорти. Розробка власних парсерів відбувається по

більшій частині в основному інтерфейсі продукту, для опису подій використовується regex.

HP ArcSight

Платформою підтримується більше 300 стандартних джерел подій. Всі події категоризовані, і їхні імена визначені. Крім цього, коннектори багатьох систем містять у собі кілька варіантів парсерів, що дозволяє вибрати найбільш підходящий під конкретні цілі алгоритм обробки аудита. Відновлення коннекторів/парсерів подій випускаються вендором у міру виходу виправлень і доповнень – 2–3 рази у квартал. Можливість автооновлення парсерів подій відсутній. Вендор завіряє, що це зроблено навмисно, оскільки автооновлення у виробничому середовищі може привести до зміни кореляційної логіки. У багатьох замовників на кореляційну логіку зав'язані SLA, ескалації, документообіг – тут важливо, щоб всі зміни SIEM-системи проходили контрольованим образом. Крім того, наявність підключення SIEM-системи до мережі Інтернет створює певні ризики. Для підключення нестандартних джерел можуть застосовуватися часто використовувані транспорти. Механізм розробки коннекторів, реалізований в ArcSight, є одним із самих потужних і гнучких. Він дозволяє не тільки розкласти подію по певних полях, але й за допомогою безлічі убудованих функцій змінювати ці значення, а також реалізовувати логічні операції, ґрунтуючись на значеннях певних токенів. Коннектор являє собою текстовий файл певного формату, для опису подій використовується regex. Для розробки також існує кілька графічних утиліт.

RSA Security Analytics

Підтримується велика кількість різнорідних систем. Більше 250 підтримуваних стандартних джерел подій. Для парсінгу в платформу закладена багаторівнева модель обробки події. Є проблеми з упізнанням систем по подіях. При розборі подій виникають проблеми з кирилицею. Немає регулярності у виході відновлень коннекторів/парсерів. Відновлення виходять залежно від популярності нестандартного джерела, що підключається. Підтримується автооновлення парсерів подій. Для розробки коннекторів використовується модуль минулого

SIEM від RSA – enVision. Парсер являє собою текстовий файл XML-формату. Як приймач RSA enVision, RSA SA використовує багато хто його парсери. Платформою підтримуються наступні види транспортів: AWS, Checkpoint, File Collection, Netflow Collection, ODBC, SDEE, SNMP, VMware, Windows, Legacy Windows і NetApp. Є присутнім автознаходження джерел подій.

McAfee ESM

Рішення підтримує велика кількість різнорідних джерел подій (більше 400 систем: WMI, Syslog, SCP, FTP, HTTP(S), ODBC/MSSQL, OPsec, CEF, MEF). Обробка подій виконується коректно. Але відсутній механізм траблшутинга парсінгу подій. Наприклад, для аудита СУБД дуже складно розібратися, чому може не здійснюватися парсинг подій. Відновлення коннекторів виходять регулярно й доступні відразу для всього модельного ряду. Підтримується автознаходження джерел подій. Підтримується створення власних парсерів подій. Розробка відбувається в основному інтерфейсі продукту, для опису подій використовується regex.

Збір подій

PT MaxPatrol SIEM

Присутні механізми нормалізації, агрегації й фільтрації подій. Нормалізація виробляється тільки для певних типів подій. Підтримується можливість збору, зберігання й роботи з raw-подій. Відсутній маскування даних при зборі/відображенні в консолі. Платформою підтримується моніторинг мережного трафіку аж до 7-го рівня моделі OSI за допомогою додаткового модуля MaxPatrol X Network Traffic.

IBM QRadar

Схема нормалізації має 19 полів. Зустрічається багато подій, які погано нормалізуються. Агрегація присутня, але вона не налаштовується. Також підтримується фільтрація, але відфільтровані події враховуються в ліцензійному обмеженні. Забезпечуються маскування даних і моніторинг мережного трафіку аж до 7-го рівня моделі OSI.

HP ArcSight

Схема нормалізації має більше 200 полів. Події добре нормалізуються. Є присутнім можливість гнучкого налаштування параметрів агрегації. Підтримується маскування даних. Моніторинг NetFlow до 7-го рівня моделі OSI здійснюється шляхом інтеграції HP ArcSight і HP Tipping Point. Включення передачі подій з Tipping Point в ArcSight підтримується рішенням за замовчуванням і здійснюється однією дією в інтерфейсі керування.

RSA Security Analytics

Підтримуються нормалізація, агрегація й фільтрація подій. Для контролю цілісності даних потрібне використання додаткового модуля Archiver. Робота з raw-подій набагато повільніше в порівнянні з конкурентами. Можливість зберігання даних протягом різного періоду часу, їхнього поділу на фізичному й логічному рівні також вимагає використання додаткового модуля Archiver. Відсутнє маскування даних. Моніторинг мережного трафіку аж до 7-го рівня моделі OSI здійснюється за допомогою модуля Packet Decoder.

McAfee ESM

Процес нормалізації приводить всі події у формат MEF (McAfee Event Format). Здійснюється категоризація подій. Агрегація присутня. Є обмеження по агрегації – максимум 3 значення, по яких можна неї виконати. Параметри агрегації можна перевизначити. Розбір мережного трафіку на рівні застосунків здійснюється шляхом інтеграції з рішенням IPS від McAfee.

Кореляція

PT MaxPatrol SIEM

Реалізовано механізми real-time кореляції подій. Відсутні механізми для проведення поведінкового аналізу. Підтримується збагачення даних у межах платформи MaxPatrol X. Проведення кореляції історичних даних планується реалізувати в наступних релізах.

IBM QRadar

Реалізовано механізми real-time кореляції подій, їсти можливість провести поведінковий аналіз, здійснюється збагачення даних з інших систем, підтримується кореляція історичних даних.

HP ArcSight

У продукті реалізований один із гнучко кореляційних механізмів, що найбільше набудовуються. Можливе збагачення даних, що збираються, зі сторонніх систем на рівнях збору (у коннекторі) і обробки (на менеджері). Історична кореляція обмежена можливістю ручної перевірки правила на історичних даних певного тимчасового інтервалу. Генерація кореляційних подій при цьому не відбувається.

RSA Security Analytics

У ядрі системи закладений досить слабкий кореляційний функціонал, для повноцінної кореляції можливе використання додаткового модуля ESA. Функціонал історичної кореляції в платформі відсутній.

McAfee ESM

Реалізовано механізми real-time кореляції подій. При проведенні поведінкового аналізу є можливість переглянути два накладених графіки – статистику по подіях за попередній період і поточну статистику (онлайн). Для роботи з історичними даними необхідно використовувати компонент McAfee Advanced Correlation Engine (ACE). Цей модуль може виконувати й історичну кореляцію, але в один момент часу він може працювати тільки в одному з режимів (real-time або історичної кореляції).

Пошук даних і аналітика

PT MaxPatrol SIEM

Підтримується пошук по подіях. Є присутнім можливість угруповання подій, Drilldown по полях і застосування активного впливу. Підтримується механізм Google Like Search. Швидкість роботи інтерфейсу в бойовій системі середня. Як засоби діагностики компонентів системи використовуються файли

журналів компонентів. Події аудита роботи самої системи не попадають в основний потік подій SIEM. Є можливість активного впливу засобами самої платформи (сканування), а також виконання скриптів.

IBM QRadar

Підтримуються пошук по подіях і угруповання подій. Drilldown по полях здійснюється в кілька дій. Є присутнім механізм Google Like Search. Швидкість роботи інтерфейсу висока з урахуванням виконання апаратних вимог. Події аудита роботи самої системи є частиною основного потоку подій SIEM. Існують передналаштовані правила, що реагують на критичні події діагностики внутрішніх компонентів. Є присутнім можливість створення своїх правил. Підтримується можливість виконання скриптів.

HP ArcSight

Підтримуються пошук по подіях і угруповання подій. Drilldown по полях здійснюється в кілька дій, є можливість створення декількох різних Drilldown для одного Dashboard. Механізм Google Like Search реалізований тільки в web-інтерфейсах продукту. Швидкість роботи інтерфейсу висока з урахуванням виконання апаратних вимог. Події аудита роботи самої системи є частиною основного потоку подій SIEM. Існують передналаштовані правила, що реагують на критичні події діагностики внутрішніх компонентів. Є присутнім можливість створення своїх правил. Також є можливість виконання команд на деяких продуктах HP (а також деяких інших вендорів), кастомних скриптів (на менеджері або коннекторах).

RSA Security Analytics

Підтримуються пошук по подіях і угруповання подій. Drilldown по полях здійснюється в кілька дій. Механізм Google Like Search доступний при використанні додаткового модуля Warehouse. Швидкість роботи інтерфейсу висока з урахуванням виконання апаратних вимог. Існують убудовані засоби діагностики компонентів системи.

McAfee ESM

На практиці зустрічаються ситуації, коли швидкість роботи інтерфейсу падає: наприклад, при вибірці по невеликому проміжку даних, які були зафіксовані більше 3 місяців назад. Створення власного dashboard і пошук по ньому приводять до затримок. Є припущення, що індексується тільки строго певні значення (SRC IP, DST IP і т.п.). Відповідно, пошук за іншими значеннями забирає тривалий час. Присутні засобу діагностики компонентів системи, але найчастіше потрібно звертатися до вендору, тому що штатна діагностика звичайно говорить про те, що події не надходять, ресивер не доступний і т.п.

Візуалізація й звітність

PT MaxPatrol SIEM

Доступні гістограми й графіки, а також таблиці й звіти. Інтерфейс русифікований. Є убудовані звіти, формат, полючи, проміжки часу, але для їх кастомизації необхідно використовувати SDK.

IBM QRadar

За замовчуванням доступні 9 типів графічного подання. Існують деякі обмеження в кастомизації графічних панелей. Інтерфейс русифікований. Звіти можуть бути експортовані у файли наступних форматів: MS Excel, RTF, PDF, XML, HTML.

HP ArcSight

Доступні більше 20 типів графічного подання. Як поля графічного подання використовуються Data Monitor, Dashboard, Query Viewer. Русифікований інтерфейс рішення доступний з лютого 2015 року. Звіти можуть бути експортовані в наступні формати: MS Excel, RTF, PDF, CSV, HTML.

RSA Security Analytics

Доступні більше 5 типів графічного подання. Як поля графічного подання використовуються Dashboard, System Stats. Російський інтерфейс відсутній. Звіти можуть бути експортовані в наступні формати: MS Excel, RTF, PDF, CSV, HTML.

McAfee ESM

Доступні наступні типи графічного подання: табличне, pie chart, bar chart, графіки, граф комунікацій на основі аналізу NetFlow. Російський інтерфейс відсутній. Звіти можуть бути експортовані в наступні формати: PDF, CSV, HTML. Є присутнім можливість запуску звітів за більші проміжки часу.

Оповіщення й пріоритизація

PT MaxPatrol SIEM

Відсутнє застосування активного впливу й реакції на оповіщення. Неможливо провести кастомизацію параметрів оповіщення. Як можливі способи оповіщення доступний тільки SMTP. Низька гнучкість налаштування. Кастомизація пріоритизації подій доступна при використанні SDK. Відсутня можливість об'єднання подій по параметрах інциденту.

IBM QRadar

Застосування активного впливу й реакції на оповіщення доступно тільки для ряду продуктів IBM. При кастомизації параметрів оповіщення неможливо використовувати змінні. Доступні наступні способи оповіщення: E-mail, Syslog, Console. Відсутня кастомизація пріоритизації подій.

HP ArcSight

Доступне застосування активного впливу, реакції на оповіщення (нативні процедури для деяких продуктів HP, в інших продуктах це можливо за допомогою виконання скриптів) і кастомизації параметрів оповіщення. Можливі способи оповіщення: E-mail, SMS, консоль, виконання команд в ОС хоста менеджера або коннектора. Є можливість кастомизації й пріоритизації подій.

RSA Security Analytics

Доступне застосування активного впливу й реакції на оповіщення, а також кастомизації параметрів оповіщення. Можливі способи оповіщення: SMTP, SNMP. Є можливість кастомизації пріоритизації подій.

McAfee ESM

Доступні способи оповіщення починаючи з оповіщення по електронній пошті й запуску зовнішньої команди (скрипту) на зазначеному пристрої й закінчуючи повноцінною інтеграцією на рівні API із продуктами McAfee Network Security Platform (IPS/IDS рішення), всіма агентськими продуктами McAfee (від антивірусу до захисту БД) на рівні запуску спеціалізованих команд і перепризначення політик на агенті, а також зі сканером уразливості McAfee Vulnerability Manager (запуск сканування прямо з консолі SIEM).

Загальні налаштування й передвстановлений функціонал

PT MaxPatrol SIEM

Відсутня інтеграція з LDAP і AD для забезпечення автентифікації, але даний функціонал вендор обіцяє реалізувати в 12-м релізі своєї платформи. Для розмежування доступу між користувачами доступна рольова модель. Присутні убудовані кореляційні правила, графічні панелі й звіти. Реалізовано базового функціонала керування інцидентами**.**

IBM QRadar

Підтримується автентифікація користувачів у різних LDAP. Існує убудований функціонал Workflow. Підтримується інтеграція зі сторонніми Workflow-системами (обмежена по підтримуваних операціях). Убудовано велику кількість передвстановлених кореляційних ресурсів, звітів і графічних панелей.

HP ArcSight

Підтримується автентифікація користувачів у різних LDAP. Реалізовано убудованого функціонала Workflow із гнучкими можливостями кастомизації. Підтримується інтеграція зі сторонніми Workflow-системами. Убудована велика кількість передвстановлених кореляційних ресурсів, звітів і графічних панелей.

RSA Security Analytics

Підтримується автентифікація користувачів у різних LDAP. Існує убудований функціонал Workflow, підтримується інтеграція зі сторонніми системами. Убудовано велику кількість передвстановлених кореляційних ресурсів,

звітів і графічних панелей McAfee ESM. Підтримується автентифікація користувачів у різних LDAP. Здійснюється інтеграція з Remedy на рівні підключення по API;

– будь-яка зовнішня система рівня Service Desk інтегрується на рівні шаблонових повідомлень SMTP для автоматичного закладу інцидентів. Убудовано велику кількість передвстановлених кореляційних ресурсів, звітів і графічних панелей.

Масштабування, відказостійкість і зберігання

PT MaxPatrol SIEM

Обмеженням по кількості оброблюваних подій у секунду є поріг в 30 000 (максимальна інсталяція, за інформацією від вендору). Відсутня можливість резервування компонентів системи й реалізації відказостійкої конфігурації. Зберігання подій здійснюється у вихідному й нормалізованому виді. Здійснюється стиск даних до 30%. Для зберігання подій використовується MongoDB. Існує можливість інтеграції із зовнішніми масивами для зберігання архівних даних.

IBM QRadar

Обмеженням по кількості оброблюваних подій у секунду є поріг в 1 200 000 (максимальна інсталяція, за інформацією від вендору). Є можливість реалізації конфігурації High Availability. Здійснюється резервування всіх компонентів системи, аж до ядра. Стиск даних при зберіганні відбувається в співвідношенні 1 до 10. Для зберігання подій використовуються Ariel.

HP ArcSight

Обмеженням по кількості оброблюваних подій у секунду є поріг в 1 500 000 (максимальна інсталяція, за інформацією від вендору). Є можливість реалізації конфігурації High Availability. Здійснюється резервування всіх компонентів системи, аж до ядра. Стиск даних при зберіганні відбувається в співвідношенні 1 до 10. Для зберігання подій використовуються CORR-Engine.

RSA Security Analytics

Обмеженням по кількості оброблюваних подій у секунду є поріг в 20 000 (максимальна інсталяція, за інформацією від вендору). Є можливість реалізації конфігурації High Availability. Здійснюється резервування всіх компонентів системи, аж до ядра. Стиск даних при зберіганні відбувається в співвідношенні 1 до 10. Для зберігання подій використовується RAW/Meta і у випадку з Warehouse – Hadoop.

McAfee ESM

Обмеженням по кількості оброблюваних подій у секунду є поріг в 300 000 (максимальна інсталяція, за інформацією від вендору). Можливість збільшення потужності ядра й компонентів системи доступна для віртуальних комплексів. У випадку ПАК потрібне придбання нової платформи. Є можливість реалізації конфігурації High Availability. Здійснюється резервування всіх компонентів системи, аж до ядра. Для зберігання подій використовується власна внутрішня база даних.

Як уже сказали спочатку, цей розділ містить у собі інформацію, сформовану на основі багаторічного досвіду роботи з SIEM-рішеннями. Просимо звернути увагу на те, що використовувані для порівняння критерії – це досвід фахівців. Для кожної окремої організації вони можуть мінятися, так само як і значення для кожного із критеріїв. Досвід проведених пілотів і проектів по впровадженню показує, що кожний замовник формує свій унікальний перелік критеріїв їхньої успішності, і від компанії до компанії, від проекту до проекту вони збігаються лише частково. Проте, у цьому порівнянні постаралися узагальнити критерії й сформувати перелік, що підходить практично будь-якій організації. Цією статтею хочемо призвати своїх читачів до зваженого підходу при виборі подібних рішень і надати приклад, на який можна було б опиратися. розуміємо, що скільки людей, стільки й думок, тому рекомендуємо якщо буде потреба спробувати змінити ваги й додати в даний перелік критеріїв свої, специфічні параметри.

HP ArcSight раніше інших продуктів огляду з'явився на ринку SIEM-рішень України, тому встиг завоювати чимало шанувальників і супротивників. Продукт HP підтримує широкий перелік різноманітних джерел подій, виконуючи нормалізацію на дуже високому рівні. Він має найбільш широкі можливості тонкого налагодження, кастомизації й дійсно потужний кореляційний функціонал. На ArcSight уже побудована безліч SOC у найбільших телекомунікаційних, добувних і фінансових холдингах. Платою за міць і гнучкість є складність первинного вивчення продукту, його висока вартість і невелика кількість кваліфікованих українських фахівців, що вміють працювати з рішенням. IBM QRadar виявився на українському ринку небагато пізніше й на той момент явно відставав від ArcSight по кореляційному функціоналі, але зате мав набагато більше простий і зрозумілий інтерфейс. За цей час його кореляційні можливості значно зросли, але поки не дотягають до можливостей ArcSight. У продукту з'явилося багато нового, передового функціонала, завдяки якому він виявився на лідируючих позиціях звіту Гартнера.

QRadar має відмінні можливості горизонтальної масштабованості, є присутнім функціонал аналізу мережних потоків, а також можливість інтеграції з безліччю додаткових модулів від IBM. Можливості тонкого налагодження й кастомизації обмежені.

Рішення McAfee ESM у першу чергу розраховано на поточних замовників вендору. Завдяки єдиному API реалізована найбільш тісна інтеграція з усією продуктовою лінійкою виробника, що дозволяє організувати 2-сторонній зв'язок практично з усією інфраструктурою безпеки (побудованої на McAfee). У продукті реалізований досить зручний і зрозумілий web-інтерфейс, що дозволяє швидко змінювати подання даних при розслідуванні. Кореляційний функціонал, реалізований у компоненті ERC, не відрізняється високою гнучкістю налаштування. Існує також ACE – окремий пристрій, що підтримує як real-time, так і історичну кореляцію й що реалізує risk-based кореляцію. Можливості тонкого налагодження й кастомизації обмежені.

Продукт RSA Security Analytics з'явився на українському ринку SIEM порівняно недавно, він увібрав у себе досвід минулого SIEM вендору (EnVision) і можливості придбаного ними NetWitness. Основна концепція продукту заснована на більше тісному включенні інформації про мережні потоки в стандартну логіку роботи SIEM. Продукт має зручний і зрозумілий web-інтерфейс, що полегшує його вивчення. Кореляційний функціонал обмежений, існує додатковий кореляційний модуль. Продукт позиціонується вендором як рішення для більших інсталяцій, цьому сприяють його багата модульна структура й інтеграція з високорівневими продуктами RSA (Archer, ECAT, RSA Security Operations Management). Можливості тонкого налагодження й кастомизації обмежені.

PT SIEM: українські вендори тільки починають свій шлях на ринку SIEM-рішень, і цей продукт має всі шанси міцно закріпитися на внутрішньому ринку завдяки позиціям вендору й тенденціям до імпортозаміщення. Продукт активно розвивається, і ще занадто рано порівнювати його з гігантами індустрії, але незважаючи на це, навіть поточна версія виглядає життєздатної. У ній уже реалізована більша частина функціонала сучасних SIEM-рішень. Основний продукт вендору – добре, що зарекомендував себе, MaxPatrol, отже, інтеграція SIEM-системи з функціоналом керування уразливостями й аудита систем максимальна.

Виходячи з результатів тестування, хочемо зробити наступні виводи: для більших організацій, холдингових структур рекомендуємо використовувати як платформа SOC рішення від HP через його гнучкість і багатого функціонала. Організаціям не такого великого масштабу рекомендуємо розглянути рішення від компаній McAfee і IBM. Вони підходять тим організаціям, які не готові займатися тонким налаштуванням системи і яким не потрібна специфічна гнучкість платформи для реалізації основних функцій SOC.

2.2 Обґрунтування вибору засобів для побудови системи та мови програмування

Python – динамічна інтерпретована об'єктно-орієнтована скриптова мова програмування із строгою динамічною типізацією. Офіційний сайт мови програмування Python https://www.python.org/. Python – багатоцільова мова програмування, яка дозволяє писати код, що добре читається. Відносний лаконізм мови Python дозволяє створити програму, яка буде набагато коротше свого аналога, написаного на іншій мові. Python – багатоплатформова мова програмування. Це означає, що програми на Python можна запускати в різних операційних системах без будь-яких змін.

Ще однією перевагою Python є його стандартна бібліотека, яка встановлюється разом з Python і містить готові інструменти для роботи з операційною системою, веб-сторінками, базами даних, різними форматами даних, для побудови графічного інтерфейсу програм тощо. Програми, написані на мові програмування Python, можуть бути як невеликими скриптами, так і складними системами. Python абсолютно безкоштовний.

Швидкість виконання коду Python

Один з можливих недоліків Python – швидкість виконання коду. Python не є компільованою мовою. Код на Python спочатку компілюється у внутрішній байт-код, який потім виконується інтерпретатором Python. У більшості випадків при використанні Python виходять програми повільніші в порівнянні з такими мовами, як C.

Втім, сучасні комп'ютери мають таку обчислювальну потужність, що для більшості застосунків швидкість розробки важливіша швидкості виконання, а програми на Python зазвичай пишуться набагато швидше.

Окрім того, Python легко розширюється модулями, написаними на C або C++. Такі модулі можуть використовуватися для виконання частин програми, що створюють інтенсивне навантаження на процесор.

Використання Python

Python використовується для різних цілей: для створення ігор і веб-застосунків, розробки внутрішніх інструментів для різноманітних проектів. Мова також широко застосовується в науковій області для досліджень і розв'язування прикладних завдань.

Застосування мови програмування Python:

1. BitTorrent – протокол для обміну даними.

2. Ubuntu Software Center – вільне програмне забезпечення для пошуку, установки і видалення пакунків в системі Ubuntu Linux.

3. Blender – програма для створення тривимірної комп'ютерної графіки, що включає засоби моделювання, анімації, вимальовування, пост-обробки відео, а також створення відеоігор.

4. GIMP – растровий графічний редактор, із підтримкою векторної графіки.

5. World of Tanks.

6. Вільна енциклопедія Вікіпедія.

7. Пошукова система Google.

8. DropBox – файловий хостинг, що включає персональне хмарне сховище, синхронізацію файлів і програму-клієнт.

9. YouTube – популярне відеосховище.

Версії Python

Мови програмування з часом змінюються – розробники додають в них нові можливості, а також виправляють помилки. Так з'являються різні версії мови. Наприклад, код написаний на Python 2 у більшості випадків не буде працювати у версії Python 3 без внесення додаткових змін.

Процесор є найважливішим компонентом в комп'ютері. Одна з основних функцій процесора – це обробка даних згідно комп'ютерної програми, яка є списком інструкцій, шляхом виконання арифметичних і логічних операцій над фрагментами даних.

