НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»
ФІЗИКО-ТЕХНІЧНИЙ ІНСТИТУТ
Кафедра інформаційної безпеки

«До захисту допущено»
В.о. завідувача кафедри

_____ М.В.Грайворонський
(підпис)
“___” _____2019 р.

Дипломна робота

на здобуття ступеня бакалавра

з напряму підготовки 6.170101 «Безпека інформаційних і комунікаційних систем»

на тему: Метод оцінки систем виявлення вторгнень _____

_____

Виконав (-ла): студент (-ка) _4__ курсу, групи __ФБ-52___
(шифр групи)

_____Коваль Тарас _____ _____
(прізвище, ім’я, по батькові) (підпис)

Керівник_____доц.каф.ІБ - Родіонов А.М _____ _____
(посада, науковий ступінь, вчене звання, прізвище та ініціали) (підпис)

Консультант _____ _____ _____
(назва розділу) (посада, вчене звання, науковий ступінь, прізвище, ініціали) (підпис)

Рецензент_____ _____
(посада, науковий ступінь, вчене звання, науковий ступінь, прізвище та ініціали) (підпис)

Засвідчую, що у цій дипломній роботі немає запозичень з праць інших авторів без відповідних посилань.
Студент _____
(підпис)

Київ - 2019 року
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ

«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»
ФІЗИКО-ТЕХНІЧНИЙ ІНСТИТУТ
Кафедра інформаційної безпеки

Рівень вищої освіти – перший (бакалаврський)

Напрям підготовки 6.170101 «Безпека інформаційних і комунікаційних систем»

ЗАТВЕРДЖУЮ
В.о. завідувача кафедри
_____ М.В.Грайворонський
(підпис)
«___»_____2019 р.

ЗАВДАННЯ
на дипломну роботу студенту

_____Коваль Тарас_____
(прізвище, ім'я, по батькові)

1. Тема роботи Метод оцінки систем виявлення вторгнень _____
_____ ,
науковий керівник роботи_____доц.каф.ІБ - Родіонов А.М _____
_____,
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)
затверджені наказом по університету від «____ » 2019 р. № _____

2. Термін подання студентом роботи 10 червня 2019 р.

3. Вихідні дані до роботи _____
_____
_____

4. Зміст роботи_____
_____
_____
_____
_____

5. Перелік ілюстративного матеріалу (із зазначенням плакатів, презентацій тощо) _____
_____
_____
_____

6. Дата видачі завдання _____

Календарний план

№ з/п	Назва етапів виконання дипломної роботи	Термін виконання етапів дипломної роботи	Примітка

Студент _____ _____
(підпис) (ініціали, прізвище)

Науковий керівник роботи _____ _____
(підпис) (ініціали, прізвище)

РЕФЕРАТ

Робота обсягом 64 сторінки містить 5 ілюстрацій, 4 таблиці та 21 літературне посилання.

Метою роботи є оцінка Систем виявлення вторгнень за допомогою методу, який буде розроблений в ході даної дипломної роботи.

Завданням роботи є побудова методу оцінки Систем виявлення вторгнень; розробка математичних методів що будуть застосовані при оцінці СВВ; а також апробація даного методу на готових програмних рішеннях.

Об’єктом дослідження є Системи виявлення вторгнень.

Предметом дослідження є захищеність та ефективність Систем виявлення вторгнень.

Результати роботи викладені у вигляді таблиці та методу, що демонструє оцінку обраних для аналізу Систем виявлення вторгнень згідно запропонованого методу.

Результати роботи можуть бути використані для вибору Системи виявлення вторгнень. Також можна використовувати представлений метод для оцінки систем виявлення вторгнень та порівняння з результатами оцінки захищеності інших систем виявлення вторгнень.

СИСТЕМА ВИЯВЛЕННЯ ВТОРГНЕНЬ, СИСТЕМА ЗАПОБІГАННЯ ВТОРГНЕННЯМ, БЕЗПЕКА ІНФОРМАЦІЙНИХ І КОМУНІКАЦІЙНИХ СИСТЕМ

РЕФЕРАТ

Работа объемом 64 страницы содержит 5 иллюстраций, 4 таблицы и 21 литературное ссылки.

Целью работы является оценка Систем обнаружения вторжений с помощью метода, который будет разработан в ходе данной дипломной работы.

Задачей работы является построение метода оценки Систем обнаружения вторжений (СОВ); разработка математических методов которые будут применены при оценке СОВ; а также апробация данного метода на готовых программных решениях.

Объектом исследования является Системы обнаружения вторжений.

Предметом исследования является защищенность и эффективность Систем обнаружения вторжений.