Кожна інструкція в програмі – це команда, яка «повідомляє» процесору, яку операцію він повинен виконати. Процесор комп'ютера може розуміти лише ті інструкції, які написані на машинній мові. Машинна мова – це штучна мова, створена для передачі команд комп'ютеру. За допомогою машинної мови створюються ефективні програми, оскільки розробник отримує доступ до всіх можливостей процесора. Машинна мова – мова низького рівня.

Інструкція машинної мови існує для кожної операції, яку процесор здатний виконати – є інструкція для додавання чисел, є інструкція для віднімання чисел і т.д. Увесь набір інструкцій, який центральний процесор може виконати, відомий як набір інструкцій процесора.

Наприклад, у вас є певна програма, яка зберігається на диску вашого комп'ютера. Для виконання програми, ви здійснюєте подвійний клік на значку програми. Це змушує програму копіюватися з диска в оперативну пам'ять, після чого процесор комп'ютера виконує копію програми, яка знаходиться в оперативній пам'яті.

Коли процесор виконує інструкції програми, він бере участь у процесі, який є відомим як цикл fetch – decode – execute (отримати – декодувати – виконати). Цей цикл виконується для кожної інструкції у програмі і складається з трьох кроків:

Отримати

Програма – це послідовність інструкцій на машинній мові. Першим кроком циклу є завантаження (отримання) наступної інструкції з пам'яті в процесор.

Декодувати

Інструкція машинної мови – це двійкове число, яке представляє команду, що повідомляє процесору виконати певну операцію. На цьому кроці процесор декодує інструкцію, яку було «витягнуто» з пам'яті, для визначення того, яка операція повинна виконуватись.

Виконати

Останній крок циклу – виконати операцію.

Хоча процесор комп'ютера розуміє тільки машинну мову, людині непрактично писати програми на машинній мові. Така програма може мати тисячі або навіть мільйони бінарних інструкцій, і написання такої програми буде дуже обтяжливим процесом.

З цієї причини була створена мова асемблера як альтернатива машинній мові. Замість використання двійкових чисел для написання інструкцій, мова асемблера використовує короткі слова, відомі як мнемокоди.

Незважаючи на те, що мова асемблера не вимагає двійкових інструкцій, як у випадку машинної мови, проте вона вимагає високих знань про процесор. Використовуючи мову асемблера, навіть для найпростішої програми, необхідно написати велику кількість інструкцій.

Мова програмування високого рівня дозволяє створювати складні програми, не знаючи, як працює процесор, і не записуючи великої кількості інструкцій низького рівня. Крім того, більшість мов програмування високого рівня використовують слова, які легко зрозуміти.

Python – одна із популярних сучасних мов програмування високого рівня. Python – інтерпретована мова програмування. Python – це високорівнева інтерпретована мова програмування, на відміну від C++, яка є прикладом компільованої мови програмування. Назва Python відноситься як до мови програмування, так і до інтерпретатора – комп'ютерної програми, яка зчитує початковий код (написаний на Python) і виконує інструкції (команди).

Для перекладу мови високого рівня на машинну мову доступні два типи програм:

1. Компілятор.

2. Інтерпретатор.

Завантаження Python

Версії інтерпретатора Python для різних операційних систем доступні для безкоштовного завантаження за адресою https://www.python.org/downloads.

Середовище програмування для Python

Для написання програм використовують текстові редактори або інтегровані середовища розробки, які включають в себе різні інструменти для роботи з кодом: засіб для написання коду (текстовий редактор), інтерактивний інтерпретатор, відлагоджувач тощо.

Текстові редактори та інтегровані середовища програмування для Python:

– IDLE – стандартний редактор Python. Встановлюється разом з Python для користувачів Windows, окремим пакунком для користувачів Linux.

– Notepad++ – безкоштовний текстовий редактор початкового коду, який підтримує велику кількість мов, в тому числі і Python. Лише для користувачів Windows.

– Visual Studio Code – це легкий, але потужний редактор початкового коду, який розповсюджується безкоштовно і доступний у версіях для платформ Linux, Windows і macOS.

– PyScripter – інтегроване середовище розробки для мови програмування Python. Для користувачів Windows. Поширюється безкоштовно.

– Wing IDE 101 – вільне інтегроване середовище для Python, розроблене для навчання програмістів-початківців. Для користувачів Linux, Windows і macOS. Поширюється безкоштовно.

– Geany – вільний текстовий редактор з базовими елементами інтегрованого середовища розробки, доступний для операційних систем Linux, Windows і macOS.

– PyCharm – інтегроване середовище розробки для мови програмування Python. PyCharm є власницьким програмним забезпеченням. Наявна безкоштовна версія Community з усіченим набором можливостей. Для користувачів Linux, Windows і macOS.

2.3 Розгорнута постановка завдання

Згідно з технічним завданням на випускну кваліфікаційну роботу за другим (магістерським) рівнем вищої освіти, реалізації підлягає програмне забезпечення, яке призначено для системи мережевого корпоративного центру управління інформаційною безпекою (SOC).

В процесі розробки випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти необхідно виконати наступний обсяг роботи:

а) провести аналіз існуючих систем-аналогів для виявлення їх позитивних і негативних якостей. Результати аналізу врахувати в подальших розробках;

б) вибрати та обґрунтувати методику побудови системи контролю роботи технологічного обладнання на виробництві в автоматизованому режимі. Розробити функціональну та структурну схеми системи;

в) розробити програмне забезпечення системи, що дозволить реалізувати поставлену технічним завданням задачу. Побудувати блок-схеми алгоритмів програми та підпрограми;

г) організувати інтерфейс користувача з метою формування та виводу на екран ЕОМ повідомлень про некоректні дії користувача та нестандартні ситуації в роботі технологічного обладнання;

д) розробити рекомендації по організаційних та методичних заходах, які забезпечать впровадження системи в промислову експлуатацію та її подальшу успішну експлуатацію;

е) провести розрахунки по визначенню економічної ефективності розробленої системи;

ж) розробити заходи по охороні праці при впровадженні та експлуатації системи, а також розробити заходи з цивільного захисту;

з) сформувати висновки про виконаний обсяг робіт та одержані результати.

3 ОПИС І ОБҐРУНТУВАННЯ ПРОЕКТНИХ РІШЕНЬ

3.1 Опис функціонування системи

Центр операцій безпеки (SOC) – це команда експертів з інформаційної безпеки, які відповідають за моніторинг, виявлення, аналіз, запобігання та реагування на інциденти безпеки. Керований Центр операцій безпеки зазвичай надає консультації, розробку послуг та підтримку організаціям, яким потрібна зовнішня експертиза для захисту від загроз безпеці.

Окрім реагування на вторгнення, SOC також моніторить мережі на предмет незвичайної активності, такої як моделі трафіку, поведінка користувачів або зміни в обмеженнях доступу. Якщо щось здається підозрілим, SOC досліджує це, переглядаючи журнали та файли конфігурації на комп'ютерах і серверах, щоб з'ясувати, що сталося.

Центр кібербезпеки (SOC) можна розглядати як центральну нервову систему мережі, оскільки цей відділ відповідає за постійний моніторинг систем та проведення аналізу для виявлення та запобігання інцидентам кібербезпеки. Мета полягає в тому, щоб діяти до того, як станеться інцидент, але при цьому слід бути готовим до відновлення після інциденту, що складається з нової загрози.

SOC повинна швидко виявляти проблеми та встановлювати їхню серйозність, а також пропонувати рішення для запобігання атакам та припинення будь-яких поточних загроз.

Як працює Центр операцій безпеки

Центр безпеки (SOC) надає критично важливу інформацію про безпеку, щоб команди могли швидко реагувати на кіберінциденти. SOC збирають, співвідносять та аналізують дані з усіх різних комп'ютерів та мереж в організації. Це включає моніторинг у режимі реального часу, виявлення та реагування на оцінку вразливостей, виявлення вторгнень, аналіз та запобігання шкідливому

програмному забезпеченню, мережеву криміналістику, а також технічне обслуговування та усунення несправностей.

Центри безпеки (SOC) найчастіше керуються спеціалізованим членом команди безпеки, який працює цілодобово, щоб контролювати активність у кількох системах та сегментах мережі. Член SOC розробляє звіти, використовуючи інформацію, отриману з журналів, для відстеження атак. Інформація, надана SOC, може бути використана для визначення пріоритетів реагування, визначення найбільш критичних вразливостей у середовищі та швидкої оцінки загального стану безпеки.

SOC має доступ до всіх різних компонентів безпеки в мережі, включаючи брандмауери, системи виявлення вторгнень (IDS), брандмауери додатків, балансувальники навантаження та веб-проксі. Член SOC може об'єднати ці різнорідні фрагменти інформації в одну цілісну картину того, що відбувається в середовищі. Крім того, член SOC може використовувати цю інформацію для швидкої оцінки всіх аспектів середовища, включаючи аномалії в трафіку та даних, нові оцінки вразливостей, незвичайну поведінку відомих інструментів та скриптів шкідливого програмного забезпечення, незвичайну активність веб-додатків, результати тестування на проникнення та інші події.

SOC не замінюють традиційні команди безпеки; радше вони забезпечують їм критичний рівень для покриття прогалин, які команди безпеки не можуть безпосередньо усунути.

Чому SOC такий важливий для бізнесу

Безпека – це дуже важливо. Якщо ви не захищаєтеся від зловмисників, ваша компанія може зазнати масової втрати даних або іншої конфіденційної інформації. І саме тому Центри операцій безпеки такі важливі – вони допомагають великим компаніям контролювати свою безпеку та захищатися від хакерів цілодобово, щодня.

Раніше компанії просто мали когось за столом, хто стежив за будь-якими потенційними загрозами. Але цього вже недостатньо. Зі зростанням компаній, а

їхні процеси (і навіть їхні власні системи безпеки) ускладнювалися, вони виявили, що відстеження всього може бути мамонтовим завданням.

Атаки методом грубої сили на корпоративні мережі стають дедалі поширенішими. Вони можуть коштувати чимало грошей, і коли зловмисник намагається зламати захист однієї корпоративної мережі, стає зрозуміло, що витрачати гроші на найм людей по суті марно, коли справа доходить до визначення реального рівня загрози. Потрібен високотехнологічний моніторинг самих систем безпеки. Те, що починалося як проста ідея для Центрів операцій безпеки, тепер перетворилося на величезну галузь, яка спирається на висококваліфікованих фахівців, які стежать за всім одночасно, забезпечуючи безпеку та надійність корпоративних даних.

Ключові переваги SOC

Звісно ж, наявність SOC має довгий список переваг для бізнесу. Ось деякі з них:

1. Центр операцій безпеки дозволяє компаніям збирати відповідні дані в одному центральному місці.

2. Центр операцій безпеки покращує здатність компаній швидко та рішуче реагувати на нові загрози.

3. Компанії, які інвестують у центри безпеки, з більшою ймовірністю дотримуються правил, що регулюють кібербезпеку.

4. Компанії з центром безпеки мають легшу співпрацю з правоохоронними органами.

5. Центр операцій безпеки дозволяє компаніям швидше виявляти вторгнення та атаки й мінімізувати їх вплив.

6. Центр операцій безпеки дозволяє компаніям краще навчати своїх співробітників кіберризикам та способам їх уникнення.

7. Центр операцій безпеки покращує можливості компаній контролювати активність авторизованих користувачів у своїх системах, значно зменшуючи внутрішні загрози.

8. Наявність центру операцій безпеки покращує можливості компаній моніторити, реагувати та повідомляти про кіберзагрози.

9. Наявність центру операцій безпеки дозволяє компаніям централізувати моніторинг, скорочуючи час повідомлення про інцидент та забезпечуючи швидше вирішення проблем.

10. Центр операцій безпеки дозволяє компаніям надавати кращі послуги своїм клієнтам, дотримуючись галузевих стандартів та державних норм.

11. Центр операцій безпеки дозволяє компаніям бути проактивними, а не реактивними, у боротьбі з кіберзагрозами.

12. Компанії, які інвестують у центри безпеки, з меншою ймовірністю стануть мішенню кіберзлочинців.

13. Центр операцій безпеки дозволяє компаніям дотримуватися галузевих стандартів щодо найкращих практик кібербезпеки.

14. Інвестиції в центр операцій безпеки дозволяють компаніям бути проактивними та звітувати про свої заходи управління ризиками, покращуючи свою репутацію як серед клієнтів, так і серед акціонерів.

15. Центри операцій безпеки дозволяють компаніям уникати штрафів та судових позовів, що виникають унаслідок витоків даних.

Які найбільші проблеми центрів безпеки операцій?

Відстеження загроз та мінімізація їхнього впливу на підприємство є постійним викликом для компаній, але цей виклик став легшим з розвитком технологій. Центри операцій безпеки є критично важливими для інформаційної безпеки. Вони відповідають за моніторинг, оцінку та захист IT-активів.

Ця відповідальність вимагає значного планування та знань як IT-інфраструктури, так і принципів кібербезпеки для успішного виконання. Постійно зростаюча мережева інфраструктура, старіюче обладнання, величезна площа атак та скорочення бюджетів ускладнюють їхню роботу.

Щоб подолати ці проблеми, центри операцій безпеки повинні впроваджувати інновації та автоматизацію для централізації своїх даних. Це

означає перехід від розрізненої, ізольованої інформації до централізованої платформи, де їхні дані доступні та використовуються для спільного використання. Цей крок може бути особливо важливим для компаній з центрами операцій фізичної безпеки (SOC), оскільки вони мають найбільшу потребу в прозорості та більших засобах захисту від фізичних загроз.

Майбутнє державних операційних компаній

У світі, де кіберзлочинці стають дедалі витонченішими, Центри операцій безпеки повинні мати можливість швидко адаптуватися та розвиватися, щоб залишатися на крок попереду конкурентів. Це означає, що їм необхідно мати ресурси та підтримку IT-відділу своєї компанії, а також інструменти, доступні для успішної роботи команд безпеки. Майбутнє центрів операцій безпеки буде рухомим, реагуючи на загрози в режимі реального часу за допомогою автоматизації та машинного навчання. Однак вони ніколи не зможуть замінити кваліфікованих та відданих фахівців з безпеки, яким доводиться виконувати складну роботу.

3.2 Розробка структурної схеми

У складному та швидкозмінному ландшафті сучасних кіберзагроз організації стикаються з безпрецедентним викликом у захисті своїх цифрових активів, конфіденційних даних та забезпеченні безперервності операційної діяльності. Величезний обсяг та складність кібератак вимагають централізованого, проактивного та вузькоспеціалізованого підходу до безпеки. Саме тут Центр операцій безпеки (SOC) стає невід'ємним компонентом загальної стратегії кібербезпеки організації.

Центр операцій безпеки (SOC) – це централізована функція в організації або послуга, що надається третьою стороною, яка займається постійним моніторингом та аналізом стану безпеки організації. Його основна мета – виявляти, запобігати, розслідувати та реагувати на кіберінциденти та загрози

цілодобово. SOC – це набагато більше, ніж просто набір інструментів безпеки, це нервовий центр, де люди, процеси та технології сходяться для забезпечення комплексних операцій безпеки. Він являє собою фундаментальний перехід від реактивних заходів безпеки до проактивного, заснованого на розвідці захисного механізму, призначеного для захисту організації від безлічі потенційних загроз, що ховаються в цифровому світі.

Основна роль Центру операцій безпеки (SOC): понад базовий захист

Роль SOC виходить далеко за рамки простого реагування на сповіщення; вона втілює стратегічний імператив підтримки цифрової стійкості. В епоху, коли кіберзагрози є постійними та дедалі складнішими, багато організацій усвідомлюють, що надійна система безпеки вимагає цілеспрямованого експертного нагляду. Команда операцій безпеки в SOC має головне завдання: забезпечити конфіденційність, цілісність та доступність (тріада ЦРУ) інформаційних активів організації.

Центр безпеки (SOC) діє як центральний розвідувальний центр, постійно збираючи та зіставляючи інформацію про безпеку з усього IT-середовища. Це включає дані з мереж, серверів, кінцевих точок, програм, баз даних та різних інструментів безпеки. Консолідуючи цей величезний обсяг інформації, аналітики SOC отримують цілісне уявлення про ландшафт безпеки, що дозволяє їм виявляти підозрілу активність, яка в іншому випадку могла б залишитися непоміченою. Така всебічна видимість має вирішальне значення для ефективного виявлення загроз, розслідування та швидкого реагування на інциденти. Стратегічна цінність SOC полягає в його здатності перетворювати необроблені дані безпеки на дієву інформацію, що дозволяє командам безпеки не лише виявляти та стримувати загрози, але й розуміти їх походження та запобігати майбутнім випадкам.

Ключові функції та обов'язки SOC

Щоденна діяльність SOC охоплює широкий спектр функцій, кожна з яких є критично важливою для створення та підтримки надійної системи безпеки. Ці

функції виконує спеціальна команда аналітиків та фахівців з безпеки, використовуючи передові рішення безпеки та чітко визначені процеси.

Безперервний моніторинг безпеки та сортування за тривогами

Одним з основоположних обов'язків SOC є постійний моніторинг усіх систем, мереж та програм у режимі реального часу в інфраструктурі організації. Ця цілодобова пильність спрямована на виявлення будь-яких ознак аномальної поведінки або потенційних загроз. Процес моніторингу генерує велику кількість сповіщень від різних інструментів безпеки, таких як брандмауери, системи виявлення/запобігання вторгненням (IDS/IPS), рішення для виявлення та реагування на мережу (NDR), рішення для виявлення та реагування на кінцеві точки (EDR) та системи управління інформацією та подіями безпеки (SIEM).

Аналітики SOC відповідають за початкове сортування цих сповіщень, розрізняючи справжні інциденти безпеки та хибнопозитивні результати. Це вимагає глибокого розуміння нормальної поведінки мережі та законної системної активності. Ефективне сортування значно знижує втому від сповіщень, дозволяючи команді зосередити свою увагу та ресурси на критичних подіях, які становлять реальний ризик. Мета полягає в тому, щоб швидко виявити підозрілу активність, перш ніж вона переросте в повноцінну кібератаку.

Виявлення та аналіз загроз

Після того, як сповіщення класифікується як потенційно шкідливе, команда SOC розпочинає глибше розслідування виявленої загрози. Це включає детальний аналіз журналів, даних мережевого трафіку, телеметрії кінцевих точок та іншої відповідної інформації безпеки, щоб зрозуміти природу, масштаб та потенційний вплив інциденту безпеки. Аналітики використовують розширену аналітику, моделювання поведінки та можливості машинного навчання, часто інтегровані в їхні інструменти безпеки, щоб зіставити, здавалося б, різнорідні фрагменти інформації та виявити закономірності, що вказують на кіберзагрозу.

Використання актуальної інформації про загрози є надзвичайно важливим на цьому етапі. Канали інформації про загрози надають інформацію про відомі

вразливості, нові вектори атак, тактику, методи та процедури зловмисників (TTP), а також індикатори компрометації (IoC). Зіставляючи внутрішні спостереження із зовнішньою інформацією про загрози, аналітики SOC можуть точніше ідентифікувати загрози, розуміти їхні характеристики та прогнозувати їх потенційний розвиток, що дозволяє проводити більш ефективні та цілеспрямовані операції безпеки.

Реагування на інциденти та їх усунення

Кінцева мета виявлення загроз полягає в забезпеченні швидкого та ефективного реагування на інциденти. Коли виявляється підтверджений інцидент безпеки, команда SOC негайно запускає план реагування на інцидент . Цей критичний процес зазвичай включає кілька фаз:

– Ідентифікація: Підтвердження інциденту та збір початкової інформації.

– Стримування: Вжиття негайних заходів для ізоляції уражених систем або мереж, щоб запобігти подальшому поширенню інциденту. Це може включати відключення пристроїв, блокування IP-адрес або карантин шкідливого програмного забезпечення .

– Викорінення: Усунення першопричини інциденту, наприклад, видалення шкідливого програмного забезпечення, виправлення вразливостей або видалення шкідливих конфігурацій.

– Відновлення: Відновлення уражених систем і даних до нормального стану, що може включати відновлення з резервних копій, перебудову систем або переналаштування засобів безпеки.

– Аналіз після інциденту (вивчені уроки): Вирішальний крок, під час якого команда аналізує, що сталося, чому це сталося та як запобігти подібним інцидентам у майбутньому. Цей цикл зворотного зв'язку зміцнює загальний рівень безпеки організації.

Швидкість і точність реагування на інциденти є життєво важливими для мінімізації збитків, скорочення простоїв та підтримки довіри в організації.

Фахівці з реагування на інциденти SOC співпрацюють, часто з іншими IT-відділами, для ефективного усунення загроз.

Управління вразливостями та оцінка ризиків

Проактивний SOC виходить за рамки реагування на активні загрози; він активно працює над зменшенням поверхні атаки організації. Це передбачає постійне управління вразливостями , яке включає виявлення, оцінку та визначення пріоритетів слабких місць безпеки в системах, додатках та мережевій інфраструктурі. Команди SOC часто проводять регулярне сканування вразливостей, тести на проникнення та аудити безпеки, щоб виявити потенційні точки входу для зловмисників.

Після виявлення вразливостей, SOC співпрацює з командами IT-операцій, щоб забезпечити своєчасне встановлення виправлень, посилення конфігурації та впровадження політик безпеки, розроблених для зменшення цих ризиків. Проактивно усуваючи слабкі місця, SOC допомагає зміцнити загальний рівень безпеки та зменшити ймовірність успішних кібератак. Оцінка ризиків – це безперервний процес, який оцінює потенційний вплив виявлених вразливостей та загроз на бізнес-операції та дані.

Полювання на загрози: проактивний захист

Хоча постійний моніторинг та автоматичні сповіщення є важливими, вони часто реагують на відомі загрози або заздалегідь визначені правила. З іншого боку, полювання на загрози – це проактивний та ітеративний процес, у якому аналітики безпеки, яких часто називають мисливцями за загрозами, активно шукають невідомі або невиявлені загрози, що ховаються в мережі організації. Це передбачає використання глибокого контекстуального розуміння середовища, розвідки загроз та розслідувань на основі гіпотез.

Мисливці за загрозами працюють, виходячи з припущення, що організація вже була скомпрометована або зазнає малопомітної атаки. Вони шукають аномальну поведінку, малопомітні індикатори компрометації (IoC) та відхилення від нормальних базових показників, які автоматизовані інструменти можуть

пропустити. Ця вузькоспеціалізована функція вимагає розширених аналітичних навичок, глибоких знань методологій зловмисників та доступу до багатих, детальних джерел даних, особливо до мережевих пакетних даних. Мета полягає у виявленні складних, прихованих кіберзагроз, таких як передові постійні загрози (APT), перш ніж вони зможуть завдати значної шкоди.

Інтеграція з управлінням інформацією та подіями безпеки (SIEM)

Система управління інформацією та подіями безпеки (SIEM) є базовою технологією майже для кожної SOC. Рішення SIEM агрегує та централізує дані журналів і подій практично з кожного пристрою та програми в IT-інфраструктурі організації. Це включає дані з брандмауерів, серверів, операційних систем, мережевих пристроїв, інструментів безпеки та програм.

Потім SIEM нормалізує ці різнорідні дані, роблячи їх узгодженими та зручними для пошуку. Найголовніше, що SIEM використовує правила кореляції та аналітичні механізми для виявлення закономірностей та зв'язків у даних, які вказують на підозрілу активність або потенційні інциденти безпеки. Наприклад, він може співвіднести невдалу спробу входу на одному сервері з успішним входом з того ж облікового запису користувача на іншому сервері в незвичному місці, позначаючи це як потенційну компрометацію.

Ключова відмінність між SIEM та SOC полягає в тому, що SIEM є потужним інструментом, який використовується *SOC* . SOC – це операційна команда та об'єкт, який використовує SIEM (разом з багатьма іншими інструментами та процесами) для досягнення своїх цілей моніторингу, виявлення, аналізу та реагування на загрози. Без SIEM здатність SOC отримувати повну видимість та співвідносити події була б серйозно обмежена, що зробить виявлення загроз, розслідування та реагування на інциденти набагато складнішими та трудомісткішими.

Автоматизація та оркестрація безпеки (SOAR)

Щоб впоратися зі зростаючим обсягом та складністю інцидентів безпеки, багато сучасних центрів охорони безпеки (SOC) інтегрують платформи

оркестрації, автоматизації та реагування на безпеку (SOAR). Технології SOAR дозволяють командам SOC стандартизувати, автоматизувати та оркеструвати робочі процеси операцій безпеки.

Автоматизація передбачає автоматичне виконання завдань і процесів, таких як блокування шкідливої IP-адреси, виявленої за допомогою сповіщення SIEM, ізоляція зараженої кінцевої точки або збагачення інциденту даними розвідки про загрози. Оркестрація означає координацію кількох інструментів та систем безпеки для безперебійної спільної роботи в межах визначеного робочого процесу, часто ініційованого сповіщенням. Можливості реагування в SOAR надають інструменти для управління інцидентами, співпраці та створення звітів.

Впровадження автоматизації та оркестрації значно підвищує ефективність та швидкість роботи SOC. Це зменшує ручне навантаження на аналітиків безпеки, дозволяючи їм зосередитися на складних розслідуваннях та стратегічному пошуку загроз, а не на повторюваних завданнях. Це також призводить до швидшого реагування на інциденти, мінімізуючи вікно можливостей для зловмисників та зменшуючи потенційний вплив кібератаки.

Відповідність та звітність

Окрім безпосереднього зменшення загроз, SOC відіграє вирішальну роль у забезпеченні дотримання організацією різних нормативних вимог та галузевих стандартів. Багато організацій підпадають під дію таких вимог щодо дотримання, як GDPR, HIPAA, PCI DSS та SOX, які вимагають суворого контролю безпеки та надійних механізмів звітності. SOC постійно контролює системи, щоб забезпечити їх відповідність цим правилам та внутрішнім політикам безпеки. Вони генерують детальні звіти про стан безпеки, виявлені інциденти, вразливості та зусилля з усунення наслідків, надаючи необхідну документацію для аудитів та демонструючи належну перевірку. Ця звітність не лише служить регуляторним цілям, але й надає вищому керівництву цінну інформацію щодо стану безпеки організації та поточних ризиків, допомагаючи приймати стратегічні рішення та розподіляти ресурси для рішень безпеки.

Неперевершена потужність пакетних даних у SOC

Хоча журнали та сповіщення надають цінну інформацію про те, що сталося в системі, вони часто пропонують узагальнений або відфільтрований огляд подій. Для справді повного розуміння мережевої активності та найдеталізованішого рівня деталізації, необхідного для розширеного виявлення та пошуку загроз, SOC значною мірою покладається на дані мережевих пакетів . Цей необроблений, нефільтрований потік інформації, що проходить мережею, забезпечує неперевершений рівень видимості, виступаючи головним джерелом достовірної інформації в розслідуваннях кібербезпеки.

Чому саме пакетні дані? Нефільтрована правда та глибока прозорість

Мережеві пакетні дані представляють собою кожен фрагмент інформації, що передається мережею, включаючи джерело, пункт призначення, протокол і корисне навантаження кожного зв'язку. На відміну від журналів, які генеруються певними програмами або системами і можуть не фіксувати всі деталі або навіть бути підроблені зловмисниками, пакетні дані пропонують незмінний запис мережевого трафіку. Це «істина», яка точно показує, що відбувалося в мережі.

Ця глибока видимість є критично важливою, оскільки:

– Відсутність сліпих зон: Пакетні дані фіксують усі мережеві комунікації, незалежно від того, чи генерують вони запис у журналі, чи їх бачать інші засоби безпеки. Це означає, що можна виявити приховані канали командування та управління (C2), спроби прихованого витоку даних або складні горизонтальні переміщення, що обходять засоби контролю безпеки кінцевих точок.

– Незаперечні докази: Для криміналістики та реагування на інциденти пакетні дані надають переконливі докази зловмисної діяльності. Вони можуть реконструювати цілі шляхи атаки, показати послідовність подій та точно визначити момент компрометації або витоку даних.

– Поза межами сигнатур: Хоча системи виявлення на основі сигнатур спираються на відомі шкідливі шаблони, аналіз пакетів може виявити аномальну поведінку, навіть для загроз нульового дня або варіацій відомого шкідливого

програмного забезпечення, для яких ще не існує сигнатури. Поведінковий аналіз пакетних даних може виявити незвичне використання протоколу, шаблони зв'язку або розміри передачі даних.

Практичні висновки: від фрагментів до порушень

Здатність захоплювати, зберігати та аналізувати мережеві пакетні дані трансформує можливості SOC ефективно виявляти, розуміти, виявляти та видаляти загрози.

– Ідентифікація: Точне виявлення аномалій та вторгнень

Пакетні дані дозволяють аналітикам SOC швидко виявляти аномалії, які можуть сигналізувати про кібератаку. Це включає:

– Незвичайні мережеві потоки: виявлення підключень до підозрілих IP-адрес, нетипового використання портів або обсягів трафіку, що суттєво відхиляються від встановлених базових значень.

– Зв'язок командування та управління (C2): Виявлення закономірностей, що вказують на канали C2, таких як активність маяків, використання нестандартних протоколів або зв'язок з відомими шкідливими доменами.

– Витік даних: розпізнавання передачі великих обсягів даних до зовнішніх, несанкціонованих місць призначення або надсилання незвичайних типів файлів, що може свідчити про крадіжку даних.

– Внутрішні загрози: моніторинг внутрішнього мережевого трафіку на предмет несанкціонованого доступу до конфіденційних систем, незвичайного переміщення даних привілейованими користувачами або порушень політик, що свідчать про зловмисну інсайдерську діяльність.

Аналізуючи пакетні дані в режимі реального або майже реального часу, команди SOC можуть отримати негайне уявлення про підозрілу активність, що дозволяє швидко виявляти потенційні інциденти безпеки до їх ескалації.

– Полювання на загрози: слідування цифровим хлібним крихтам

Для проактивного пошуку загроз пакетні дані є незамінними. Мисливці за загрозами використовують пакетні дані для:

– Реконструкція шляхів атаки: Відстежуючи потік мережевого трафіку, мисливці можуть відстежувати горизонтальне переміщення зловмисника в мережі, розуміти, як він отримав доступ, та ідентифікувати скомпрометовані системи. Це забезпечує повну картину ланцюжка знищення кібератаки.

– Виявлення прихованих загроз: Досвідчені зловмисники часто використовують легітимні інструменти та протоколи, щоб зливатися зі звичайним мережевим трафіком, що ускладнює їх виявлення за допомогою традиційних заходів безпеки. Аналіз пакетів може виявити ці тонкі індикатори, такі як незвичне використання протоколів, зашифровані тунелі або приховані канали, що використовуються для зв'язку.

– Перевірка гіпотез: Коли мисливець за загрозами підозрює певний тип атаки або наявність індикатора компрометації (IoC) у мережі, він може використовувати пакетні дані для перевірки своєї гіпотези. Наприклад, якщо відомо, що певне шкідливе програмне забезпечення зв'язується через певний порт, пакетні дані можуть підтвердити, чи справді будь-які внутрішні хости здійснюють такі з'єднання.

– Розуміння масштабу: Пакетні дані допомагають визначити повний масштаб порушення, ідентифікуючи всі уражені системи та дані, що має вирішальне значення для комплексного стримування.

– Видалення та відновлення: цілеспрямована та науково обґрунтована відповідь

Коли інцидент підтверджено, пакетні дані надають незаперечні докази, необхідні для цілеспрямованого стримування та ліквідації:

– Точне стримування: Визначивши точне джерело та пункт призначення шкідливого трафіку, команди SOC можуть впроваджувати високоточні заходи стримування, такі як блокування певних з’єднань або ізоляція скомпрометованих хостів, мінімізуючи порушення законних операцій.

– Ефективне знищення: Пакетні дані розкривають методи, що використовуються зловмисниками, спрямовуючи зусилля з знищення.

Наприклад, якщо витік даних стався через певний протокол, команда точно знає, на чому зосередити свої зусилля для очищення та запобігання.

– Криміналістика після інциденту: Для ретельного аналізу після інциденту пакетні дані є безцінними. Вони дозволяють судовим слідчим відтворювати мережеву активність, аналізувати корисні навантаження та розуміти повний вплив порушення, сприяючи покращенню політик та практик безпеки.

– Аналіз першопричин: Розуміння того, «як» і «чому» відбувається атака, є надзвичайно важливим. Пакетні дані дозволяють провести глибокий аналіз першопричин, визначити початкову точку проникнення та використані вразливості, що життєво важливо для запобігання повторенню.

За межами реактивності: проактивна безпека з пакетними даними

Стратегічна інтеграція пакетних даних в операції SOC перетворює безпеку з виключно реактивної моделі, що керується сповіщеннями, на проактивний захист, що керується розвідкою. Забезпечуючи нефільтрований, комплексний огляд мережевої активності, це дає командам безпеки можливість виявляти, виявляти та усувати загрози з неперевершеною точністю та швидкістю. Ця можливість є основою для досягнення справжньої кіберстійкості, дозволяючи організаціям випереджати складних супротивників та захищати свої найважливіші активи. Сучасні рішення для виявлення та реагування на мережі (NDR) спеціально розроблені для використання можливостей пакетних даних, пропонуючи глибоку видимість та поведінкову аналітику для розширення можливостей SOC.

Ключові ролі в команді SOC

Високоефективна SOC спирається на міждисциплінарну команду, кожен член якої вносить свій спеціалізований внесок у зусилля з колективної безпеки. Хоча конкретні посади та рівні можуть відрізнятися, основні функції загалом однакові:

– Менеджер/керівник SOC: Ця особа здійснює стратегічний нагляд та керівництво всім SOC. Вона відповідає за визначення бачення SOC, встановлення операційних цілей, управління персоналом, забезпечення дотримання політик безпеки та ескалацію критичних інцидентів вищому керівництву. Менеджер SOC також відіграє ключову роль в ініціативах щодо постійного вдосконалення та сприяння культурі досконалості в команді.

– Аналітики безпеки (рівень 1, 2, 3): це захисники на передовій, які формують ядро команди SOC. Їхні ролі зазвичай розподіляються на рівні залежно від досвіду та складності інцидентів, з якими вони працюють:

o Аналітики 1-го рівня: часто їх називають «спеціалістами з сортування сповіщень», ці аналітики відповідають за початковий моніторинг сповіщень безпеки, фільтрацію хибнопозитивних результатів та проведення попередніх розслідувань. Вони дотримуються встановлених методичних рекомендацій для ескалації підтверджених інцидентів на вищий рівень. Їхні обов'язки зазвичай включають цілодобовий моніторинг, перевірку початкових сповіщень та збір базових даних.

o Аналітики 2-го рівня: це більш досвідчені аналітики, які проводять глибші розслідування інцидентів, що ескалирували з 1-го рівня. Вони виконують детальний аналіз, використовують передові інструменти безпеки (включаючи SIEM та NDR) та працюють над стратегіями стримування. Вони вміють розуміти тактику зловмисників та розробляти дії негайного реагування.

o Аналітики 3-го рівня: Найстарші та найкваліфікованіші аналітики безпеки, також відомі як «мисливці за загрозами» або «судові слідчі». Вони обробляють найскладніші та найсучасніші загрози, виконують проактивне полювання на загрози, проводять поглиблений судово-медичний аналіз та розробляють власні правила виявлення. Вони часто мають досвід у зворотному проектуванні, аналізі шкідливих програм та передових методологіях боротьби з постійними загрозами (APT) .

– Мисливці за загрозами: Хоча вони часто є підгрупою аналітиків 3-го рівня, деякі організації присвячують певні ролі пошуку загроз. Ці фахівці є проактивними, керованими гіпотезами слідчими, які активно шукають невідомі загрози, що обійшли існуючі засоби контролю безпеки. Вони використовують складні методи та величезні набори даних, включаючи мережеві пакетні дані, для виявлення прихованих супротивників.

– Реагування на інциденти: Ці фахівці спеціалізуються на фазах стримування, ліквідації та відновлення після інцидентів. Хоча всі аналітики SOC беруть участь у реагуванні на інциденти, спеціалізовані фахівці з реагування на інциденти часто керують повним життєвим циклом серйозного порушення, координуючи зусилля кількох команд та забезпечуючи швидке повернення до нормальної роботи. Вони вміють аналізувати та звітувати про інциденти після інциденту.

– Фахівці з управління вразливостями: Ці члени команди зосереджуються на виявленні, оцінці та визначенні пріоритетів вразливостей в інфраструктурі організації. Вони проводять сканування, аналізують результати та співпрацюють з IT-командами, щоб забезпечити усунення вразливостей, тим самим зменшуючи поверхню атаки.

– Інженери/архітектори безпеки: Хоча інженери безпеки не завжди входять до безпосередньої операційної команди SOC, вони часто тісно співпрацюють з SOC. Вони відповідають за проектування, впровадження та підтримку інфраструктури та інструментів безпеки (SIEM, EDR, NDR, брандмауери тощо), на які покладається SOC. Вони надають експертизу в оптимізації рішень безпеки та інтеграції нових технологій.

Спільний характер цих ролей є важливим для ефективної роботи SOC, що забезпечує постійний моніторинг, захист та вдосконалення всіх аспектів кібербезпеки організації.

Типи моделей SOC

Організації можуть впроваджувати Центр операцій безпеки кількома способами, кожен з яких має свої переваги та міркування, залежно від таких факторів, як бюджет, внутрішні можливості та конкретні вимоги безпеки.

– Внутрішній SOC: Ця модель передбачає створення та функціонування SOC повністю всередині організації.

o Переваги: Забезпечує повний контроль над операціями, процесами та даними безпеки. Дозволяє глибоку інтеграцію з внутрішніми бізнес-процесами та конкретними політиками безпеки. Команда отримує глибокі знання про унікальне середовище та ризики організації.

o Недоліки: Вимагає значних початкових інвестицій у технології, інфраструктуру та кваліфікований персонал. Укомплектування цілодобової операції експертами-аналітиками безпеки може бути складним та дорогим через нестачу фахівців з кібербезпеки. Поточні операційні витрати, включаючи навчання та обслуговування інструментів, можуть бути високими.

– Аутсорсинг SOC (SOC-as-a-Service / SOCaaS): У цій моделі організація укладає договір зі стороннім постачальником керованих послуг безпеки (MSSP) для виконання функцій SOC. SOC-as-a-Service (SOCaaS) – це популярна пропозиція, коли постачальник керує моніторингом, виявленням загроз, попереднім розслідуванням та реагуванням на інциденти з власних потужностей.

o Переваги: Економічно ефективний, оскільки дозволяє уникнути капітальних витрат та проблем із персоналом, пов’язаних із внутрішньою SOC. Забезпечує негайний доступ до спеціалізованих експертів з кібербезпеки, часто з цілодобовим покриттям. Пропонує масштабованість та може надавати інформацію про загрози, що перевищує ту, яку може зібрати одна організація. Швидке розгортання рішень безпеки.

o Недоліки: Менший прямий контроль над операціями безпеки та даними. Організації повинні ретельно перевіряти постачальників, щоб забезпечити довіру та дотримання вимог відповідності. Потенційні проблеми щодо суверенітету

даних та того, як конфіденційна інформація безпеки обробляється третьою стороною.

– Гібридний SOC: Ця модель поєднує елементи як внутрішнього, так і аутсорсингового підходів. Наприклад, організація може підтримувати менший внутрішній SOC для критичних систем та реагування на основні інциденти, одночасно передаючи цілодобовий моніторинг та первинне сортування сповіщень постачальнику послуг з управління операційною службою (MSSP).

o Переваги: Поєднує контроль з економічною ефективністю та доступом до спеціалізованих експертів. Дозволяє організації зосередитися на своїх найбільш чутливих активах, використовуючи зовнішні ресурси для ширшого охоплення.

o Недоліки: Потрібна ретельна координація та чіткий розподіл обов'язків між внутрішньою командою та зовнішнім постачальником, щоб уникнути прогалин або дублювання.

– Кероване виявлення та реагування (MDR): Хоча MDR часто надається постачальниками послуг з управління загрозами (MSSP), це окрема послуга, яка зосереджена саме на розширеному виявленні загроз, розслідуванні загроз, пошуку загроз та швидкому реагуванні на інциденти. На відміну від традиційного SOCaaS, який може більше зосереджуватися на моніторингу безпеки та базовому оповіщенні, MDR йде глибше, пропонуючи проактивне виявлення загроз та управління інцидентами під керівництвом експертів.

o Переваги: Забезпечує більш проактивний та практичний підхід до управління загрозами. Пропонує спеціалізованих мисливців за загрозами та реагування на інциденти, які активно виявляють та усувають загрози. Може доповнити існуючу внутрішню команду безпеки.

o Недоліки: Може бути дорожчим за базовий SOCaaS. Вимагає високого рівня довіри з постачальником, враховуючи його глибокий доступ до середовища організації.

Вибір моделі SOC значною мірою залежить від унікального профілю ризиків організації, наявності ресурсів та стратегічних цілей безпеки.

Побудова ефективної SOC: найкращі практики

Створення та функціонування ефективного Центру операцій безпеки вимагає не лише придбання передових інструментів безпеки; воно вимагає стратегічного підходу, що охоплює людей, процеси та технології. Дотримання найкращих практик може значно покращити здатність Центру операцій безпеки захищати організацію від кіберзагроз.

– Визначте чіткі цілі та показники: Перш ніж створювати або оптимізувати SOC, вкрай важливо визначити його цілі. Які конкретні ризики він враховуватиме? Як буде вимірюватися його успіх? Цілі повинні відповідати загальним бізнес-цілям та дотриманню нормативних вимог. Ключові показники ефективності (KPI) та показники, такі як середній час виявлення (MTTD), середній час отримання знань (MTTK), середній час реагування (MTTR), кількість хибнопозитивних результатів та серйозність інцидентів, є важливими для постійного вдосконалення та демонстрації цінності.

– Інвестуйте в правильний технологічний стек: Потужний SOC спирається на надійний набір рішень безпеки. Зазвичай це включає SIEM для агрегації та кореляції журналів, Endpoint Detection and Response (EDR) або Extended Detection and Response (XDR) для видимості кінцевих точок, Network Detection and Response (NDR) для глибокої видимості мережі (особливо пакетних даних), Security Orchestration, Automation, and Response (SOAR) для ефективності, а також платформи розвідки загроз. Ключем є інтеграція та сумісність між цими інструментами для створення єдиної екосистеми безпеки.

– Формуйте кваліфіковану команду: лише технологій недостатньо. Ефективність SOC залежить від досвіду її аналітиків та спеціалістів з безпеки. Інвестуйте в постійне навчання та професійний розвиток, щоб підтримувати навички в актуальному стані з урахуванням кіберзагроз та технологій, що розвиваються. Сприяйте культурі безперервного навчання, обміну знаннями та

співпраці в командах SOC. Розгляньте можливість перехресного навчання для розвитку резервних можливостей та стійкості.

– Встановлення надійних процесів та інструкцій: Чіткі, добре задокументовані процеси є основою ефективних операцій SOC. Розробіть комплексні інструкції реагування на інциденти для різних типів інцидентів безпеки, що окреслюють покрокові процедури виявлення, аналізу, стримування, ліквідації та відновлення. Впроваджуйте стандартизовані робочі процеси для сортування тривог, управління вразливостями та звітності. Регулярні навчання та симуляції (навчання «Червона команда» проти «Синьої команди») допомагають удосконалити ці процеси та забезпечити готовність команди до реальних сценаріїв.

– Інтеграція інформації про загрози: Щоб випереджати розвиток кіберзагроз, SOC повинен постійно отримувати та інтегрувати відповідну інформацію про загрози. Це включає інформацію про нові вразливості, нові вектори атак, групи зловмисників та індикатори компрометації (IoC). Інтеграція цієї інформації в правила SIEM, запити на пошук загроз та процеси управління вразливостями дозволяє SOC проактивно виявляти та пом'якшувати ризики.

– Впровадження автоматизації та оркестрації: для боротьби зі втомою від сповіщень та покращення часу реагування використовуйте автоматизацію для повторюваних завдань (наприклад, блокування шкідливих IP-адрес, ізоляція кінцевих точок, збагачення сповіщень). Платформи SOAR можуть оркеструвати складні робочі процеси, забезпечуючи послідовне та швидке виконання дій реагування, звільняючи аналітиків безпеки для зосередження на складніших аналітичних та пошукових заходах.

– Постійне вдосконалення: SOC не є статичною структурою; вона повинна постійно адаптуватися та вдосконалюватися. Регулярно переглядайте дані про інциденти, проводите аналіз після інцидентів для виявлення отриманих уроків та вдосконалюйте процеси та технології на основі цих даних. Збирайте відгуки від команди SOC та інших зацікавлених сторін для оптимізації операцій

та покращення загального стану безпеки. Регулярно оцінюйте нові рішення безпеки та коригуйте стратегію для реагування на нові кіберзагрози.

Ретельно впроваджуючи ці найкращі практики, організації можуть створити та вдосконалити високоефективний Центр операцій безпеки, який забезпечує надійні, проактивні можливості кіберзахисту від сучасного складного ландшафту загроз.

Рисунок 3.1 – Структурна схема системи

3.3 Розробка функціональної схеми

Функціональна схема розробленої системи зображена на рисунку 3.2.

З рисунку видно, що розроблена система складається з наступних частин:

– Блок виявлення дестабілізуючого трафіку у корпоративному центрі управління інформаційною безпекою (SOC).

– Блок визначення деструктивних дій та виду атаки у корпоративному центрі управління інформаційною безпекою (SOC).

– Блок зберігання результатів.

– Блок моніторингу мережі у корпоративному центрі управління інформаційною безпекою (SOC).

– Блок аналізу мережної статистики у корпоративному центрі управління інформаційною безпекою (SOC).

– Блок визначення топології мережі у корпоративному центрі управління інформаційною безпекою (SOC).

Блок виявлення дестабілізуючого трафіку у корпоративному центрі управління інформаційною безпекою (SOC)

Блок виявлення дестабілізуючого трафіку у корпоративному центрі управління інформаційною безпекою (SOC):

– Блок визначення профілю поводження нормального трафіку у корпоративному центрі управління інформаційною безпекою (SOC).

– Блок заміни направлення трафіку у корпоративному центрі управління інформаційною безпекою (SOC).

– Блок усунення дестабілізуючого трафіку у корпоративному центрі управління інформаційною безпекою (SOC).

При первісному розгортанні рішення по DDoS адміністратор створює профіль поводження нормального трафіку у корпоративному центрі управління інформаційною безпекою (SOC). Цей процес іменується навчанням. Компанія використовує додатки звичайним образом протягом 24 годин протягом одного тижня, і трафік додатка проходить через Детектор аномалій трафіку у корпоративному центрі управління інформаційною безпекою (SOC). У період

навчання Детектор аномалій трафіку у корпоративному центрі управління інформаційною безпекою (SOC) збирає базову інформацію для розуміння нормальної роботи мережі у корпоративному центрі управління інформаційною безпекою (SOC), куди входять:

– Інтенсивність пакетів для кожного типу пакетів, обмірювана як кількість пакетів у секунду (pps).

– Співвідношення пакетів, наприклад, співвідношення пакетів SYN і пакетів FIN.

– Кількість одночасних TCP-з'єднань, відкритих одним джерелом.

Базова інформація збирається по кожній цільовій адресі хост-ПК, цільовій підмережі у корпоративному центрі управління інформаційною безпекою (SOC), вихідній адресі хост-ПК і вихідній підмережі у корпоративному центрі управління інформаційною безпекою (SOC).

Після закінчення періоду навчання Детектор аномалій трафіку у корпоративному центрі управління інформаційною безпекою (SOC) переводиться в режим моніторингу, а Блок усунення дестабілізуючого трафіку у корпоративному центрі управління інформаційною безпекою (SOC) – у резервний режим готовності. Доти, поки немає атаки у корпоративному центрі управління інформаційною безпекою (SOC), що активно розвивається, вхідний трафік з мережі у корпоративному центрі управління інформаційною безпекою (SOC) Інтернет проходить через комутатор без якого-небудь втручання з боку Блоку усунення дестабілізуючого трафіку у корпоративному центрі управління інформаційною безпекою (SOC). Копія вхідного трафіку у корпоративному центрі управління інформаційною безпекою (SOC) посилає для аналізу на Детектор аномалій трафіку у корпоративному центрі управління інформаційною безпекою (SOC) через зовнішній аналізатор протоколів (SPAN) або віртуальні списки ACL. Якщо Детектор аномалій трафіку у корпоративному центрі управління інформаційною безпекою (SOC) виявляє дестабілізуюче в порівнянні з базовою інформацією поводження трафіку у корпоративному центрі управління інформаційною безпекою (SOC), починається процес усунення:

– Детектор аномалій трафіку у корпоративному центрі управління інформаційною безпекою (SOC) направляє в Блок усунення дестабілізуючого трафіку у корпоративному центрі управління інформаційною безпекою (SOC) команду почати процес зміни напрямку.