Результаты работы изложены в виде таблицы и метода, демонстрирующий оценку избранных для анализа Систем обнаружения вторжений согласно предложенного метода.

Результаты работы могут быть использованы для выбора Системы обнаружения вторжений. Также можно использовать представлен метод для оценки систем обнаружения вторжений и сравнение с результатами оценки защищенности других систем обнаружения вторжений.

СИСТЕМА ОБНАРУЖЕНИЯ ВТОРЖЕНИЙ, СИСТЕМА ПРЕДОТВРАЩЕНИЯ ВТОРЖЕНИЙ, БЕЗОПАСНОСТЬ ИНФОРМАЦИОННЫХ И КОММУНИКАЦИОННЫХ СИСТЕМ

ABSTRACT

The work includes 64 pages, contains 5 illustrations, 4 tables and 21 literary references.

The purpose of the work is to evaluate the Intrusion Detection Systems using a method that will be developed during this work.

The task of the work is to construct a method for estimating Intrusion Detection Systems; development of mathematical methods that will be used in the estimation of IDS; as well as approbation of this method on existing software solutions.

The object of the research is Intrusion Detection Systems.

The subject of the research is the security and effectiveness of Intrusion Detection Systems.

The results of the work are presented in the form of a table and a method that demonstrates the evaluation of the selected intrusion detection systems according to the proposed method.

The results of the work can be used to select the Intrusion Detection System. You can also use the provided method to evaluate intrusion detection systems and compare them with the evaluation results of other intrusion detection systems.

INTRUSION DETECTION SYSTEM, INTRUSION PREVENTION SYSTEM, SAFETY OF INFORMATION AND COMMUNICATION SYSTEMS

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

Система виявлення вторгнень (СВВ) (англ. Intrusion Detection System, IDS) - програмний або апаратний засіб, призначений для виявлення фактів несанкціонованого доступу до комп'ютерної системи чи мережі або несанкціонованого управління ними в основному через Інтернет. Відповідний англійський термін - Intrusion Detection System (IDS). Системи виявлення вторгнень забезпечують додатковий рівень захисту комп'ютерних систем.

Автоматизована система 3 рівня (АС-3) - це комплекс який складається з багатьох комп’ютерів та розрахований на багато користувачів, а також він об'єднаний в локальну мережу (або кілька мереж) і має доступ до мережі Інтернет. У такій системі, зазвичай, циркулює як інформація з обмеженим доступом, так і відкрита інформація. Призначення – зберігання, обробка, передача інформації.

Атака — детально підібраний набір дій, які, в разі успіху, призведуть або до пошкодження ресурсів АС-3 або до небажаної операції.

Система запобігання вторгнень (СЗВ) (англ. Intrusion Prevention System, IPS) - програмна або апаратна система мережевої та комп'ютерної безпеки, що виявляє вторгнення або порушення безпеки і автоматично захищає від них.

Системи IPS можна розглядати як розширення Систем виявлення вторгнень (IDS), так як завдання відстеження атак залишається однаковою. Однак, вони відрізняються в тому, що IPS повинна відстежувати активність в реальному часі і швидко реалізовувати дії щодо запобігання атак.

ВСТУП

Більшість сучасних компаній є Автоматизованими системами 3 рівня (АС-3), тобто це комплекс який складається з багатьох комп’ютерів та розрахований на багато користувачів, а також він об'єднаний в локальну мережу (або кілька мереж) і має доступ до мережі Інтернет. У такій системі, зазвичай, циркулює як інформація з обмеженим доступом, так і відкрита інформація.

З технічної точки зору АС-3 являє собою локальну мережу (або кілька мереж) з персональних комп'ютерів не в захищеному виконанні і серверів. Включає в себе ряд керованих комутаторів, маршрутизаторів, систему виявлення атак, апаратний (або програмний) фаєрвол, комплекс додаткового ПЗ (антивірус, засоби розмежування доступу і т.д.), за потребою ще й комплекс засобів технічного і / або криптографічного захисту інформації.

Згідно Закону України "Про захист інформації в інформаційно-телекомунікаційних системах" від 31.05.2005 р., а саме згідно статті 8 цього Закону - Умова обробки інформації в системі: `Інформація, яка є власністю держави, або інформація з обмеженим доступом, вимога щодо захисту якої встановлена законом, повинна оброблятися в системі із застосуванням комплексної системи захисту інформації з підтвердженою відповідністю. `

Комплексна система захисту інформації - взаємопов'язана сукупність організаційних та інженерно-технічних заходів, засобів і методів захисту інформації.

Підсумовуючи, АС-3 це комплекс що складається з обладнання для обробки інформації та з обладнання яке захищає інформацію, що обробляється.