– Блок усунення дестабілізуючого трафіку у корпоративному центрі управління інформаційною безпекою (SOC) відхиляє (“захоплює”) трафік, адресований на атакуєму IP-адресу, переадресуючи його на самого себе.

– Блок усунення дестабілізуючого трафіку у корпоративному центрі управління інформаційною безпекою (SOC) піддає трафік багатоступінчастому аналізу й застосовує контрзаходи для відділення благонадійних джерел від джерел атаки у корпоративному центрі управління інформаційною безпекою (SOC). Цей процес іменується очищенням або вичищенням.

– Блок усунення дестабілізуючого трафіку у корпоративному центрі управління інформаційною безпекою (SOC) скидає трафік атаки у корпоративному центрі управління інформаційною безпекою (SOC) й пересилає благонадійний трафік назад на нормальний маршрут проходження трафіку у корпоративному центрі управління інформаційною безпекою (SOC) до мети. Цей процес іменується ін'єкцією.

Детектор аномалій трафіку у корпоративному центрі управління інформаційною безпекою (SOC)

Детектор аномалій трафіку у корпоративному центрі управління інформаційною безпекою (SOC) – це пасивний пристрій моніторингу, що постійно виявляє ознаки, що вказують на присутність атаки у корпоративному центрі управління інформаційною безпекою (SOC) DDoS, спрямованої проти захищеного місця призначення, також іменованого зоною. Це може бути сервер, інтерфейс міжмережного екрана або інтерфейс маршрутизатора. Детектор аномалій трафіку у корпоративному центрі управління інформаційною безпекою (SOC) аналізує копії всього вхідного трафіку у корпоративному центрі управління інформаційною безпекою (SOC), адресуємого в захищені зони, через SPAN або відгалуження пасивної мережі у корпоративному центрі управління інформаційною безпекою (SOC). Цей аналіз включає зіставлення поточного

поводження трафіку у корпоративному центрі управління інформаційною безпекою (SOC) з базовими граничними параметрами, які також іменуються зональною політикою, для виявлення дестабілізуючого трафіку у корпоративному центрі управління інформаційною безпекою (SOC). Якщо дестабілізуюче поводження виявлене й виглядає як можлива атака, Детектор аномалій трафіку у корпоративному центрі управління інформаційною безпекою (SOC) через позаполосну управлінську мережу Ethernet посилає в Блок усунення дестабілізуючого трафіку у корпоративному центрі управління інформаційною безпекою (SOC) сигнал про початок аналізу й усунення атаки у корпоративному центрі управління інформаційною безпекою (SOC).

Блок усунення дестабілізуючого трафіку у корпоративному центрі управління інформаційною безпекою (SOC)

Блок усунення дестабілізуючого трафіку у корпоративному центрі управління інформаційною безпекою (SOC) – це автономний пристрій аналізу й фільтрації трафіку у корпоративному центрі управління інформаційною безпекою (SOC). Починаючи прийом трафіку у корпоративному центрі управління інформаційною безпекою (SOC), адресованого в конкретну зону, що, очевидно, піддається атаці, Блок усунення дестабілізуючого трафіку у корпоративному центрі управління інформаційною безпекою (SOC) проводить точний аналіз цього трафіку у корпоративному центрі управління інформаційною безпекою (SOC). Якщо результати аналізу підтверджують, що трафік злочинний, Блок усунення дестабілізуючого трафіку у корпоративному центрі управління інформаційною безпекою (SOC) застосовує контрзаходи, наприклад, механізми анти-спуфінга й фільтрацію різного рівня. Кінцевий результат полягає в тому, що трафік зі злочинних джерел скидається, а трафік із благонадійних джерел пересилається в передбачений пункт призначення.

Атаки у корпоративному центрі управління інформаційною безпекою (SOC) DDoS – Виявлення й усунення

Блок усунення дестабілізуючого трафіку у корпоративному центрі управління інформаційною безпекою (SOC).

Можливі варіанти зміни напрямку трафіку у корпоративному центрі управління інформаційною безпекою (SOC)

Фахівці з IT можуть використовувати описані нижче варіанти зміни напрямку трафіку у корпоративному центрі управління інформаційною безпекою (SOC) з його пересиланням з мережі у корпоративному центрі управління інформаційною безпекою (SOC), розташованого вище лежачого оператора зв'язку, на Блок усунення дестабілізуючого трафіку у корпоративному центрі управління інформаційною безпекою (SOC). Цей процес також іменується "захватом" трафіку у корпоративному центрі управління інформаційною безпекою (SOC):

– Повідомлення прикордонного шлюзового протоколу (Border Gateway Protocol, BGP) із Блок усунення дестабілізуючого трафіку у корпоративному центрі управління інформаційною безпекою (SOC) на маршрутизатори, розташовані у вище лежачого оператора зв'язку, з інформацією про те, що трафік, адресований на захищену адресу призначення, буде переспрямований на Блок усунення дестабілізуючого трафіку у корпоративному центрі управління інформаційною безпекою (SOC).

– Використання зовнішніх механізмів зміни напрямку трафіку у корпоративному центрі управління інформаційною безпекою (SOC), наприклад, маршрутизаторів віддаленого відновлення BGP.

– Повідомлення про ін'єкцію очищеного трафіку у корпоративному центрі управління інформаційною безпекою (SOC) на маршруті (Route Health Injection, RHI) від Блок усунення дестабілізуючого трафіку у корпоративному центрі управління інформаційною безпекою (SOC) для процесу маршрутизації в Catalyst

серії 6500 або в систему нагляду серії 7600. Ці повідомлення поміщають статичний маршрут у глобальну таблицю маршрутизації, у якій модуль Блок усунення дестабілізуючого трафіку у корпоративному центрі управління інформаційною безпекою (SOC) позначений як наступний вузол.

Можливі варіанти ін'єкції трафіку у корпоративному центрі управління інформаційною безпекою (SOC)

Ін'єкція трафіку у корпоративному центрі управління інформаційною безпекою (SOC) – це процес, застосовуваний у Блок усунення дестабілізуючого трафіку у корпоративному центрі управління інформаційною безпекою (SOC) для пересилання очищеного благонадійного трафіку у корпоративному центрі управління інформаційною безпекою (SOC) в точку призначення, що піддається атаці. Рішення підтримує різні варіанти ін'єкції трафіку у корпоративному центрі управління інформаційною безпекою (SOC). У варіанті 2-ого рівня топології, очищений трафік пересилається із Блок усунення дестабілізуючого трафіку у корпоративному центрі управління інформаційною безпекою (SOC) на статично-конфігуруєму наступну адресу заходу. Ця адреса перебуває на маршрутизаторі, розташованому нижче й з'єднаним з тої ж VLAN або підмережею, що й інтерфейс/VLAN ін'єкції трафіку у корпоративному центрі управління інформаційною безпекою (SOC). Ін'єкцію трафіку у корпоративному центрі управління інформаційною безпекою (SOC) на 2-му рівні найпростіше конфігурувати, оскільки тут не потрібно вносити які-небудь істотні зміни в конфігурацію маршрутизатора, розташованого нижче.

Варіанти ін'єкції трафіку у корпоративному центрі управління інформаційною безпекою (SOC) 3-го рівня:

– Маршрутизація й пересилання по VPN (VPN Routing and Forwarding, VRF).

– Маршрутизація на основі політики (Policy-Based Routing, PBR).

– Транкинг VLAN (VLAN Trunking).

– Інкапсуляція по загальній маршрутизації (GRE) або інкапсуляція IP у тунелі IP (IPIP).

Блок визначення деструктивних дій та виду атаки у корпоративному центрі управління інформаційною безпекою (SOC)

Блок визначення деструктивних дій та виду атаки у корпоративному центрі управління інформаційною безпекою (SOC):

– Широкомовний шторм.

– Додатки, що роблять інтенсивне широкомовне розсилання, наприклад: широкомовні чати й мережні ігри.

– Атака ARP-spoofing на таблицю mac-адрес комутаторів.

Широкомовний шторм

Широкомовний шторм – лавина (сплеск) широкомовних акетів (на другому рівні моделі OSI – кадрів). Розмноження некоректно сформованих широкомовних повідомлень у кожному вузлі приводить до експонентного росту їхнього числа й паралізує роботу мережі у корпоративному центрі управління інформаційною безпекою (SOC). Звичайно такі пакети використовуються мережними сервісами для оповіщення станцій про свою присутність. Вважається нормальним, якщо широкомовні пакети становлять не більше 10% від загального числа пакетів у мережі у корпоративному центрі управління інформаційною безпекою (SOC).

Також досить часто до шторму приводять кільця в мережі у корпоративному центрі управління інформаційною безпекою (SOC) при некоректному настроюванні протоколу Spanning Tree, оскільки в заголовку пакетів Ethernet немає інформації про час життя кадру, як, наприклад, у пакетів IP. Крім цього широкомовний шторм застосовується (навмисно) зломщиками.

Відповідно до галузевого стандарту де-факто число широкомовних і багатоадресних кадрів у мережі у корпоративному центрі управління інформаційною безпекою (SOC) не повинне перевищувати 8-10% від загального числа кадрів.

Широкомовний кадр – це кадр, адресований всім станціям у домені мережі у корпоративному центрі управління інформаційною безпекою (SOC). Багатоадресний кадр – це кадр, адресований групі станцій у домені мережі у корпоративному центрі управління інформаційною безпекою (SOC). Оскільки широкомовний кадр адресований всім станціям, то, одержавши його, станції повинні перервати свою роботу й обробити такий кадр. Це сповільнює роботу всієї мережі у корпоративному центрі управління інформаційною безпекою (SOC).

Якщо відношення числа широкомовних кадрів до загального числа кадрів більше 10%, то такий ефект називається "широкомовним штормом".

Широкомовний шторм може бути наслідком дефектів устаткування або неправильного настроювання параметрів активного встаткування. Найчастіше це явище спостерігається в розподілених мережах NetWare, побудованих на основі комутаторів, або коли дані між сегментами або доменами мережі у корпоративному центрі управління інформаційною безпекою (SOC) можуть передаватися більш ніж по одному потенційному шляху. Якщо один з комутаторів такої мережі у корпоративному центрі управління інформаційною безпекою (SOC) не підтримує протокол Spanning Tree (звичайно IEEE 802.1d) або останній неправильно настроєний або збоїть, то в мережі у корпоративному центрі управління інформаційною безпекою (SOC) починається некерована циркуляція широкомовних кадрів.

Виявлення "широкомовного шторму" є не настільки тривіальним завданням, як це може здатися на перший погляд. Для його виявлення недостатньо взяти загальне число широкомовних кадрів і поділити його на загальне число кадрів, що пройшли по мережі у корпоративному центрі управління інформаційною безпекою (SOC).

Для цього ви повинні визначити: яку частку становлять широкомовні кадри в кожний інтервал часу (наприклад, за одну мінуту) і яка при цьому утилізація каналу зв'язку. Якщо, наприклад, за одну мінуту по мережі у корпоративному центрі управління інформаційною безпекою (SOC) пройшло 4

кадри, а 2 з них були широкомовними, то це ще не виходить, що ви спостерігаєте "широкомовний шторм".

Захист від широкомовних штормів (broadcast storm)

Одна з характерних несправностей мережного програмного забезпечення – мимовільна генерація з високою інтенсивністю широкомовних пакетів. Широкомовним штормом вважається ситуація, у якій відсоток широкомовних пакетів перевищує 20% від загальної кількості пакетів у мережі у корпоративному центрі управління інформаційною безпекою (SOC). Звичайний комутатор або міст сліпо передає такі пакети на всі свої порти, як того вимагає його логіка роботи, засмічуючи, таким чином, мережу. Боротьба із широкомовним штормом у мережі у корпоративному центрі управління інформаційною безпекою (SOC), з'єднаної комутаторами, жадає від адміністратора відключення портів, що генерують широкомовні пакети. Маршрутизатор не поширює такі ушкоджені пакети, оскільки в коло його завдань не входить копіювання широкомовних пакетів в усі поєднувані їм мережі у корпоративному центрі управління інформаційною безпекою (SOC). Тому маршрутизатор є прекрасним засобом боротьби із широкомовним штормом, щоправда, якщо мережа розділена на достатню кількість підмереж.

ARP-spoofing

ARP-spoofing – техніка атаки у корпоративному центрі управління інформаційною безпекою (SOC) в Ethernet мережах, що дозволяє перехоплювати трафік між хостами. Заснована на використанні протоколу ARP.

При використанні в розподіленій обчислювальній системи (РВМ) алгоритмів віддаленого пошуку існує можливість здійснення в такій мережі у корпоративному центрі управління інформаційною безпекою (SOC) типової віддаленої атаки у корпоративному центрі управління інформаційною безпекою (SOC) «помилковий об'єкт РВМ». Аналіз безпеки протоколу ARP показує, що, перехопивши на атакуючому хості усередині даного сегмента мережі у корпоративному центрі управління інформаційною безпекою (SOC) широкомовний ARP-запит, можна послати помилкову ARP-відповідь, у якій

оголосити себе шуканим хостом (наприклад, маршрутизатором), і надалі активно контролювати мережний трафік дезінформованного хосту, впливаючи на нього за схемою «помилковий об'єкт РВМ».

Протокол ARP призначений для перетворення IP-адрес в MAC-адреси. Найчастіше мова йде перетворенні в адреси Ethernet, але ARP використовується й у мережах інших технологій: Token Ring, FDDI і інших.

Алгоритм роботи ARP

Протокол може використовуватися в наступних випадках:

1. Хост А хоче передати IP-пакет вузлу В, що перебуває з ним в одній мережі у корпоративному центрі управління інформаційною безпекою (SOC).

2. Хост А хоче передати IP-пакет вузлу В, що перебуває з ним у різних мережах, і користується для цього послугами маршрутизатора R.

У кожному із цих випадку вузлом А буде використовуватися протокол ARP, тільки в першому випадку для визначення MAC-адреси вузла В, а в другому – для визначення MAC-адреси маршрутизатора R. В останньому випадку пакет буде переданий маршрутизатору для подальшої ретрансляції.

Далі для простоти розглядається перший випадок, коли інформацією обмінюються вузли, що перебувають безпосередньо в одній мережі у корпоративному центрі управління інформаційною безпекою (SOC). (Випадок коли пакет адресований вузлу, який знаходиться за маршрутизатором, відрізняється тільки тим, що в пакетах переданих після того як ARP-перетворення завершене, використовується IP-адреса одержувача, але MAC-адреса маршрутизатора, а не одержувача.)

Проблеми ARP

Протокол ARP є абсолютно незахищеним. Він не має ніякого способу перевірки дійсності пакетів: як запитів, так і відповідей. Ситуація стає ще більш складною, коли може використовуватися мимовільний ARP (gratuitous ARP).

Мимовільний ARP – таке поводження ARP, коли ARP-відповідь надсилається, коли в цьому (з погляду одержувача) немає особою необхідності. Мимовільна ARP-відповідь це пакет-відповідь ARP, присланий без запиту. Він

застосовується для визначення конфліктів IP-адрес у мережі у корпоративному центрі управління інформаційною безпекою (SOC): як тільки станція одержує адресу по DHCP або адреса привласнюється вручну, розсилається ARP-відповідь gratuitous ARP.

Мимовільний ARP може бути корисний у наступних випадках:

– Відновлення ARP-таблиць, зокрема, у кластерних системах.

– Інформування комутаторів.

– Повідомлення про включення мережного інтерфейсу.

Незважаючи на ефективність мимовільного ARP, він є особливо небезпечним, оскільки з його допомогою можна запевнити віддалений вузол у тому, що MAC-адреса якої-небудь системи, що перебуває з нею в одній мережі у корпоративному центрі управління інформаційною безпекою (SOC), змінилася й указати, яка адреса використовується тепер.

До виконання ARP-spoofing'a в ARP-таблиці вузлів A і B існують записи з IP- і MAC-адресами один одного. Обмін інформацією виробляється безпосередньо між вузлами A і B.

У ході виконання ARP-spoofing'a комп'ютер **C**, що виконує атаку, відправляє ARP-відповіді (без одержання запитів):

– вузлу A: з IP-адресою вузла B і MAC-адресою вузла C;

– вузлу B: з IP-адресою вузла A і MAC-адресою вузла C.

У силу того що комп'ютери підтримують мимовільний ARP (gratuitous ARP), вони модифікують власні ARP-таблиці й поміщають туди записи, де замість справжніх MAC-адрес комп'ютерів A і B коштує MAC-адреса комп'ютера C.

Після того як атака виконана, коли комп'ютер A хоче передати пакет комп'ютеру B, він знаходить в ARP-таблиці запис (він відповідає комп'ютеру C) і визначає з її MAC-адресу одержувача. Відправлений по цьому MAC-адресу пакет приходить комп'ютеру C замість одержувача. Комп'ютер C потім ретранслює пакет тому, кому він дійсно адресований – тобто комп'ютеру B.

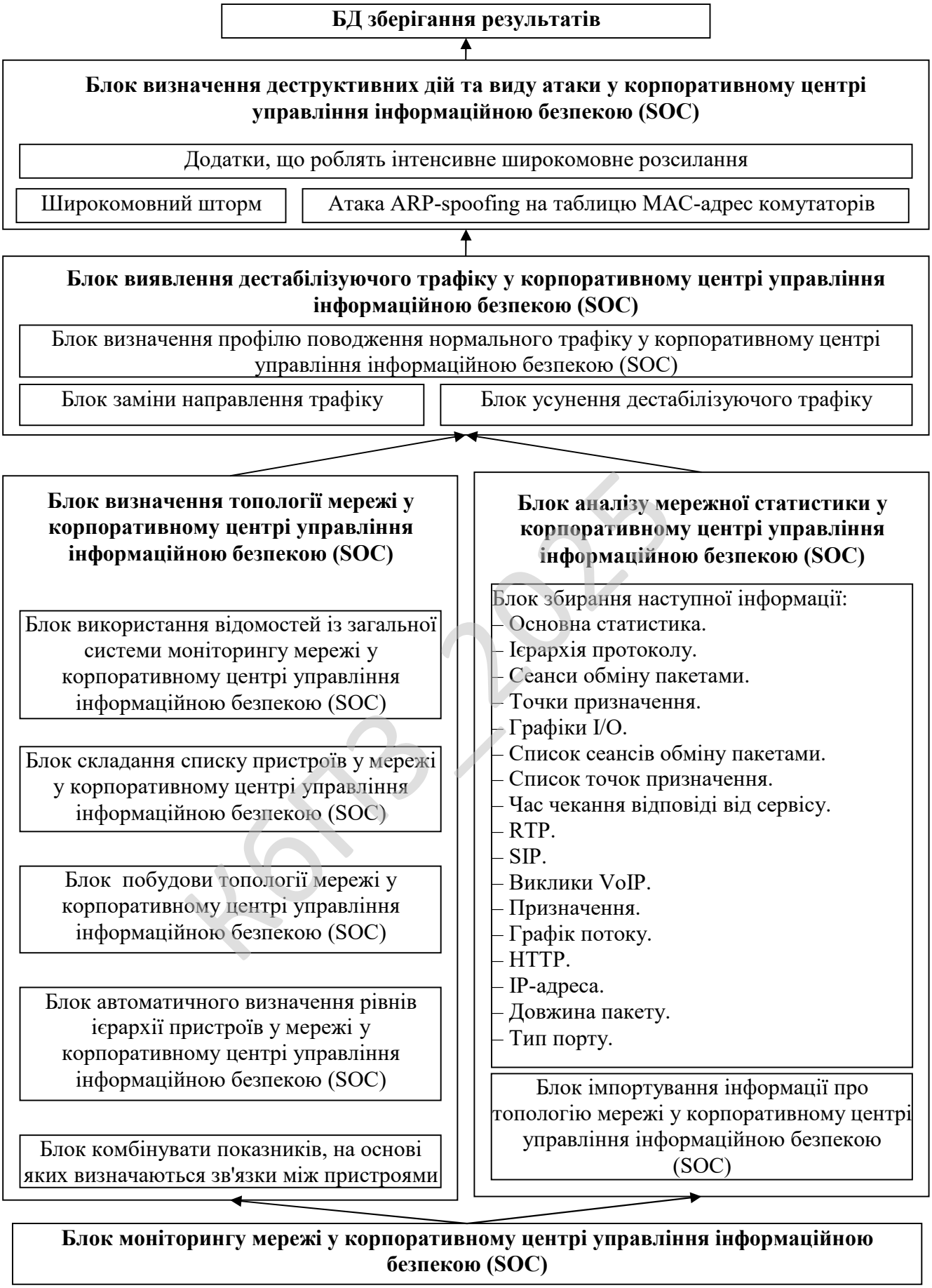

Рисунок 3.2 – Функціональна схема системи

Блок аналізу мережної статистики у корпоративному центрі управління інформаційною безпекою (SOC)

Блок збирання наступної інформації:

– Основна статистика (Summary).

– Ієрархія протоколу (Protocol Hierachy).

– Сеанси обміну пакетами (Conversations).

– Точки призначення (Endpoints).

– Графіки I/O (IO Graphs).

– Список сеансів обміну пакетами (Conversation List).

– Список точок призначення (Endpoint List).

– Час чекання відповіді від сервісу (Service Response Time).

– RTP.

– SIP.

– Виклики VoIP (VoIP Calls).

– Призначення (Destination).

– Графік потоку (Flow Graph).

– HTTP.

– IP-адреса (IP address).

– Довжина пакету (Packet Length).

– Тип порту (Port Type).

Розпишемо їх більш детально.

1. Основна статистика. Доступні такі елементи основної статистики у корпоративному центрі управління інформаційною безпекою (SOC), як:

– Властивості захоплених файлів.

– Час захвату.

– Інформація про фільтр захвату.

– Інформація про фільтр відображення.

2. Ієрархія протоколу. Статистика ієрархії протоколу допомагає аналізувати пакети, розбиваючи відображені дані, які належать чинному рівню OSI.

3. Сеанси обміну пакетами. Якщо ви використовуєте протокол TCP/IP або програму, яка працює із цим протоколом, ви маєте побачити чотири активних вкладок для обміну пакетами за допомогою Ethernet, IP, TCP та UDP. «Діалог» між комп'ютерами відображає трафік між двома активними хостами. Номер, зазначений на вкладці після назви протоколу, означає кількість «діалогів» між хостами. Номер, зазначений на вкладці після назви протоколу, означає кількість «діалогів» між хостами, наприклад, «Ethernet:6».

4. Точки призначення. Точки призначення забезпечують статистику даними про відправку та прийом пакетів. Номер, зазначений на вкладці після назви протоколу, вказує на кількість точок призначення. Наприклад, «Ethernet:6».

5. Графіки I/O. Основний графік може бути отриманий за допомогою команди «IO graphs» (Графіки I/O). Ще декілька графіків можуть бути додані у тому ж вікні на основі фільтрів відображення.

6. Час чекання відповіді від сервісу. 13 протоколів доступні для глибокого аналізу.

7. RTP. RTP (Real-time Transport Protocol, протокол передачі у реальному часі, RFC 3550) – це протокол для передачі звука та відео через IP-мережу. Він працює у початку протоколу дейтаграм користувача (User Datagram Protocol, UDP). Він часто використовується у сукупності з протоколами SIP або H.233, забезпечуючи виконання сигнальних завдань.

8. SIP. SIP (Session Initiation Protocol, протокол встановлення сесії, RFC 3261) – це сигнальний протокол, який оголошує відео– або VoIP-сесії. Він працює разом із протоколом RTP, який використовується для передачі мультимедійних даних.

9. Виклики VoIP.

VoIP (Voice over IP, голосовий зв'язок за допомогою Інтернету) взагалі використовує два типи протоколів:

– сигнальні протоколи, такі, як SIP або H.323

– переносні протоколи, наприклад, RTP

10 Призначення. Відображення усіх IP-адрес призначення мережевих пакетів.

11. Графік потоків. Графіки потоків забезпечує послідовний аналіз TCP-з'єднань. Перші три строки містять оголошення TCP– з'єднання з послідовностями «SYN», «SYN ACK» та «ACK».

12 HTTP. HTTP (Hypertext Transfer Protocol, протокол передачі гіпертексту) – це протокол типу «клієнт-сервер», який використовується для передачі HTML-файлів. HTTP-клієнт (у більшості випадків це web-браузер) відсилає HTTP-запит до web-серверу із полем «URL», який допомагає знайти потрібний файл. Web-сервер відповідає HTTP-пакетом та забезпечує клієнт необхідною web-сторінкою.

Меню «HTTP» містить три підменю:

– «Load Distribution» (Розподіл пакетів).

– «Packet Counter» (Лічильник пакетів).

– «Requests» (Запити).

14 IP-адреса. Відображення IP-адреси джерела або призначення мережевих пакетів.

15. Довжина пакету.

16. Тип порту. Відображення статистики у корпоративному центрі управління інформаційною безпекою (SOC) портів TCP або UDP.

Блок визначення топології мережі у корпоративному центрі управління інформаційною безпекою (SOC)

Блок визначення топології мережі у корпоративному центрі управління інформаційною безпекою (SOC) включає в себе наступні блоки:

– Блок використання відомостей із загальної системи моніторингу мережі у корпоративному центрі управління інформаційною безпекою (SOC), а не опитування пристрою додатково.

– Блок складання списку пристроїв у мережі у корпоративному центрі управління інформаційною безпекою (SOC), автоматично, ґрунтуючись на дані системи моніторингу.

– Блок побудови топології мережі у корпоративному центрі управління інформаційною безпекою (SOC), за станом на задану дату й відстеження змін у топології протягом часу.

– Блок автоматичного визначення рівнів ієрархії пристроїв у мережі у корпоративному центрі управління інформаційною безпекою (SOC), з виділенням периферійних, проміжних і центральних вузлів.

– Блок побудови топології мережі у корпоративному центрі управління інформаційною безпекою (SOC), незалежно від використовуваної системи моніторингу й програмно-апаратних платформ;

– Блок комбінувати показників, на основі яких визначаються зв'язки між пристроями, і при їхньому обчисленні виконувати перевірку на значимість із використанням статистичних критеріїв.

Розглянувши усі блоки функціональної схеми перейдемо до розгляду діаграми взаємодії процесів, які відбуваються у системі.

3.4 Розробка діаграми процесів

Діаграма процесів розробленої системи зображена на рисунку 3.3. При детальному її розгляді можна побачити як саме проходе взаємодія у розробленій системі. Використовується модель проектування, графічне представлення «потоків» даних в інформаційній системі.

Діаграма взаємодії процесів використовується для візуалізації процесів обробки даних (структурне проектування). Для розробника вважається звичним спочатку креслити діаграму взаємодії процесів даних рівня контексту, завдяки чому буде показано взаємодію системи. Ця діаграма в подальшому підлягає

уточненню шляхом деталізації процесів та потоків даних з метою показати систему що розробляється.

Діаграми потоків даних містять чотири типи елементів:

– Процеси які являють собою трансформацію даних в рамках описуваної системи.

– Сховища даних (репозиторії).

– Зовнішні по відношенню до системи сутності.

– Потоки даних між елементами трьох попередніх типів.

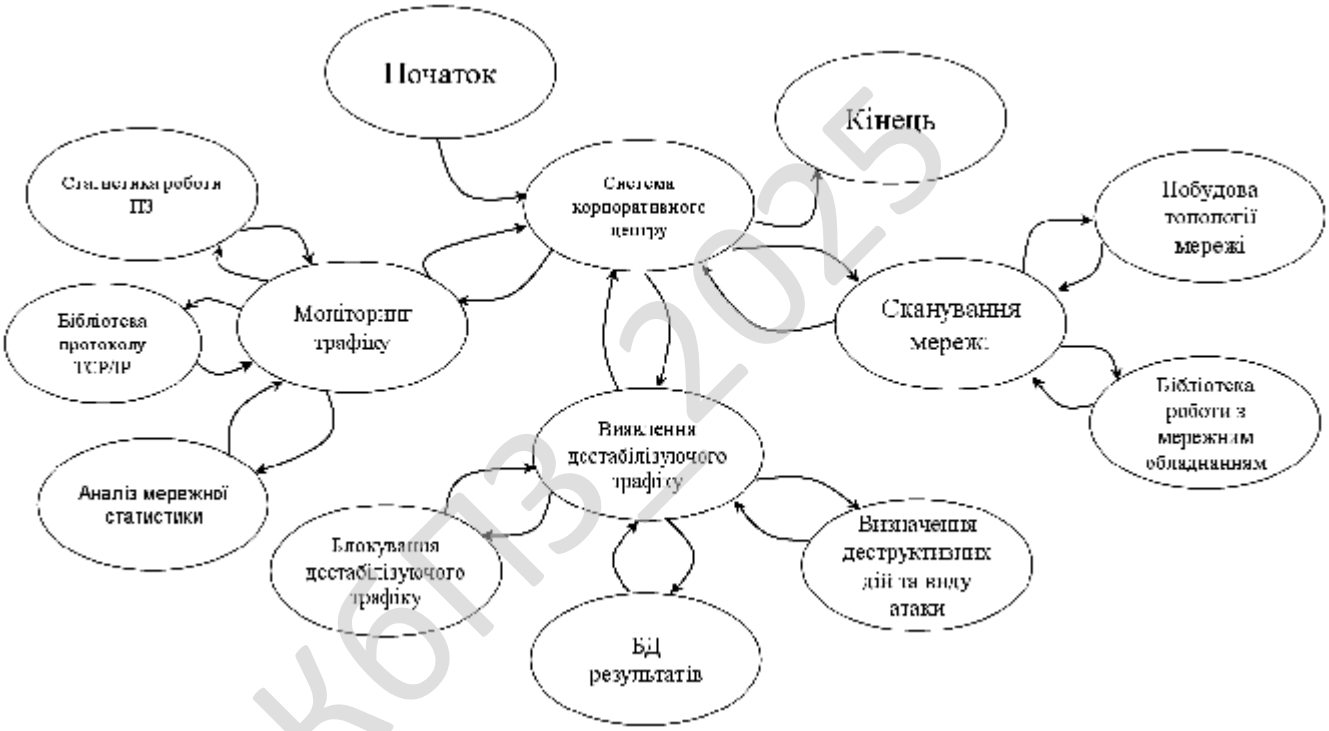

Рисунок 3.3 – Діаграма взаємодії процесів

Таким чином, розглянувши опис системи, структурну, функціональну схеми системи, та діаграму взаємодії процесів перейдемо до опису блок-схем основної програми, та підпрограм, які використовуються, для реалізації системи.

4 РЕАЛІЗАЦІЯ ПРОЕКТУ. РОЗРАХУНКИ І ЕКСПЕРИМЕНТАЛЬНІ ДАНІ, ЩО ПІДТВЕРДЖУЮТЬ ПРАВИЛЬНІСТЬ ПРОЕКТНИХ РІШЕНЬ

4.1 Блок-схеми та опис алгоритмів функціонування системи

Під час роботи над магістерською дипломною роботою було створено блок-схеми. Перед їх розглядом необхідно провести роз’яснення який саме тип блок-схем використовується.

Блок-схема це представлення задачі для її аналізу або розв'язування за допомогою спеціальних символів (геометричних образів), які позначають такі елементи, як операції, потік, дані тощо. Блок вхідних та вихідних даних прийнято позначати паралелограмом, блок обчислень (обробки) даних – прямокутником, блок прийняття рішень – ромбом, еліпсом – початок та кінець алгоритму.

У інформаційних технологіях функціональна схема складається з функціональних блоків, які являють собою конструктивно відособлені частини (елементи або пристрої) автоматичних систем, які виконують певні функції. Функціональні блоки на схемі позначають прямокутниками, всередині яких надписують їх найменування відповідно до функцій, що виконуються. Зв'язки між функціональними блоками (внутрішні впливи) позначаються лініями зі стрілками, які вказують напрям впливів.

Функціональні схеми можуть виконуватися в укрупненому і розгорненому вигляді. У першому випадку на схемі зображають найважливіші блоки системи і зв'язки між ними.

У другому варіанті схема відображається більш детально, що полегшує її читання та ілюструє принцип роботи.

Основні елементи схем алгоритму це термінатор, процес, рішення, зумовлений процес (підпрограма), дані та з'єднувач.

Термінатор це елемент відображає вхід із зовнішнього середовища або вихід з неї (найчастіше застосування – початок і кінець програми). Всередині фігури записується відповідна дія.

Процес це виконання однієї або кількох операцій, обробка даних будь-якого виду (зміна значення даних, форми подання, розташування). Всередині фігури записують безпосередньо самі операції.

Рішення це показує рішення або функцію перемикального типу з одним входом і двома або більше альтернативними виходами, з яких тільки один може бути обраний після обчислення умов, визначених всередині цього елементу.

Вхід в елемент позначається лінією, що входить зазвичай у верхню вершину елементу. Якщо виходів два чи три то зазвичай кожен вихід позначається лінією, що виходить з решти вершин (бічних і нижній).

Якщо виходів більше трьох, то їх слід показувати однією лінією, що виходить з вершини (частіше нижній) елемента, яка потім розгалужується. Відповідні результати обчислень можуть записуватися поруч з лініями, що відображають ці шляхи.

Зумовлений процес (підпрограма) це символ відображає виконання процесу, що складається з однієї або кількох операцій, що визначені в іншому місці програми (у підпрограмі, модулі). Всередині символу записується назва процесу і передані в нього дані.

Дані це перетворення у форму, придатну для обробки (введення) або відображення результатів обробки (виведення).

Цей символ не визначає носія даних (для вказівки типу носія даних використовуються специфічні символи).

З'єднувач це символ відображає вихід в частину схеми і вхід з іншої частини цієї схеми. Використовується для обриву лінії та продовження її в іншому місці (приклад: поділ блок-схеми, що не поміщається на листі).

Відповідні сполучні символи повинні мати одне (при тому унікальне) позначення.

Блок-схеми є першоджерелами стратегії розвитку ПЗ. Тому від точності і детальної блок-схеми залежить результат всієї програми.

При виборі початкової точки відліку при побудові схем було враховано, що виходячи з вибору мови програмування і інших технічних засобів, програма буде об’єктно-орієнтована що вимагає оптимізації програми високого рівня, також те, що при розробці програми слід надати особливу увагу модулю корпоративного центру управління інформаційною безпекою (SOC).

На рисунку 4.1 зображена основна блок-схема програми, на рисунку 4.2 зображено роботу підпрограми.

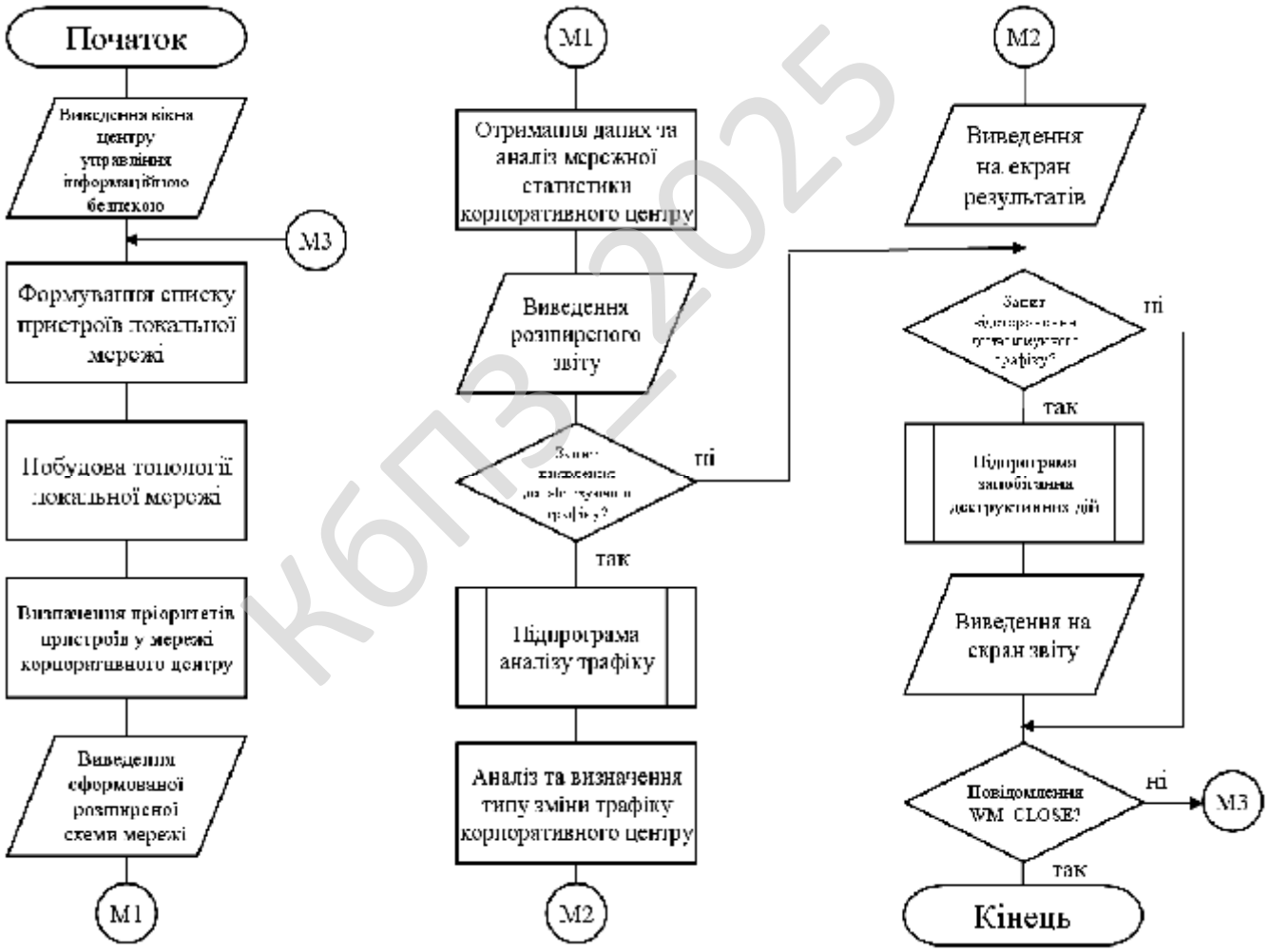

Рисунок 4.1 – Блок-схема основної програми

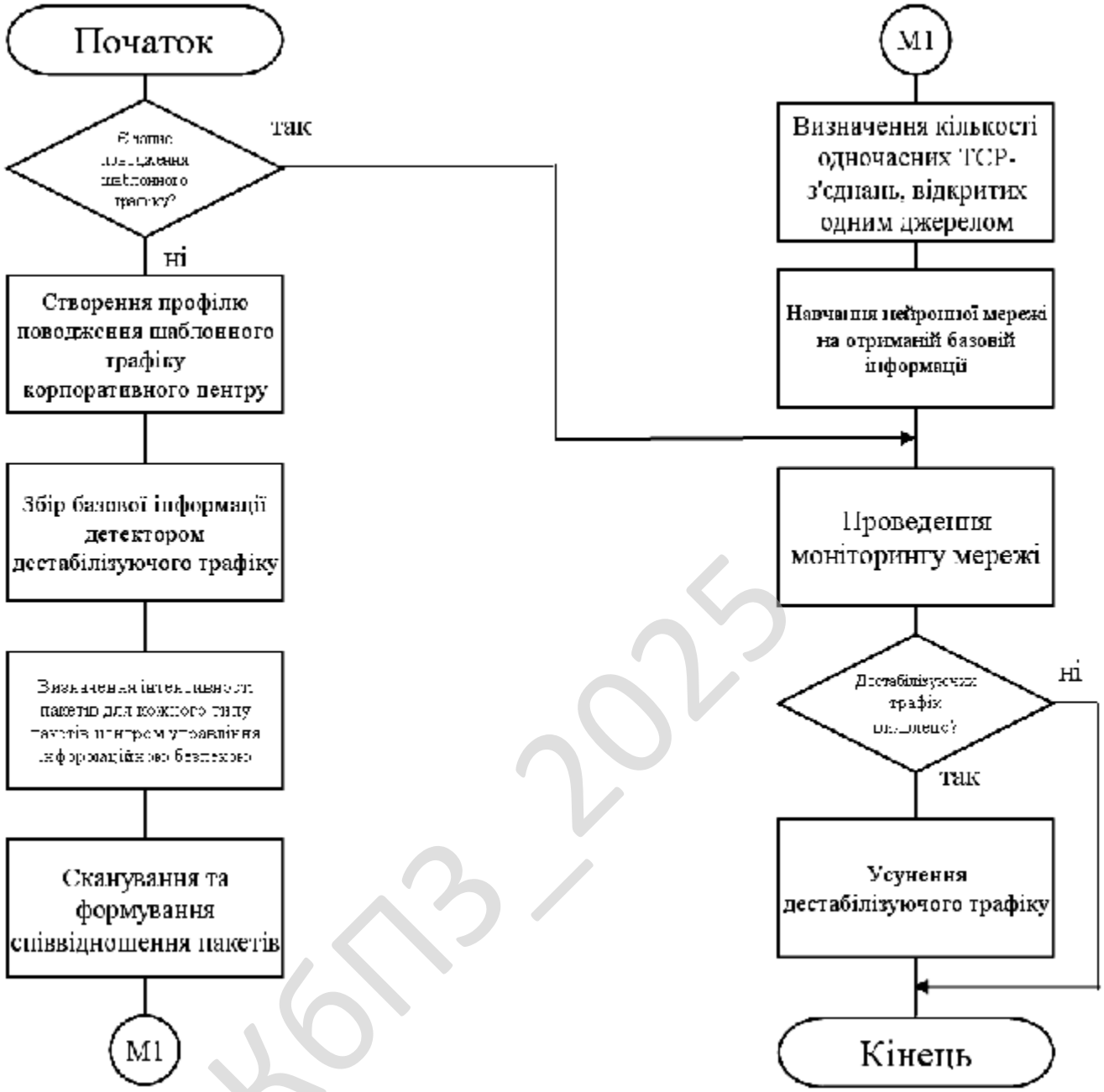

Рисунок 4.2 – Блок-схема роботи підпрограми

З яких видно що робота основної програми складається з початкових етапів ініціалізації ПЗ, перевірки наявності ресурсів системи, блоку початку основного циклу з чеканням запиту від користувача в якому відбувається виклик підпрограми та останньої стадії – перевірка поточного стану з завершенням роботи розробленого ПЗ. При роботі підпрограми виконується основний

функціонал системи з циклічними послідовностями, перевіркою поточного стану та поверненням в основну програму прапорів стану виконання.

Було використано підходи з використанням UML, це уніфікована мова моделювання, використовується у парадигмі об'єктно-орієнтованого програмування. Є невід'ємною частиною уніфікованого процесу розробки програмного забезпечення. UML є мовою широкого профілю, це відкритий стандарт, що використовує графічні позначення для створення абстрактної моделі системи, називаної UML-моделлю. UML був створений для визначення, візуалізації, проектування й документування в основному програмних систем. UML не є мовою програмування, але в засобах виконання UML-моделей як інтерпретованого коду можлива кодогенерація.

UML може бути застосовано на всіх етапах життєвого циклу аналізу бізнес-систем і розробки прикладних програм. Різні види діаграм які підтримуються UML, і найбагатший набір можливостей представлення певних аспектів системи робить UML універсальним засобом опису як програмних, так і ділових систем.

Діаграми дають можливість представити систему (як ділову, так і програмну) у такому вигляді, щоб її можна було легко перевести в програмний код. Основною причиною використання мови UML є спілкування розробників між собою.

Крім того, UML спеціально створювалася для оптимізації процесу розробки програмних систем, що дозволяє збільшити ефективність їх реалізації у кілька разів і помітно поліпшити якість кінцевого продукту.

UML прекрасно зарекомендувала себе в багатьох успішних програмних проектах. Засоби автоматичної генерації кодів дозволяють перетворювати моделі мовою UML у вихідний код об’єктно-орієнтованих мов програмування, що ще більш прискорює процес розробки. Практично усі CASE-засоби (програми автоматизації процесу аналізу і проектування) мають підтримку UML. Моделі

розроблені в UML, дозволяють значно спростити процес кодування і направити зусилля програмістів безпосередньо на реалізацію системи.

Діаграми підвищують супроводжуваність проекту і полегшують розробку документації.

UML необхідний:

– Керівникам проектів, які керують розподілом завдань і контролем за проектом.

– Проектувальникам інформаційних систем які розробляють технічні завдання для програмістів.

– Бізнес-аналітикам, які досліджують реальну систему і здійснюють інжиніринг і реінжиніринг бізнесу компанії.

– Програмістам які реалізовують модулі інформаційної системи.

При модифікації системи об'єктний підхід дозволяє легко включати в систему нові об'єкти і виключати застарілі без істотної зміни її життєздатності. Використання побудованої моделі при модифікаціях системи дає можливість усунути небажані наслідки змін, оскільки вони не ламають структури системи, а тільки змінюють поведінку об'єктів.

Архітектура клієнт-сервер є одним із архітектурних шаблонів програмного забезпечення та є домінуючою концепцією у створенні розподілених мережних програм і передбачає взаємодію та обмін даними між ними. Вона передбачає такі основні компоненти:

– набір серверів, які надають інформацію або інші послуги програмам, які звертаються до них;

– набір клієнтів, які використовують сервіси, що надаються серверами;

– мережа, яка забезпечує взаємодію між клієнтами та серверами.

Сервери є незалежними один від одного. Клієнти також функціонують паралельно і незалежно один від одного. Немає жорсткої прив'язки клієнтів до серверів. Більш ніж типовою є ситуація, коли один сервер одночасно обробляє запити від різних клієнтів; з іншого боку, клієнт може звертатися то до одного

сервера, то до іншого. Клієнти мають знати про доступні сервери, але можуть не мати жодного уявлення про існування інших клієнтів.

Дуже важливо ясно уявляти, хто або що розглядається як «клієнт». Можна говорити про клієнтський комп'ютер, з якого відбувається звернення до інших комп'ютерів. Можна говорити про клієнтське та серверне програмне забезпечення. Нарешті, можна говорити про людей, які бажають за допомогою відповідного програмного та апаратного забезпечення отримати доступ до тієї чи іншої інформації.

Загальноприйнятим є положення, що клієнти та сервери – це перш за все програмні модулі. Найчастіше вони знаходяться на різних комп'ютерах, але бувають ситуації, коли обидві програми – і клієнтська, і серверна, фізично розміщуються на одній машині; в такій ситуації сервер часто називається локальним.

Модель клієнт-серверної взаємодії визначається перш за все розподілом обов'язків між клієнтом та сервером. Логічно можна відокремити три рівні операцій:

– рівень представлення даних, який по суті являє собою інтерфейс користувача і відповідає за представлення даних користувачеві і введення від нього керуючих команд;

– прикладний рівень, який реалізує основну логіку ПЗ і на якому здійснюється необхідна обробка інформації;

– рівень управління даними, який забезпечує зберігання даних та доступ до них.

Дворівнева клієнт-серверна архітектура передбачає взаємодію двох програмних модулів – клієнтського та серверного. В залежності від того, як між ними розподіляються наведені вище функції, розрізняють:

– модель тонкого клієнта, в рамках якої вся логіка ПЗ та управління даними зосереджена на сервері. Клієнтська програма забезпечує тільки функції рівня представлення;

– модель товстого клієнта, в якій сервер тільки керує даними, а обробка інформації та інтерфейс користувача зосереджені на стороні клієнта. Товстими клієнтами часто також називають пристрої з обмеженою потужністю: кишенькові комп'ютери, мобільні телефони та ін.

Типовим прикладом клієнт-серверної взаємодії є WWW. Існує величезна кількість веб-серверів, на яких розміщується та чи інша інформація. У найпростішому випадку ця інформація являє собою набір веб-сторінок, які можуть зберігатися на сервері у вигляді файлів, розмічених за допомогою мови розмітки HTML. Але ситуація, як правило, є складнішою; значна частина веб-ресурсів на сучасному етапі є динамічними, тобто вони не існують в заздалегідь підготовленому вигляді, а створюються безпосередньо в процесі обробки запиту від користувача.

Для того, щоб людина, яка працює в Інтернеті, могла переглянути ту чи іншу сторінку, на її комп'ютері повинно бути встановлено відповідне програмне забезпечення. Програми для перегляду веб-сторінок називаються браузерами.

Але, крім браузерів, до серверів можуть звертатися і інші клієнти, а саме – автономні програми. Вони можуть передбачати взаємодію з людиною, а можуть працювати в цілком автоматичному режимі. Типовим класом таких програм є роботи, призначені для автоматичного перегляду веб-ресурсів. Зокрема, роботи є важливим елементом пошукових систем і використовуються ними для перегляду сторінок і збору інформації про них.

Для запиту до веб-сервера клієнтська програма повинна задати місцезнаходження комп'ютера, на якому розміщується серверна програма, назву потрібного документа і, можливо, інші дані, які специфікують запит. Мережа забезпечує знаходження сервера і передачу йому клієнтського запиту. Серверні програми обробляють цей запит, відповідь пересилається по мережі клієнтові.

Трирівнева клієнт-серверна архітектура, яка почала розвиватися з середини 90-х років, передбачає відділення прикладного рівня від управління даними. Відокремлюється окремий програмний рівень, на якому зосереджується

прикладна логіка ПЗ. Програми проміжного рівня можуть функціонувати під управлінням спеціальних серверів ПЗ, але запуск таких програм може здійснюватися і під управлінням звичайного веб-сервера. Нарешті, управління даними здійснюється сервером даних.

Для роботи з системою користувач використовує стандартне програмне забезпечення –звичайний браузер. Це позбавляє його необхідності завантажувати та інсталювати спеціальні програми (хоча інколи така необхідність все-таки виникає).

Але користувачеві слід надати в розпорядженні інтерфейс, який дозволяв би йому взаємодіяти з системою і формувати запити до неї. Форми, що визначають цей інтерфейс, розміщуються на веб-сторінках та завантажуються разом з ними.

Веб-оглядач формує запит та пересилає його до сервера, який здійснює обробку. При необхідності сервер викликає серверні програмні модулі, які забезпечують обробку запиту і в разі потреби звертаються до сервера даних. Сервер даних здійснює операції з даними, що зберігаються в системі та складають її інформаційну основу. Зокрема, він може здійснити вибірку з інформаційної бази відповідно до запиту та передати її модулю проміжного рівня для подальшої обробки. Дані, з якими працює сервер даних, найчастіше організовані як реляційна база даних.

Найчастіше веб-сервер і серверні модулі проміжного рівня розміщуються на одному комп'ютері, хоч і являють собою окремі і логічно незалежні програмні модулі.

На сучасному етапі для програмування модулів проміжного рівня використовується мова серверних сценаріїв PHP, а для управління даними – СУБД MySQL. Таким чином, зв'язку PHP-MySQL слід розглядати як стандартний інструмент для створення порівняно простих інтерактивних веб-сайтів та систем електронної комерції; близько 90% комерційних систем сьогодні створюється саме на цій основі. Водночас як засоби управління даними, так і middleware-

засоби можуть бути найрізноманітнішими. Так, для створення серверних програм, крім PHP, широко застосовуються Java, Perl, Python, Delphi.

Взагалі, технології створення розподілених, зокрема веб-програм, стрімко розвиваються. Слід згадати про технології EJB (Enterprise Java Beans), CORBA, а також про .NET – порівняно нову ініціативу компанії Microsoft. Для зберігання даних та їх передачі часто використовується так звана розширювана мова розмітки XML (Extensible Markup Language).

Незважаючи на те що я працював над ПЗ один в реалізації програми я використовував підходи пришвидшення розробки на основі методологій Agile.

Гнучка́ розробка програмного забезпечення (Agile software development, agile-методи) – клас методологій розробки програмного забезпечення, що базується на ітеративній розробці, в якій вимоги та розв'язки еволюціонують через співпрацю між самоорганізовуваними багатофункціональними командами.

Гнучка розробка – найкращий засіб для підвищення продуктивності розробників програмного забезпечення.

Більшість гнучких методологій націлені на мінімізацію ризиків, шляхом зведення розробки до серії коротких циклів, що мають назву ітерацій, які зазвичай тривають один-два тижні. Кожна ітерація сама по собі виглядає як програмний проект в мініатюрі, і включає всі завдання, необхідні для видачі мінімального приросту за функціональністю: планування, аналіз вимог, проектування, кодування, тестування і документування. Хоча окрема ітерація, як правило, недостатня для випуску нової версії продукту, мається на увазі те, що гнучкий програмний проект готовий до випуску наприкінці кожної ітерації. Після закінчення кожної ітерації, команда виконує переоцінку пріоритетів розробки.

Agile акцентує увагу на безпосередньому спілкуванні «віч-на-віч». Більшість agile команд розташовані в одному офісі, його іноді називають bullpen. Як мінімум вона включає і «замовників» (замовники, які визначають продукт, також це можуть бути менеджери продукту, бізнес аналітики або клієнти). Офіс

може також включати тестувальників, дизайнерів інтерфейсу, технічних авторів і менеджерів.

Основною метрикою agile методів є робочий продукт. Віддаючи перевагу безпосередньому спілкуванню, agile-методи зменшують обсяг письмової документації в порівнянні з іншими методами. Це привело до критики цих методів як недисциплінованих.

Agile – родина процесів розробки, а не єдиний підхід в розробці програмного забезпечення, і визначається Agile Manifesto. Agile не включає практик, а визначає цінності та принципи, якими керуються успішні команди.

Agile Manifesto розроблений і прийнятий 17 розробниками 11-13 лютого 2001 року на лижному курорті The Lodge at Snowbird в горах Юти. Маніфест підписали представники наступних методологій Extreme programming, Scrum, DSDM, Adaptive software development, Crystal Clear, Feature driven development, Pragmatic Programming. Agile Manifesto містить 4 основні ідеї та 12 принципів. Примітно, що Agile Manifesto не містить практичних порад.

Основні ідеї:

– Особистості та їхні взаємодії важливіші, ніж процеси та інструменти;

– Робоче програмне забезпечення важливіше, ніж повна документація;

– Співпраця із замовником важливіша, ніж контрактні зобов'язання;

– Реакція на зміни важливіша, ніж дотримання плану.

Принципи, які роз'яснює Agile Manifesto:

– задоволення клієнта за рахунок ранньої та безперебійної поставки коштовного програмного забезпечення;

– вітання змін вимог навіть наприкінці розробки (це може підвищити конкурентоспроможність отриманого продукту);

– часта поставка робочого програмного забезпечення (кожен місяць або тиждень або ще частіше);

– тісне, щоденне спілкування замовника з розробниками впродовж всього проекту;

– проектом займаються мотивовані особистості, які забезпечені потрібними умовами роботи, підтримкою і довірою;

– рекомендований метод передачі інформації – особиста розмова (віч-на-віч);

– робоче програмне забезпечення – найкращий вимірювач прогресу;

– спонсори, розробники та користувачі повинні мати можливість підтримувати постійний темп на невизначений термін;

– постійну увагу поліпшенню технічної майстерності та зручному дизайну;

– простота – мистецтво не робити зайвої роботи;

– найкращі технічні вимоги, дизайн та архітектура виходять у самоорганізованої команди;

– постійна адаптація до мінливих обставин.

Маніфест та Принципи гнучкої розробки містять високорівневі ідеї щодо того, як потрібно вибудовувати процес розробки програмного забезпечення, щоб успішно завершувати проекти й створювати команди, в яких приємно та цікаво працювати.

Документи визначають, що потрібно для цього зробити, але не говорять, як це зробити. По-іншому й не могло бути, оскільки Маніфест та Принципи народилися внаслідок консенсусу представників різних (хоча й споріднених) напрямів, які могли знайти спільну основу лише на рівні базових цінностей та принципів.

Критика. Багато керівників проектів, що працюють у традиційних методологіях на кшталт «водоспаду», критикують agile-методи.

Один з повторюваних пунктів критики: при agile-підході часто нехтують створенням «дорожньої карти» розвитку продукту, так само як і управлінням вимогами, в процесі якого і формується така «карта». Гнучкий підхід до управління вимогами не має на увазі далекосяжних планів (по суті, управління вимогами просто не існує в даній методології), а має на увазі можливість замовника раптом і несподівано наприкінці кожної ітерації виставляти нові

вимоги, що часто суперечать архітектурі вже створеного і поставленого продукту. Таке іноді призводить до катастрофічних «авралів» з масовим рефакторингом і переробками практично на кожній черговій ітерації.

Крім того вважається, що робота в agile мотивує розробників вирішувати всі прибулі завдання найпростішим і найшвидшим можливим способом, при цьому часто не звертаючи уваги на коректність коду з точки зору вимог базової платформи (підхід «працює, та й добре», при цьому не враховується, що може перестати працювати при найменшій зміні або ж породити важкі до відтворення дефекти після реального розгортання у клієнта). Це призводить до зниження якості продукту і накопиченню дефектів.

Методології. Існують методології, які дотримуються цінностей і принципів заявлених в Agile Manifesto, деякі з них:

1. Agile Modeling – набір понять, принципів і прийомів (практик), що дозволяють швидко і просто виконувати моделювання і документування в проектах розробки програмного забезпечення. Не включає в себе детальну інструкцію з проектування, не містить описів, як будувати діаграми на UML.

Основна мета – ефективне моделювання і документування; але не охоплює програмування та тестування, не включає питання управління проектом, розгортання і супроводу системи. Однак включає в себе перевірку моделі кодом.

2. Agile Unified Process (AUP) спрощена версія IBM Rational Unified Process (RUP), розроблена Скоттом Амблером, яка описує просте і зрозуміле наближення (модель) для створення програмного забезпечення для бізнес-додатків.

3 Agile Data Method – група ітеративних методів розробки програмного забезпечення, в яких вимоги та рішення досягаються в рамках співпраці різних крос-функціональних команд.

4. DSDM заснований на концепції швидкої розробки додатків (Rapid Application Development, RAD). Являє собою ітеративний і інкрементний підхід,

який надає особливого значення тривалій участі в процесі користувача/споживача.

5. Essential Unified Process (EssUP).

6. Екстремальне програмування (Extreme programming, XP).

7. Feature driven development (FDD) – функціонально-орієнтована розробка. Використовуване в FDD поняття функції або властивості (feature) Системи досить близько до поняття прецеденту використання, використовуваному в RUP, істотна відмінність – це додаткове обмеження: «кожна функція повинна допускати реалізацію не більше, ніж за два тижні». Тобто якщо сценарій використання досить малий, його можна вважати функцією. Якщо ж великий, то його треба розбити на декілька відносно незалежних функцій.

8. Getting Real – ітераційний підхід без функціональних специфікацій, що використовується для веб-додатків. У даному методі спершу розробляється інтерфейс програми, а потім її функціональна частина.

9. OpenUP – це ітераційно-інкрементний метод розробки програмного забезпечення. Позиціюється, як легкий і гнучкий варіант RUP. OpenUP ділить життєвий цикл проекту на чотири фази: початкова фаза, фази уточнення, конструювання та передачі. Життєвий цикл проекту забезпечує надання зацікавленим особам та членам колективу точок ознайомлення і прийняття рішень впродовж усього проекту. Це дозволяє ефективно контролювати ситуацію і вчасно приймати рішення про задовільність результатів. План проекту визначає життєвий цикл, а кінцевим результатом є остаточний додаток.

10. Scrum встановлює правила керування процесом розробки та дозволяє використовувати вже існуючі практики кодування, коректуючи вимоги або вносячи тактичні зміни. Використання цієї методології дає можливість виявляти і усувати відхилення від бажаного результату на більш ранніх етапах розробки програмного продукту.

11. Бережлива розробка програмного забезпечення (lean software development). Використовує підходи з концепції бережливого виробництва.

4.2 Захист розробленого програмного забезпечення

Для захисту розробленого програмного забезпечення запропоновано використовувати алгоритм Khufu. Khufu – це 64-бітовий блоковий шифр. 64-бітовий відкритий тест спочатку розщеплюється на дві 32-бітові половини, *L* і *R.* Над обома половинами й певними частинами ключа виконується операція XOR. Потім, аналогічно DES, результати проходять деяку послідовність раундів. У кожному раунді молодший значущий байт *L* використовується як вхід S-блоку. У кожного S-блоку 8 вхідних біт і 32 вихідних біта. Далі обраний в S-блоці 32-бітовий елемент піддається операції XOR з *R.* Потім *L* циклічно зрушується на число, кратним восьми биткам, *L* і *R* міняються місцями, і раунд завершується. Сам S-блок не статичний, він міняється кожні вісім раундів. Нарешті, по закінченні останнього раунду, над *L* і *R* виконується операція XOR з іншими частинами ключа, і половини поєднуються, утворюючи блок шифртексту.

Хоча частини ключа використовуються для операції XOR із блоком шифрування на початку й кінці виконання алгоритму, головне призначення ключа – генерація S-блоків. Ці S-блоки секретні, по суті, це частина ключа. Повний розмір ключа алгоритму Khufu дорівнює 512 біт (64 байт), алгоритм надає спосіб генерації S-блоків по ключу.

5 МЕТОДИКА ВПРОВАДЖЕННЯ СИСТЕМИ В ПРОМИСЛОВУ ЕКСПЛУАТАЦІЮ

Розглянемо розроблене ПЗ корпоративного центру управління інформаційною безпекою (SOC) яке зображено на рисунку 5.1.

З рисунку можна побачити що інтерфейс головного вікна розподілено на наступні функціональні розділи:

– Навігаційне меню: Система; Звіти; Налаштування; Довідка.

– Функції обрання дії.

– Розділу обрання групи.

– Розділу виведення результату роботи системи – журнал напрацювань.

– Навігаційного меню яке викликається натисканням правої клавіші маніпулятора миші.

– Функціональних кнопок ПЗ.

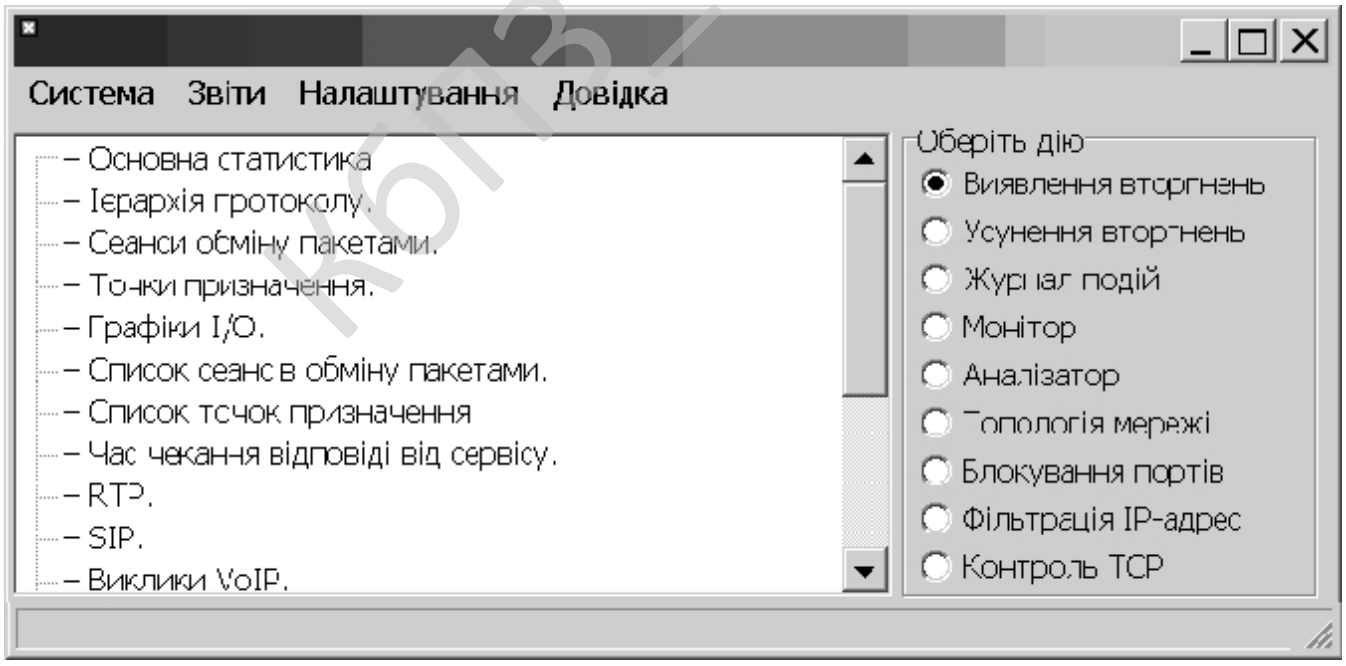

Рисунок 5.1 – Головне вікно ПЗ

Коли в організації немає єдиного центра моніторингу, інформація про інциденти розрізнена й не систематизована, ця обставина утрудняє, як оперативне реагування на інцидент, так і швидке і якісне його розслідування. Тому потрібна єдина база для збору інформації про всі інциденти ІБ, що відбулися в організації.

Розглянемо поетапно, як відбувається робота з обробки інцидентів:

– Виявлення інцидентів. На цьому етапі інформація про інциденти збирається централізовано з різних джерел, класифікується, аналізується.

– Реагування. Призначення відповідальних осіб і групи реагування, контроль строків і дій.

– Розслідування інциденту. На даному етапі збирається доказова база, свідчення, виявляються причини й обставини інциденту, що відбувся.

– Аналіз і статистика. Формуються статистичні дані по відділах, філіям, по типах. Виявляються основні зв'язки й залежності.

– Звітність. Формування й вивід різного виду звітів (для керівництва, для регуляторів і т.д.).

Інформація про інциденти при цьому може надходити в централізовану базу даних різними способами (по e-mail, через програмний інтерфейс API або вводиться вручну через веб-форму).

У рамках фіксації інцидентів, як правило, реєструються такі параметри інцидентів як: рівні критичності, рівні збитку, джерело інциденту, ступінь навмисності, статус реалізації, імовірність повторного виникнення, пріоритет і т.д.

Розроблена програма має дуже простий і інтуїтивно зрозумілий інтерфейс з користувачем.

Кожен, хто в достатньому обсязі володіє операційним середовищем Windows без особливих складностей освоїть і цю програму, оскільки її інтерфейс інтуїтивно зрозумілий.

Якщо програма не видала ніяких помилок, і працює, то можна використовувати, інакше слід слідувати інструкціям, які пропонує програма.

На рисунку 5.2 зображено авторські дані розробленого програмного забезпечення.

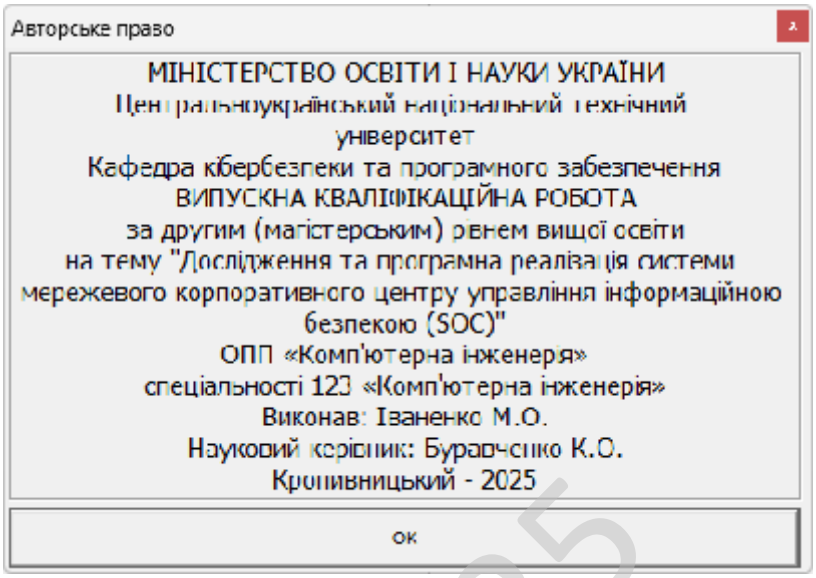

Рисунок 5.2 – Авторське право

Розглянемо процес впровадження програмного забезпечення, це процес налаштування програмного забезпечення під певні умови використання, а також навчання користувачів роботі з програмним продуктом. Впровадження програмного забезпечення це усі дії, що роблять розроблену програмну систему готовою до використання. Даний процес є частинною життєвого циклу програмного забезпечення.

Загалом процес розгортання складається з кількох взаємопов'язаних дій із можливими переходами між ними. Ця активність може відбуватися як з боку виробника так і з боку споживача. Оскільки кожна програмна система є унікальною, то усі процеси та процедури під час розгортання важко передбачити. Тому, "розгортання" можна трактувати як загальний процес відповідно до певних вимог та характеристик. Розгортання може здійснюватись програмістом і в процесі розробки програмного забезпечення.

До діяльностей пов'язаних із розгортанням програмного забезпечення відносять:

– Випуск.

– Встановлення та активація.

– Деактивація.

– Адаптація.

– Обновлення.

– Вмонтування.

– Відстежування версій.

– Видалення.

– Вилучення з обігу.

При впровадженні програмного забезпечення потрібно урахувати наступні дії:

– Виділення критичних, з точки зору загального результату, процедур в діяльності організації. Коли набір таких процедур визначений, необхідно в першу чергу використовувати ІТ рішення для автоматизації операцій усередині саме цих процедур. Таким чином, розроблене ІТ рішення автоматично стає життєво важливим і затребуваним для організації, а також буде забезпечена публічність процесу впровадження;

– Розширення нормативної бази організації шляхом включення до неї регламентів, що описують порядок виконання процедур автоматизованих процесів. В іншому випадку є небезпека виникнення неузгодженості між автоматизованими процедурами та іншими процесами організації.

– Виконання робіт з загальної стандартизації існуючої діяльності організації, коли виділяються кращі практики виконання процедур і включаються в ІТ рішення за принципом найбільшої корисності для більшості учасників. Відсоток таких процедур щодо загального обсягу автоматизації може бути невеликий, але це надає процесу побудови рішення вагу в організації за рахунок збільшення його необхідності.

Під час роботи над програмою було проведено тестування програмного забезпечення, тобто технічне дослідження, призначене для виявлення інформації про якість продукту відносно контексту, в якому воно має використовуватись.

Тестування включає як процес пошуку помилок або інших дефектів, так і випробування програмних складових з метою їх оцінки.

Проводилась оцінка:

– відповідності поставленим вимогам;

– правильна відповідь для усіх можливих вхідних даних;

– виконання функцій за прийнятний час;

– практичність;

– сумісність з ОС та стороннім ПЗ.

Оскільки число можливих тестів для програмних компонент практично нескінченне, тому стратегія тестування полягала в тому, щоб провести всі можливі тести з урахуванням наявного часу та ресурсів.

Як результат ПЗ тестувалось стандартним виконанням програми з метою виявлення помилок або інших дефектів.

Проводилось тестування форматом білої скриньки засноване на аналізі керуючої структури програми. Програма вважається повністю перевіреною, якщо проведено вичерпне тестування маршрутів (шляхів) її графа управління.

У цьому випадку формуються тестові варіанти, в яких:

– Гарантується перевірка всіх незалежних маршрутів програми.

– Знаходяться гілки True, False для всіх логічних рішень.

– Виконуються всі цикли (у межах їхніх кордонів та діапазонів).

– Аналізується правильність внутрішніх структур даних.

Недоліки тестування "білої скриньки":

– Кількість незалежних маршрутів може бути дуже велика.

– Повне тестування маршрутів не гарантує відповідності програми вихідним вимогам до неї.

– У програмі можуть бути пропущені деякі маршрути.

– Не можна виявити помилки, поява яких залежить від даних.

Переваги тестування "білої скриньки" пов'язані з тим, що принцип «білої скриньки» дозволяє врахувати особливості програмних помилок:

– Кількість помилок мінімально в «центрі» і максимально на «периферії» програми.

– Попередні припущення про ймовірність потоку керування або даних у програмі часто бувають некоректними. У результаті типовим може стати маршрут, модель обчислень за яким опрацьована слабо.

– При записі алгоритму програмного забезпечення у вигляді тексту на мові програмування можливе внесення типових помилок трансляції (синтаксичних та семантичних).

– Деякі результати в програмі залежать не від вихідних даних, а від внутрішніх станів програми.

Проводилось тестування чорної скриньки.

Основне місце програми тестів «чорної скриньки» – інтерфейс ПЗ. Відомі: функції програми. Досліджується: робота кожної функції на всій області визначення.

Ці тести демонструють:

– Як виконуються функції програми.

– Як приймаються вихідні дані.

– Як виробляються результати.

– Як зберігається цілісність зовнішньої інформації.

При тестуванні «чорної скриньки» розглядаються системні характеристики програм, ігнорується їхня внутрішня логічна структура. Вичерпне тестування, як правило, неможливе.

Наприклад, якщо в програмі 10 вхідних величин і кожна приймає по 10 значень, то кількість тестових варіантів становитиме 10^10. Тестування «чорної скриньки» не реагує на багато особливостей програмних помилок.

Тестування «чорної скриньки» (функціональне тестування) дозволяє отримати комбінації вхідних даних, які забезпечують повну перевірку всіх функціональних вимог до програми.

Програмний виріб тут розглядається як «чорна скринька», чию поведінку можна визначити тільки дослідженням його входів та відповідних виходів. При такому підході бажано мати:

– Набір, утворений такими вхідними даними, які призводять до аномалій у поведінці програми (назвемо його ITc).

– Набір, утворений такими вхідними даними, які демонструють дефекти програми (назвемо його OT).

Будь-який спосіб тестування «чорної скриньки» повинен:

– Виявити такі вхідні дані, які з високою ймовірністю належать набору ITc;

– Сформулювати такі очікувані результати, які з високою імовірністю є елементами набору OT.

Принцип «чорної скриньки» не альтернативний принципу «білої скриньки». Скоріше це доповнює підхід, який виявляє інший клас помилок.

Тестування «чорної скриньки» забезпечує пошук наступних категорій помилок:

– Некоректних чи відсутніх функцій;

– Помилок інтерфейсу;

– Помилок у зовнішніх структурах даних або в доступі до зовнішньої бази даних;

– Помилок характеристик (необхідна ємність пам'яті і т.д.);

– Помилок ініціалізації та завершення.

Обрано умови розповсюдження – Shareware. Під умовно-безплатним програмним забезпеченням можна розуміти спосіб або метод розповсюдження комерційного ПЗ на ринку (тобто на шляху до кінцевого користувача), при якому випробувачеві пропонується обмежена за можливостями (не повнофункціональна

або демонстраційна версія), терміном дії (тріал версія) або версія з вбудованим набридливим нагадуванням про необхідність оплати використання програми.

В угоді про використання (ліцензії для кінцевого користувача, EULA) також може бути обумовлена заборона на комерційне або професійне (не тестове) її використання.

Основний принцип умовно-безплатного ПЗ – «спробуй, перш ніж купити» (try before you buy). ПЗ що поширюється як умовно-безплатний, надається користувачам безоплатно. Звичайно користувач платить тільки за час завантаження файлів через Інтернет або за носій (CD диск, флешку, ключ). Протягом певного терміну, що становить зазвичай тридцять днів, він може користуватися програмою, тестувати її, освоювати її можливості.

Якщо після закінчення цього терміну користувач вирішить продовжити використання ПЗ, він зобов'язаний купити його (зареєструватися), заплативши авторові певну суму.

В іншому випадку користувач повинен припинити використання ПЗ та видалити його зі свого комп'ютера.

6 НАУКОВА НОВИЗНА

У випускній кваліфікаційній роботі за другим (магістерським) рівнем вищої освіти розроблено програмне забезпечення, яке призначено для системи мережевого корпоративного центру управління інформаційною безпекою (SOC).

Метою розробки є дослідження та програмна реалізація системи мережевого корпоративного центру управління інформаційною безпекою (SOC).

Об'єктом дослідження є процес мережевого корпоративного центру управління інформаційною безпекою (SOC).

Предметом дослідження є методи мережевого корпоративного центру управління інформаційною безпекою (SOC).

Методи дослідження базуються на методах захисту інформації у мережі, методах математичної статистики, методах розробки програмного забезпечення.

Наукова новизна отриманих результатів. У процесі рішення завдань, обумовлених цілями дослідження, отримані наступні результати:

– Удосконалено метод мережевого корпоративного центру управління інформаційною безпекою (SOC).

– Розроблено вітчизняний продукт мережевого корпоративного центру управління інформаційною безпекою (SOC), який має більш широкі можливості, на відміну від існуючих аналогів.

7 МАРКЕТИНГОВЕ ТА ЕКОНОМІЧНЕ ОБГРУНТУВАННЯ ІТ-ПРОЄКТУ

7.1 Визначення цільової аудиторії кінцевого готового продукту

Результати розробки та дослідження системи мережевого корпоративного SOC (Security Operations Center) можуть бути цікавими насамперед великим корпораціям, які щоденно працюють із великими обсягами даних і мають розгалужену ІТ-інфраструктуру. Для таких компаній важливо мати можливість централізовано контролювати інформаційні потоки, виявляти підозрілу активність і своєчасно реагувати на потенційні загрози. SOC допомагає скоротити час реагування на інциденти, підвищити ефективність внутрішнього моніторингу й мінімізувати ризики витоку даних або зупинки бізнес-процесів.

Не менш значущою така система буде для державних установ, що зберігають критично важливі дані – наприклад, інформацію про громадян, фінансові операції або національні проєкти. SOC у державному секторі може стати елементом національної системи кіберзахисту, що об'єднує різні відомства в єдину інформаційну мережу безпеки. Це дає змогу оперативно обмінюватися даними про загрози та підвищує стійкість держави до кібератак.

Крім цього, система буде корисною компаніям, які займаються фінансовими технологіями, банківською справою, телекомунікаціями та логістикою. У цих сферах інформаційна безпека є не лише технічною потребою, а частиною репутації бізнесу. Потенційними користувачами SOC також можуть бути компанії середнього рівня, які прагнуть створити власну службу безпеки, але не мають ресурсів для повномасштабного впровадження – у такому випадку вони можуть використовувати SOC як сервіс (SOC-as-a-Service).

Водночас наукові установи, які досліджують кібербезпеку, також можуть бути зацікавлені у використанні такої системи як експериментального

середовища для аналізу кіберзагроз, тестування нових методів виявлення атак і відпрацювання сценаріїв реагування. Таким чином, розробка SOC має міжгалузевий характер і може бути затребуваною як у державному, так і у приватному секторі, сприяючи підвищенню рівня цифрової безпеки на всіх рівнях.

7.2 Оцінка привабливості шляхом застосування методів експертних оцінок

Для оцінки привабливості системи SOC було проведено опитування серед експертів у галузі інформаційної безпеки, адміністраторів корпоративних мереж, аналітиків з кіберзахисту та представників компаній, які вже впровадили аналогічні рішення. Кожен експерт оцінював систему за кількома критеріями – рівень автоматизації процесів моніторингу, ефективність виявлення загроз, масштабованість, простоту інтеграції у корпоративну інфраструктуру та економічну доцільність впровадження.

У середньому система отримала оцінку 9,2 бала з 10. Найвищі бали були надані за швидкість обробки подій і ефективність реагування – експерти відзначили, що завдяки автоматизованим сценаріям реагування SOC здатен виявляти інциденти за хвилини, тоді як раніше цей процес міг тривати годинами. Високо оцінено також рівень візуалізації даних – інтерфейс системи дозволяє в реальному часі відстежувати статус загроз і отримувати аналітику з різних сегментів мережі.

Дещо нижчі оцінки отримав критерій вартості, оскільки впровадження SOC потребує початкових інвестицій у апаратну та програмну інфраструктуру. Проте більшість експертів підкреслили, що окупність таких проєктів становить менше року завдяки суттєвому скороченню фінансових втрат від інцидентів. Також було зазначено, що система має високий потенціал масштабування і може бути використана як для одного підприємства, так і для групи компаній.

Результати експертного оцінювання показали, що SOC є конкурентоспроможним і сучасним рішенням, яке відповідає вимогам ринку, має високу практичну цінність і перспективу широкого впровадження. Це підтверджує не лише технічну, а й економічну доцільність його реалізації.

7.3 Вибір методу оцінки вартості ПЗ

Для оцінки вартості впровадження системи SOC найкраще використовувати комбінований підхід, який поєднує витратний і дохідний методи. Витратний метод дозволяє врахувати всі прямі й непрямі витрати – починаючи від розробки програмного забезпечення, закупівлі серверного обладнання, налаштування мережевої інфраструктури, і закінчуючи витратами на навчання персоналу. Такий підхід забезпечує реалістичне розуміння того, у скільки фактично обходиться створення SOC для компанії.

Дохідний метод, у свою чергу, дозволяє визначити потенційну вигоду від функціонування SOC, виражену у зменшенні фінансових втрат від інцидентів безпеки, економії часу працівників IT-відділу та зниженні вартості зовнішніх аудитів. Це допомагає оцінити не лише поточну, а й довгострокову ефективність інвестицій у систему.

Застосування обох методів у комплексі дозволяє розробити обґрунтовану фінансову модель, що відображає реальні умови роботи підприємства. Якщо, наприклад, компанія в середньому зазнає 1 млн грн збитків від кіберінцидентів на рік, а SOC дозволяє скоротити їх на 80%, це означає, що економічна віддача перевищить інвестиції вже протягом першого року експлуатації.

Таким чином, комбінований метод оцінки є найбільш доцільним, оскільки він не лише фіксує витрати, а й відображає стратегічну вигоду від посилення інформаційної безпеки. У випадку SOC економічна цінність полягає не лише в заощаджених коштах, а й у запобіганні катастрофічним наслідкам, які можуть виникнути через компрометацію даних.

7.4 Розрахунок економічної ефективності від впровадження реалізованого ПЗ як фактору його привабливості

Корпоративна мережа великого підприємства включає понад 1 000 користувачів, 50 серверів, десятки віддалених підрозділів і постійно підключених партнерських сервісів. До впровадження SOC фіксувалися часті інциденти інформаційної безпеки – спроби фішингових атак, зараження шкідливим ПЗ, витоки внутрішніх даних. Відсутність централізованого моніторингу призводила до того, що на виявлення загроз витрачалося в середньому 48 годин, а реагування – до 72 годин. Через це компанія щороку зазнавала збитків через простої, відновлення систем і втрату репутації. Мета створення SOC – централізувати моніторинг, скоротити час виявлення та реагування на інциденти, автоматизувати аудит безпеки та підвищити рівень інформаційної стійкості всієї корпоративної інфраструктури. Вхідні дані зафіксовано в таблиці 7.1.

Розрахунок економічного ефекту демонструє наступне: зменшення збитків від інцидентів – 1 350 000 грн/рік, економія на аудиті та перевірках – 300 000 грн/рік, сукупний річний економічний ефект – 1 650 000 грн/рік, чистий ефект – 1 400 000 грн, термін окупності (Payback Period) – 0,86 року (~10 місяців), коефіцієнт економічної ефективності (ROI) – 117%.

Додаткові вигоди (немонетарні показники): зниження ризику витоку конфіденційних даних на понад 70%, скорочення часу реагування на загрози із трьох діб до кількох годин, автоматизація звітності для керівництва та зовнішніх перевірок, підвищення довіри клієнтів і партнерів завдяки сертифікації ISO/IEC 27001, зниження навантаження на IT-відділ за рахунок автоматизованих сценаріїв реагування (SOAR).

Таким чином, створення SOC не лише підвищує економічну ефективність компанії, а й формує цифрову культуру безпеки, що стає важливою складовою конкурентоспроможності сучасного бізнесу.

Таблиця 7.1 – Вихідні дані для розрахунку

Показник	До впровадження	Після впровадження	Економічний ефект
Кількість серйозних інцидентів на рік	12	3	–9
Середні втрати від одного інциденту	150 000 грн	40 000 грн	–110 000 грн
Середній час виявлення інциденту	48 годин	4 години	–44 години
Середній час реагування	72 години	8 годин	–64 години
Річні витрати на аудит безпеки та зовнішні перевірки	600 000 грн	300 000 грн	–300 000 грн
Вартість впровадження SOC (одноразово)	—	—	1 200 000 грн
Річні витрати на обслуговування SOC	—	—	250 000 грн

7.5 Пропозиція алгоритму просування проєкту розробки ПЗ

Просування проєкту системи SOC має починатися з формування довіри до продукту. Найефективнішим кроком є створення демонстраційної лабораторії, де потенційні клієнти можуть побачити роботу системи в реальному часі – від збору даних до реагування на загрози. Така демонстрація створює відчуття надійності та показує практичну користь продукту.

Наступним кроком є активна участь у професійних конференціях, форумах і виставках з кібербезпеки. Це не лише сприяє підвищенню впізнаваності продукту, а й дає змогу отримати перші відгуки від фахівців, які можуть допомогти вдосконалити систему. Варто також співпрацювати із ЗМІ, що

висвітлюють IT-тематику, та створювати аналітичні публікації, у яких показано переваги SOC перед іншими рішеннями.

Після цього доцільно запустити програму партнерських інтеграцій з IT-компаніями, які займаються інфраструктурними рішеннями для підприємств. Завдяки таким партнерствам SOC може бути запропонований як частина комплексного пакету послуг, що значно розширить охоплення аудиторії.

Важливу роль відіграє також робота з клієнтами після продажу. Надання якісної технічної підтримки, оновлень і консультацій створює довіру до розробників і формує довгострокові відносини з користувачами. Просування SOC має бути не лише маркетинговим процесом, а постійним діалогом між командою розробників і спільнотою фахівців з інформаційної безпеки.

7.6 Оптимізація каналів збуту та шляхів реалізації ПЗ

Оптимізація каналів збуту системи SOC повинна орієнтуватися на поєднання прямих продажів і партнерських програм. Найефективніше продавати систему через інтеграторів IT-рішень, які вже мають клієнтів у корпоративному секторі. Це дозволить скоротити час виходу на ринок і знизити маркетингові витрати.

Водночас доцільно впровадити модель "SOC-as-a-Service", яка дозволяє клієнтам користуватися послугами центру без необхідності створювати власну інфраструктуру. Це особливо привабливо для середніх і малих підприємств, які не мають великого бюджету, але потребують високого рівня кіберзахисту.

Необхідно також посилити цифрову присутність – створити офіційний сайт, сторінку у професійних мережах, публікувати кейси успішного впровадження. Розміщення інформації на тематичних порталах із кібербезпеки допоможе досягти більшої аудиторії. Крім того, можна використовувати освітні заходи – вебінари, тренінги й сертифікаційні програми, щоб залучати фахівців до співпраці та поширювати знання про SOC.

Зрештою, ключем до ефективної реалізації SOC є створення гнучкої моделі обслуговування клієнтів. Вона має включати швидке реагування, адаптацію під потреби кожного підприємства та регулярні звіти про стан безпеки. Такий підхід забезпечить не лише продаж, а й довгострокову лояльність клієнтів.

7.7 Визначення ключових факторів успіху конкретного проєкту

Успіх проєкту SOC базується на трьох основних складових: технологічній надійності, людському факторі та довірі користувачів. У технічному аспекті система повинна забезпечувати цілодобовий моніторинг, швидке виявлення інцидентів і безперебійну роботу навіть під великим навантаженням. Надійність і точність виявлення загроз є критично важливими – саме вони формують перше враження про якість продукту.

Другим ключовим фактором є професіоналізм команди, що обслуговує SOC. Навіть найкраща технологія не буде ефективною без кваліфікованих аналітиків, які можуть правильно інтерпретувати події, оцінити рівень загрози та прийняти рішення. Саме людський досвід є тим елементом, що перетворює систему з технічного рішення на ефективний інструмент управління безпекою.

Не менш важливим чинником є гнучкість системи – здатність адаптуватися до змін ринку, нових видів атак і специфіки кожного клієнта. SOC, який легко інтегрується з іншими рішеннями (SIEM, EDR, DLP тощо), має набагато більші шанси на успіх.

І, нарешті, успіх будь-якого SOC вимірюється рівнем довіри, який він здобуває. Якщо користувачі відчувають стабільність, швидкість реагування й відкритість розробників до вдосконалення, це формує довгострокові відносини. Саме поєднання технологічної досконалості, професіоналізму та клієнтської довіри є запорукою того, що SOC стане не просто проєктом, а невід'ємною частиною культури безпеки сучасної організації.

8 ЗАХОДИ З ОХОРОНИ ПРАЦІ ТА ТЕХНІКИ БЕЗПЕКИ

8.1 Вступ

Наявний в даний час в нашій країні комплекс розроблених організаційних заходів та технічних засобів захисту, накопичений передовий досвід роботи ряду обчислювальних центрів показує, що є можливість домогтися значно більших успіхів у справі усунення впливу на працюючих небезпечних і шкідливих виробничих факторів. Проте стан умов праці та його безпеки в ряді обчислювальних центрів (ОЦ) та підприємств ще не задовольняють сучасним вимогам. Оператори ЕОМ, оператори підготовці даних, програмісти та інші працівники ОЦ та підприємств ще стикаються з впливом таких фізично небезпечних і шкідливих виробничих факторів, як підвищений рівень шуму, підвищена температура зовнішнього середовища, відсутність або недостатня освітленість робочої зони, електричний струм, статична електрика і інші.

Багато працівників ОЦ та підприємств пов'язані з впливом таких психофізичних факторів, як розумова перенапруга, перенапруження зорових і слухових аналізаторів, монотонність праці, емоційні перевантаження. Вплив зазначених несприятливих факторів призводить до зниження працездатності, викликане розвиваються втомою. Поява і розвиток втоми пов'язане зі змінами, які виникають під час роботи в центральній нервовій системі, з гальмівними процесами в корі головного мозку. Наприклад сильний шум викликає труднощі з розпізнаванням колірних сигналів, знижує швидкість сприйняття кольору, гостроту зору, зорову адаптацію, порушує сприйняття візуальної інформації, зменшує на 5 – 12% продуктивність праці. Тривала дія шуму з рівнем звукового тиску 90 дБ знижує продуктивність праці на 30 – 60%.

Медичні обстеження працівників ОЦ та підприємств показали, що крім зниження продуктивності праці високі рівні шуму призводять до погіршення

слуху. Тривале перебування людини в зоні комбінованого впливу різних несприятливих факторів може призвести до професійного захворювання. Аналіз травматизму серед працівників ВЦ показує, що в основному нещасні випадки відбуваються від впливу фізично небезпечних виробничих факторів при заправці носія інформації на обертовий барабан при знятому кожусі, під час співробітниками невластивих їм робіт. На другому місці випадки, пов'язані з дією електричного струму.

8.2 Шкідливі і небезпечні фактори при роботі з комп'ютером

Електронно-обчислювальна машина (ЕОМ) та інше обладнання є джерелами небезпеки ураження електричним струмом. Так як робота програміста характеризується істотним зоровим навантаженням, то вимагає належного освітлення. У приміщенні, в якому працюють програмісти необхідно створити належний мікроклімат, параметри якого регламентуються, Державними санітарними правилами і нормами, зокрема ДСанПіН 3.3.2.007-98.

При роботі з використанням ЕОМ відзначають наступні небезпечні та шкідливі фактори:

– ризик виникнення надзвичайних ситуацій природного або штучного характеру на об'єкті або території.

– ризик виникнення пожежі;

– негативний вплив на органи зору людини;

– ризики ураження електричним струмом;

– недостатня, або надмірна освітленість робочого місця;

– електромагнітні (у тому числі високочастотні) випромінювання (коливання);

– несприятливі мікрокліматичні умови;

– нервово-емоційна напруженість праці;

– інтелектуальні навантаження;

– монотонність праці;

– невідповідність ергономічних показників робочого місця діючим вимогам;

– шум;

– статичні навантаження на кістково-м’язовий апарат.

8.3 Аналіз санітарно-гігієнічних умов праці на робочому місці програміста

Розглянемо умови праці у приміщенні, в якому працюють програмісти. Геометричні розміри приміщення наведено у таблиці 8.1.

Таблиця 8.1 – Розміри приміщення

Найменування	Значення, м
Ширина	5
Довжина	5,95
Висота	2,8

Таблиця 8.2 – Площа та обсяг приміщення, на одного працюючого*

Геометрична характеристика	Одиниця виміру	Нормативне значення*	Фактичне значення
Площа, S	$м^2$	не менше 6.0	7,4
Об'єм, V	$м^3$	не менше 20.0	20,8

* Згідно ДСанПіН 3.3.2.007-98 (Державні санітарні правила і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин).

У зазначеному приміщенні працюють четверо людей. За даними, які наведено у табл. 8.1, та табл. 8.2, можна зробити висновок, що площа та об’єм приміщення у розрахунку на одне робоче місце програміста не відповідають

нормативним вимогам ДСанПіН 3.3.2-007-98 «Державні санітарні правила і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин» [5], але відповідають нормативним вимогам Наказу Міністерства соціальної політики України № 207, від 14.02.2018 «Про затвердження Вимог щодо безпеки та захисту здоров'я працівників під час роботи з екранними пристроями» [5] та НПАОП 0.00-1.28-10 «Правила охорони праці під час експлуатації електронно-обчислювальних машин»). Таним чином можна зробити висновок, що санітарно-гігієнічні умови праці на робочому місці програміста відповідають вимогам.

Температура повітря в приміщенні визначається впливом температури зовнішнього повітря і тепловою енергією, яка виділяється всередині приміщення. Джерелами виділення теплоти в даному приміщенні є електроустаткування, освітлювальні прилади, а також люди. У світлий час доби джерелом надлишкового тепла є сонячна радіація. Згідно Постанови № 42 від 01.12.1999 Головного державного санітарного лікаря України, робота, виконувана в даному приміщенні, відноситься до категорії Іа. В цьому випадку людина витрачає енергії до 120 ккал у годину. Вологість повітря в приміщенні визначається впливом багатьох факторів, серед яких: вологість атмосферного повітря, виділення вологи людьми (при диханні та випарами з поверхні шкіри).

Мікроклімат повітряного середовища в приміщенні характеризується запиленістю та загазованістю повітря. Мікроклімат приміщення визначається діючим на організм людини поєднанням, вологості, температури, швидкості руху повітря та інтенсивності теплового випромінювання. Аналіз мікроклімату складається з визначення зазначених вище факторів і порівняння результатів із встановленими нормами.

У таблиці 8.3 наведено оптимальні та фактичні значення параметрів мікроклімату як для категорії ваги робіт Іа, так і розглянутого приміщення. У приміщеннях, де встановлено ЕОМ, рекомендується застосування тільки оптимальних значень показників мікроклімату.

Проведений аналіз показує, що показники мікроклімату в приміщенні відповідають установленим нормам. Штучне опалення застосовується у холодний період року.

Таблиця 8.3 – Оптимальні і фактичні значення параметрів мікроклімату

Пора року	Оптимальні для Іа			Фактичні		
	Температура, °С	Воло-гість,%	Швидкість повітря, м/с	Температура °С	Воло-гість,%	Швидкість повітря, м/с
Холодна	22-24	40-60	0,1	22-23	40-55	0,1
Тепла	23-25	50-70	0,1	24-25	50-65	0,11

В літню пору застосовується кондиціонер.

Для боротьби з пилом робляться регулярні провітрювання та вологі прибирання приміщенні.

У приміщенні знаходяться наступні джерела шуму: принтер HP 1100, електродвигуни вентиляторів ЕОМ.

Одним з найважливіших факторів, які впливають на ефективність трудової діяльності людини, та попереджають травматизм і професійні захворювання програмістів є освітлення на робочому місці.

З 2019 року діють Державні будівельні норми України “Природне і штучне освітлення” – ДБН В.2.5-28:2018 [1], у яких прописані вимоги до використання всіх освітлювальних приладів, у тому числі світлодіодних.

Працю працівника, який постійно працює за комп’ютером, згідно ДБН В.2.5-28:2018 [1], можна віднести до роботи з малою точністю (найменший розмір об'єкта розрізнення від 1 до 5 мм) V-го розряду зорової роботи, з великою контрастністю об'єкта розрізнення (символів на екрані дисплея), з темним тлом (під розряд зорової роботи В). Приміщення можна віднести до 1-ої групи приміщень, у яких проводиться розрізнення об'єктів зорової роботи при фіксованому напрямку лінії зору того, що працює на робочу поверхню. Для

такого типу приміщень і розряду зорової роботи нормоване значення коефіцієнту природної освітленості (КПО) робочої поверхні (при поєднаному, спільному освітленні), повинен становити не більше 1,5%, освітленість при штучному висвітленні повинна становити 300 Лк. [1], Крім того все поле зору повинне бути освітлено достатньо рівномірно – це основна гігієнічна вимога. Оскільки яскраве світло на ділянці периферійного зору значно збільшує напруженість очей і, як наслідок, призводить до їх швидкої стомлюваності, ступінь освітлення приміщення і яскравість екрану комп'ютера повинні бути приблизно однаковими.

8.4 Розробка заходів з умов поліпшення охорони праці

Згідно аналізу умов праці в розглянутому приміщенні, ми одержали наступні результати:

– розмірі приміщення, у розрахунку на одному працюючого, відповідають нормативам;

– мікроклімат відповідає нормативному значенню;

– акустичні умови роботи не перевищують нормативних значень;

Таким чином, можна припустити, що основною причиною можливого зниження працездатності програміста є психофізіологічний фактор, тому основна пропозиція буде така: дотримання позитивної психологічної атмосфери в колективі та регламентованого режиму праці та відпочинку, організація робочого місця з урахуванням ергономічних вимог.

Рекомендовані заходи: регулярні періодичні наочні огляди персоналом шляхів для евакуації людей із приміщення, відповідно до плану евакуації (який повинен розташовуватись на видному місці у приміщенні), включення до колективного договору мінімально можливого вмісту аптечок з обов’язковою наявністю масок-клапанів, або іншого спорядження для штучного дихання. Регулярна періодична перевірка параметрів заземлення та занулення (вимірювання опору ланцюга).

Регулярна наочне знайомство персоналу із шляхами для евакуації людей із приміщення відповідно до плану евакуації, забезпечення розподільних щитів спеціальними розетками з заземлюючими контактами; організація заземлення всіх приладів і пристроїв, які працюють при напрузі вище 36 В.

Так як при ураженні електричним струмом у людини може статися фібриляція шлуночків серця, в організації бажано мати дефібрилятор і підготовлений персонал для роботи з ним.

8.5 Розрахункова частина

Проведемо розрахунок штучного освітлення за методом коефіцієнту використання світлового потоку для приміщення ширина якого складає 5 м, довжина – 5,95 м, висота – 2,8 м.

У зазначеному приміщенні працює 4 людей.

Для того, щоб визначити потрібну кількість світильників, які повинні забезпечити нормований рівень освітленості, визначимо світловий потік, що падає на робочу поверхню за формулою [1]:

$$F = E \cdot S \cdot K \cdot Z / n,$$

де:

F – світловий потік, що розраховується, Лм;

E – нормована мінімальна освітленість, Лк; E = 300 Лк;

S – площа освітлюваного приміщення (у нашому випадку S=5 х 5,95 = 29,7 м^2);

K – коефіцієнт запасу, що враховує зменшення світлового потоку лампи в результаті забруднення світильників у процесі експлуатації (його значення залежить від типу приміщення і характеру робіт, що проводяться в ньому, в нашому випадку K = 1,5);

Z – відношення середньої освітленості до мінімальної (зазвичай приймається рівним 1.1... 1.2, в нашому випадку Z = 1,1);

n – коефіцієнт використання світлового потоку, (відношення світлового потоку, що падає на розрахункову поверхню, до сумарного потоку всіх ламп, обчислюється в долях одиниці; залежить від характеристик світильника, розмірів приміщення, забарвлення стін і стелі, що характеризуються коефіцієнтами відбиття від стін ($\rho_{стін.}$) і стелі ($\rho_{стелі}$), значення коефіцієнтів дорівнюють $\rho_{стін}$ = 50% і $\rho_{стелі}$ = 50%.

Обчислимо індекс приміщення за формулою:

$$i = S/(h\cdot (A+B)),$$

де:

S – площа приміщення, *S* = 29,7 м2;

h – розрахункова висота підвісу, *h* = 3 м (співпадає з висотою стелі, оскільки лампи освітлення закріплюються на стелі);

A – ширина приміщення, *A* = 5 м;

B – довжина приміщення, *B* = 5,95 м.

Підставимо всі значення у формулу та визначимо індекс приміщення: i=1,4.

Знаючи індекс приміщення, за знаходимо *n* = 0,29 (з табличних даних коефіцієнтів використання світлового потоку (*n*) світильників з відповідним типом лампам) [8]. Підставимо всі значення у формулу, визначимо світловий потік: F=64027 Лм.

Для розрахунку будемо використовувати світлодіодні стельові панелі Призма-72 6400К, світловий потік яких $F_л$ = 7200 *Лм.*

Число ламп визначається за формулою:

$$N=F/F_л$$

де:

F – світловий потік,

$F_л$ – світловий потік однієї лампи.

Підставимо всі значення у формулу та визначимо індекс приміщення:

N= 64027 / 7200=8,8 шт.

Приймаємо необхідну кількість світлодіодних світильників 9 шт.

Висновки до розділу

Дотримання всіх необхідних умов праці не лише сприяє збереженню здоров’я працівників, а також підвищує ефективність виробництва в цілому.

З цих міркувань було здійснено аналіз умов праці, призначеного для праці програмістів, проведено розгляд небезпечних та шкідливих факторів, що негативно впливають на програмістів під час роботи. Виконано розрахунок штучного освітлення, як одного з ключових факторів впливу на працездатність та здоров’я програміста. Розроблено заходи з умов поліпшення охорони праці.

9 ОСНОВНІ ВИСНОВКИ

Програмне забезпечення, створене в результаті виконання випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти, призначено для системи мережевого корпоративного центру управління інформаційною безпекою (SOC).

В межах України в недостатній мірі представлені вітчизняні розробки в цій області.

У випускній кваліфікаційній роботі за другим (магістерським) рівнем вищої освіти наведені теоретичне узагальнення й рішення наукового завдання дослідження методів мережевого корпоративного центру управління інформаційною безпекою (SOC).

Рішення даного завдання полягало у вирішенні наступних задач:

– Був проведений огляд існуючих систем мережевого корпоративного центру управління інформаційною безпекою (SOC).

– Досліджена система мережевого корпоративного центру управління інформаційною безпекою (SOC).

– На основі отриманих результатів досліджень створена програмна реалізація системи мережевого корпоративного центру управління інформаційною безпекою (SOC).

Розроблені під час виконання випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти алгоритми дозволяють успішно вирішувати завдання мережевого корпоративного центру управління інформаційною безпекою (SOC).

Проведено аналіз предметної галузі в ході якого були виявлені об'єкти, взаємодія яких носить істотний характер для функціональної діяльності предметної галузі, і їхні основні характеристики; побудована алгоритм і вибраний середовище розробки.

Розроблене програмне забезпечення має простий, дружній та зручний інтерфейс користувача, що забезпечує легкість у освоєнні роботи програмного продукту, зручність у використанні, і не потребує особливих спеціальних знань.

При створені програмного забезпечення було використано об'єктно-орієнтований підхід, що відповідає сучасним тенденціям у галузі розробки комерційних програмних систем.

Програма реалізована на мові високого рівня Python. Дана мова програмування дозволяє найбільш ефективно обробляти дані. Це дозволило мінімізувати строк розробки програмного забезпечення, і, як слід, зменшити витрати на його розробку. Запропоноване програмне забезпечення ділиться на загальне програмне забезпечення, що поставляється із засобами обчислювальної техніки й спеціальне програмне забезпечення, що спеціально розроблене для даної конкретної системи й включає програми, що реалізують її функції.

Програма призначена для виконання під управлінням багатозадачної операційної системи Windows 10/11.

Даються необхідні рекомендації з установки розробленого програмного забезпечення.

Для підвищення рівня безпеки запропоновано застосовувати алгоритм Khufu.

В цілому створене програмне забезпечення підтверджує правильність використаних проектних рішень та повністю відповідає вимогам технічного завдання. Створене програмне забезпечення має потенційну можливість для подальшого вдосконалення і застосування у різних галузях.

Проведено маркетингове та економічне обґрунтування ІТ-проєкту, що дозволило визначити ключові фактори успіху даного проекту.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Іваненко М.О. Дослідження та програмна реалізація системи мережевого корпоративного центру управління інформаційною безпекою (SOC) // Збірник праць молодих науковців ЦНТУ. – Вип. 15. – Кропивницький: ЦНТУ, 2025.

2. Kuznetsov, O., Smirnov, O., Kuznetsova, T., Shaikhanova, A., Svatowsky, I. «Privacy-utility trade-offs in IoT networks: A comparative analysis of differential privacy mechanisms for sensor data aggregation». *Security and Privacy of Cyber Physical Systems Emerging Trends Technologies and Applications*, 2025, pp. 589–622.

3. Lakhno, V., Malyukov, V., Smirnov, O., Bebeshko, B., Chubaievskiy, V., Zhumadilova, M., Malyukova, I., Smirnov, S. «Multifactorial Model for Targeted Attacks Counteracting Within the Framework of a Multi-Step Quality Game with Fuzzy Information». *8th International Symposium on Intelligent Informatics, ISI 2023*, 2025. vol 389. pp 377-389. Springer, Singapore.

4. Kuznetsov O., Frontoni E., Kuznetsova Y., Smirnov O., Moskovchenko I. «Trust-Based Security Architecture for Edge Computing: A Simulation Study of Dynamic Trust Evolution and Attack Detection». *CEUR Workshop Proceedings,* 2024, 3909, pp. 227–241.

5. Kuznetsov, O., Frontoni, E., Kryvinska, N., Chevardin, V., Smirnov, O. «Wireless Network Encryption Stream Ciphers, Computational Modeling, and Security Analysis». *Computational Modeling and Simulation of Advanced Wireless Communication Systems*, 2024, pp. 379–402.

6. Kuznetsov, O., Frontoni, E., Kryvinska, N., Smirnov, O., Imoize, G.L. «Computational Modeling of Enhanced Spread Spectrum Codes for Asynchronous Wireless Communication». *Computational Modeling and Simulation of Advanced Wireless Communication Systems*, 2024, pp. 403–447.

7. Ткаченко, О., Ільєнко, А., Улічев, О., Мелешко, Є., Смірнов, О.

«Правові засади поширення інформаційних впливів в соціальних мережах». *Електронне фахове наукове видання «Кібербезпека: освіта, наука, техніка»,* 2024. № 2(26), С. 170–188.

8. Смірнова Т.В., Конопліцька-Слободенюк О.К., Буравченко К.О., Смірнов С.А., Кравчук О.В., Козірова Н.Л., Смірнов О.А. «Дослідження технологій забезпечення кібербезпеки хмарних сервісів IaaS, PaaS та SaaS». *Кібербезпека: освіта, наука, техніка.* 2024. №4(24), С. 6-27.

9. Вінтенко, Б., Миронець, І., Смірнов, О., Кравчук, О., Козірова, Н., Савеленко, Г., Коваленко, А. «Дослідження вимог та аналіз кібербезпеки програмного забезпечення інформаційно-керуючих систем АЕС, важливих для безпеки». *Кібербезпека: освіта, наука, техніка.* 2024. №3(23), С. 111-131.

10. Батрак О., Смірнова Т., Гнатюк В., Одарченко Р., Смірнов О. «Дослідження показників ефективності функціонування та перспектив розвитку систем IP-телефонії». *Підводні технології,* 2024, № 13, с. 28-35.

11. Kuznetsov, O., Kryvinska, N., Ilchenko, O., Smirnova, T., Ulianovska, Y. «Comparative Analysis of Cryptocurrency Trading Platforms Using the Analytic Hierarchy Process». *CEUR Workshop Proceedings*, 2023, 3628, pp. 106-115.

12. Akhalaia, G., Iavich, M., Iashvili, G., Prysiazhnyy, D., Smirnova, T. «Secure Encrypted Connection on Georgian Website». *CEUR Workshop Proceedings,* 2023, 3550, pp. 313-320.

13. Al-Mudhafar Aqeel, A.M., Smirnova, T., Buravchenko, K., Smirnov, O. «The method of assessing and improving the user experience of subscribers in software-configured networks based on the use of machine learning». *Advanced Information Systems*, 2023, 7(2), pp. 49-56

14. Smirnov, O., Sydorenko, V., Aleksander, M., Zhyharevych, O., Yenchev, S. «Simulation of the cloud IoT-based monitoring system for critical infrastructures». *CEUR Workshop Proceedings,* Volume 3530, 2023, pp. 256-265.

15. Kuznetsov, O., Kandiy, S., Frontoni, E., Smirnov, O. «Trade-offs in Post-Quantum Cryptography: A Comparative Assessment of BIKE, HQC, and Classic

McEliece». *CEUR Workshop Proceedings,* Volume 3504, 2023, pp. 1-11.

16. Smirnov, O., Lakhno, V., Akhmetov, B., Chubaievskyi, V., Khorolska, K., Bebeshko, B. «Selection of a Rational Composition of İnformation Protection Means Using a Genetic Algorithm». *In: Rajakumar, G., Du, KL., Vuppalapati, C., Beligiannis, G.N. (eds) Intelligent Communication Technologies and Virtual Mobile Networks. Lecture Notes on Data Engineering and Communications Technologies*, vol 131. 2023. Springer, Singapore. pp. 21-34.

17. Смірнова Т.В., Гнатюк С.О., Бердибаєв Р.Ш., Сидоренко В.М., Жигаревич О.К., «Система корелювання подій та управління інцидентами кібербезпеки на об'єктах критичної інфраструктури». *Кібербезпека: освіта, наука, техніка*, №*3*(19), 2023, С. 176-196.

18. Смірнов О.А. Козлов Я.О., Смірнова Т.В. «Дослідження застосування SIEM-систем для забезпечення кібербезпеки та захисту інформації». *ІІ Міжнародна науково-практична Інтернет-конференція «Інновації та перспективні шляхи hозвитку інформаційних технологій (ІПШРІТ-2023)»* м.Черкаси 6 грудня 2023 року – Черкаси: ЧДТУ.– 2023. – С.251-252.

19. Козлов Я.О., Смірнова Т.В., Смірнов О.А. «Дослідження SIEM-систем для забезпечення кібербезпеки». *VII міжнародна науково-практична конференція "Інформаційна безпека та комп'ютерні технології" до 30-ти річчя кафедри кібербезпеки та програмного забезпечення,* м. Кропивницький. 1 листопада 2023 р. – Кропивницький: ЦНТУ. – 2023. – С. 26.

20. Козлов Я.О., Козірова Н.Л., Смірнов О.А. «Дослідження структури та принципу роботи SIEM-системи». *VII міжнародна науково-практична конференція "Інформаційна безпека та комп'ютерні технології" до 30-ти річчя кафедри кібербезпеки та програмного забезпечення,* м. Кропивницький. 1 листопада 2023 р. – Кропивницький: ЦНТУ. – 2023. – С. 59.

21. Вінтенко Б.Ю., Смірнов О.А., Коваленко О.В., Смірнов С.А., Коваленко А.С. «Дослідження нормативних документів та галузевих стандартів розробки програмного забезпечення комп'ютерних систем управління АЕС,

важливих для безпеки». *Системи управління, навігації та зв'язку*, 2023, вип. 2(72), С. 170-178.

22. Smirnov, O., Neskorodieva, T., Fedorov, E., Rudakov, K., Neskorodieva, A. «Method Detection Audit Data Anomalies on Basis Restricted Cauchy Machine» *CEUR Workshop Proceedings,* Volume 3187, 2022,

23. Smirnov O.A., Al-Oraiqat A.M., Ulichev O.S., Meleshko Ye.V., Al-Rawashdeh H.S., Polishchuk L.I. «Modeling strategies for information influence dissemination in social networks». *Journal of Ambient Intelligence and Humanized Computing* Volume 13, Issue 5. Springer, Cham. 2022, pp. 2463-2477.

24. Смірнова Т.В., Гнатюк С.О., Сидоренко В.М., Юдін О.Ю., Сидоренко С.Ю., «Модель визначення критичності галузевих інформаційно-телекомунікаційних систем». *Проблеми інформатизації та управління,* № 2(70). 2022. С. 28-37.

25. Смірнов О.А., Смірнова Т.В., Якименко Н.М., Смірнов С.А., Поліщук Л.І., «Дослідження стійкості до диференціального криптоаналізу запропонованої функції гешування удосконаленого модуля криптографічного захисту в інформаційно-комунікаційних системах» *Системи управління, навігації та зв'язку,* 2022, № 3(69). С. 93-98.

26. Смірнов О.А., Смірнова Т.В., Якименко Н.М., Поліщук Л.І., Смірнов С.А. «Дослідження статистичної стійкості та швидкісних характеристик запропонованої функції гешування удосконаленого модуля криптографічного захисту в інформаційно-комунікаційних системах» *Вісник Хмельницького національного університету. Серія: «Технічні науки»,* № 2 (307). С. 46-52. 2022.

27. Смірнов О.А., Смірнова Т.В., Константинова Л.В., Смірнов С.А., Якименко Н.М., «Дослідження стійкості до лінійного криптоаналізу запропонованої функції гешування удосконаленого модуля криптографічного захисту в інформаційно-комунікаційних системах» *Системи управління, навігації та зв'язку*, 2022, № 1(67). С. 84-89.

28. Smirnov O., Kuznetsov A., Zhora V., Onikiychuk A., Pieshkova O.

«Hiding Messages in Audio Files Using Direct Spread Spectrum». *11th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications, IDAACS 2021,* Cracow, Poland, 22-25 September 2021. P. 414-418

29. Smirnov O., Kuznetsov A., Lokotkova I., Kuznetsova T., Florov S., Lebid O. «Using Orthogonal Signals to Hide Information in Images». *4 IEEE International Conference on Advanced Information and Communication Technologies (AICT) – 2021*, Lviv, Ukraine, September 21-25, 2021. P. 255-260.

30. Smirnov O., Kuznetsov A., Girzheva O., Kiian A., Nakisko O., Kuznetsova T. «Advanced Code-Based Electronic Digital Signature Scheme». *2020 IEEE International Conference on Problems of Infocommunications Science and Technology, PIC S and T 2020,* Kharkiv, 6 October 2020-9 October 2020, P. 358-362.

31. Smirnov O., Kuznetsov A., Kiian A., Kuznetsova K. «Data hiding scheme based on spread sequence addressing». *CEUR Workshop Proceedings* Volume 2805, 2020, Pages *44-58.*

32. Smirnov, O., Kuznetsov, A., Potii, O., Poluyanenko, N., Stelnyk, I., Mialkovsky, D. «Combining and filtering functions in the framework of nonlinear-feedback shift register». *International Journal of Computing*; 2020, Volume 19, Issue 2 – Research Institute for Intelligent Computer Systems – 2020. – P. 247-256.

33. Smirnov O., Kuznetsov A., Kiian A., Kuznetsova T. «Non-binary constant weight coding technique». *CEUR Workshop Proceedings*. Volume 2740, 2020, Pages 102-114.

34. Smirnov O., Alimseitova Zh., Adranova A., Akhmetov B., Lakhno V., Zhilkishbayeva G. «Models and algorithms for ensuring functional stability and cybersecurity of virtual cloud resources». *Journal of theoretical and applied information technology* Vol.98. No 21, 2020, P. 3334-3346.

35. Smirnov O., Kuznetsov A., Arischenko A., Chepurko I., Onikiychuk A., Kuznetsova T. «Pseudorandom sequences for spread spectrum image steganography». *CEUR Workshop Proceedings* Volume 2654, 2020, Pages 122-131.

36. Smirnov O., Kuznetsov A., Kovalchuk D., Kuznetsova T. «New technique for data hiding in cover images using adaptively generated pseudorandom sequences». *CEUR Workshop Proceedings* Volume 2654, 2020, Pages 1-14.

37. Smirnov O., Lutsenko M., Kuznetsov A., Kiian A., Kuznetsova T., «Biometric cryptosystems: overview, state-of-the-art and perspective directions». *Lecture Notes in Networks and Systems*, vol 152. Springer, Cham. 2021, pp 66-84.

38. Smirnov O., Kuznetsov A., Onikiychuk A., Makushenko T., Anisimova O., Arischenko A. «Adaptive pseudo-random sequence generation for spread spectrum image steganography». *2020 IEEE 11th International Conference on Dependable Systems, Services and Technologies (DESSERT)*, Ukraine, Kyiv, May 14-18. 2020. P. 161-165.

39. Smirnov O., Kuznetsov A., Kiian A., Babenko V., Perevozova I.,Chepurko I. «New Approach to the Implementation of Post-Quantum Digital Signature Scheme». *2020 IEEE 11th International Conference on Dependable Systems, Services and Technologies (DESSERT),* Ukraine, Kyiv, May 14-18. 2020. P. 166-171.

40. Smirnov O., Kuznetsov A., Kiian A., Cherep A., Kanabekova M., Chepurko I. «Testing of code-based pseudorandom number generators for post-quantum application». *2020 IEEE 11th International Conference on Dependable Systems, Services and Technologies (DESSERT),* Ukraine, Kyiv, May 14-18. 2020. P. 172-177.

41. Smirnov O., Kuznetsov A., Pushkar'ov A., Serhiienko R., Babenko V., Kuznetsova T., «Representation of Cascade Codes in the Frequency Domain». In: Radivilova T., Ageyev D., Kryvinska N. (eds) Data-Centric Business and Applications. Lecture Notes on Data Engineering and Communications Technologies, vol 48. Springer, Cham. 2021. pp 557-587.

42. Smirnov, O., Markovets, O. Vovk, N., Turchyn, Y., «Model of informational support for social network administrators' content creation». *CEUR Workshop Proceedings* Volume 2616, 2020, Pages 125-136.

43. Smirnov, O., Shekhanin, K., Kuznetsov, A., Krasnobayev, V.

«Detecting Hidden Information in FAT». *International Journal of Computer Network and Information Security (IJCNIS)*. Vol. 12, No. 3, 2020. PP.33-43.

44. Smirnov, O., Kuznetsov, A., Gorbacheva, L., Babenko, V., «Hiding data in images using a pseudo-random sequence», *CEUR Workshop Proceedings* Volume 2608, 2020, Pages 646-660.

45. Smirnov, O., Kuznetsov, A., Kolovanova, I., Kuznetsova, T., «Noise immunity of the algebraic geometric codes». International Journal of Computing; 2019, Volume 18, Issue 4 – Research Institute for Intelligent Computer Systems – 2019. – P. 393-407.

46. Smirnov, O., Kuznetsov, A., Reshetniak, O., Ivko, N., Katkova, T., Kuznetsova, T., «Generators of Pseudorandom Sequence with Multilevel Function of Correlation». *2019 IEEE International Scientific-Practical Conference Problems of Infocommunications, Science and Technology (PIC S&T),* Kyiv, Ukraine, 8 – 11 October 2019 . P.517-522.

47. Smirnov, O., Ulichev, O., Meleshko, Y., Khokh, V., Goncharenko, I. «Method of Choosing Objects for Informational Influence in Social Networks during Information Campaign Based on the Analytic Hierarchy Process». *CEUR Workshop Proceedings*, Vol 2588, P. 215-227, 2019.

48. Smirnov, O., Krasnobayev, V., Yanko, A., Kuznetsova, T. «Methods of nulling numbers in the system of residual classes». *CEUR Workshop Proceedings*, Vol 2588, P. 90-106, 2019.

49. Smirnov, O., Kuznetsov, A., Kiian, A., Gorbenko, Y., Cherep, O., Bexhter L. «Code-based Pseudorandom Generator for the Post-Quantum Period», *2019 IEEE International Conference on Advanced Trends in Information Theory (IEEE ATIT 2019).* 18.12.19-20.12.19 Kyiv Ukraine. P. 204 – 209.

50. Smirnov, O., Kuznetsov, A., Nariezhnii, O., Stelnyk, S., Kokhanovska, T., Kuznetsova T., «Side Channel Attack on a Quantum Random Number Generator», *10th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications, IDAACS 2019*; Metz; France; 18 - 21 September 2019. P.713-718.