Одним з важливих елементів захисту інформації - це захист мережі за допомогою так званої ‘Системи виявлення вторгнень’ (СВВ). Очевидною є важливість використання найбільш надійних СВВ які забезпечують максимальний рівень захищеності системи.

Для сучасного системного адміністратора не очевидно, яке програмне рішення обрати, якщо основним критерієм є надійність і захищеність майбутньої системи. Велика кількість СВВ та відсутність методу їх порівняння є серйозною проблемою.

Актуальність роботи зумовлюється тим, що системи виявлення вторгнень в наш час стали одним з основних інструментів для захисту від атак. Розробнику надзвичайно складно обрати певне програмне рішення, коли важливим аспектом вибору є захищеність. Відсутній метод, що дозволяє оцінити рівень захищеності та ефективності сучасних систем виявлення вторгнень. Представлена робота пропонує такий метод.

Метою роботи є оцінка Систем виявлення вторгнень (СВВ) за допомогою методу, який буде розроблений в ході даної дипломної роботи.

Для досягнення даної мети було поставлено такі завдання:

- Огляд теоретичної інформації стосовно СВВ
- Огляд способів обходу СВВ
- Побудова методу оцінки СВВ
- Апробація даного методу на готових програмних рішеннях.

Об’єктом дослідження є Системи виявлення вторгнень.

Предметом дослідження є захищеність та ефективність Систем виявлення вторгнень.

Наукова новизна підтверджується тим, що в результаті роботи був запропонований метод оцінки СВВ.

Практичне значення полягає в тому, що результати роботи можуть бути застосований на етапі проектування системи захисту, що розробляються для використання у корпоративних та державних інформаційних системах. При виборі системи виявлення вторгнень з декількох, розробник матиме змогу оцінити захищеність та ефективність кожного за допомогою представленого методу. Метод є готовим для застосування, а так легко може бути розширений.

Метод дослідження базується на шести критеріях.

Детальна аргументація важливості кожного критерію наведена в розділах 3.2.1-3.2.6. В якості критеріїв використовувались особливості деяких Систем виявлення вторгнень, наприклад Suricata, які розглядались як конкуренто спроможна перевага. Набір цих переваг було виділено і до них були додані критерії, що значно полегшують життя кінцевому користувачу програмного продукту, такі як наявність документації та можливість зв'язатися з розробниками продукту напряму.

Кожному з цих критеріїв надається оцінка з точки зору повноти реалізації у продукті. Підсумкова оцінка захищеності продукту є сумою оцінок повноти реалізації таких критеріїв:

- Виявлення потенційних атак
- Можливість використання сигнатур спроектованих для використання в інших СВВ
- Підтримка роботи в багато поточному режимі
- Кількість рівнів моделі OSI доступних для перевірки
- Наявність документації
- Можливість зв'язатися з розробниками продукту напряму

1 СИСТЕМИ ВИЯВЛЕННЯ ВТОРГНЕНЬ

1.1 Введення

Система виявлення вторгнень (СВВ) - програмний або апаратний засіб, призначений для виявлення фактів несанкціонованого доступу до комп'ютерної системи чи мережі або несанкціонованого управління ними в основному через Інтернет. Відповідний англійський термін - Intrusion Detection System (IDS). Системи виявлення вторгнень забезпечують додатковий рівень захисту комп'ютерних систем.

Системи виявлення вторгнень використовуються для виявлення деяких типів шкідливої активності, яка може негативно вплинути на безпечність комп'ютерної системи. До такої активності відносяться мережеві атаки проти вразливих сервісів, атаки, спрямовані на підвищення привілеїв, неавторизований доступ до важливих файлів, а також дії шкідливого програмного забезпечення (комп'ютерних вірусів, троянів і черв'яків).

В кінці 90-х років Агентство прогресивних захисних проектів (Агентство просунутих дослідницьких проектів) зробило спробу структурувати складові елементи СВВ. Свою схему вони назвали Загальною системою виявлення вторгнень (Common Intrusion Detection Framework (cidf)).

За цією схемою архітектура СВВ включає (Рисунок 1.1):

- сенсорну підсистему, призначену для збору подій, пов'язаних з безпекою системи що захищається (E-boxes);
- підсистему аналізу, призначену для виявлення атак і підозрілих дій на основі даних сенсорної системи (A-boxes);
- сховище або лог , що забезпечує накопичення подій і результатів аналізу цих подій (D-boxes);

- засоби протидії, які реєструють виявлені потенційно небезпечні події та сигналізують про це або виконують певні маніпуляції щоб зупинити атаку (C-boxes);

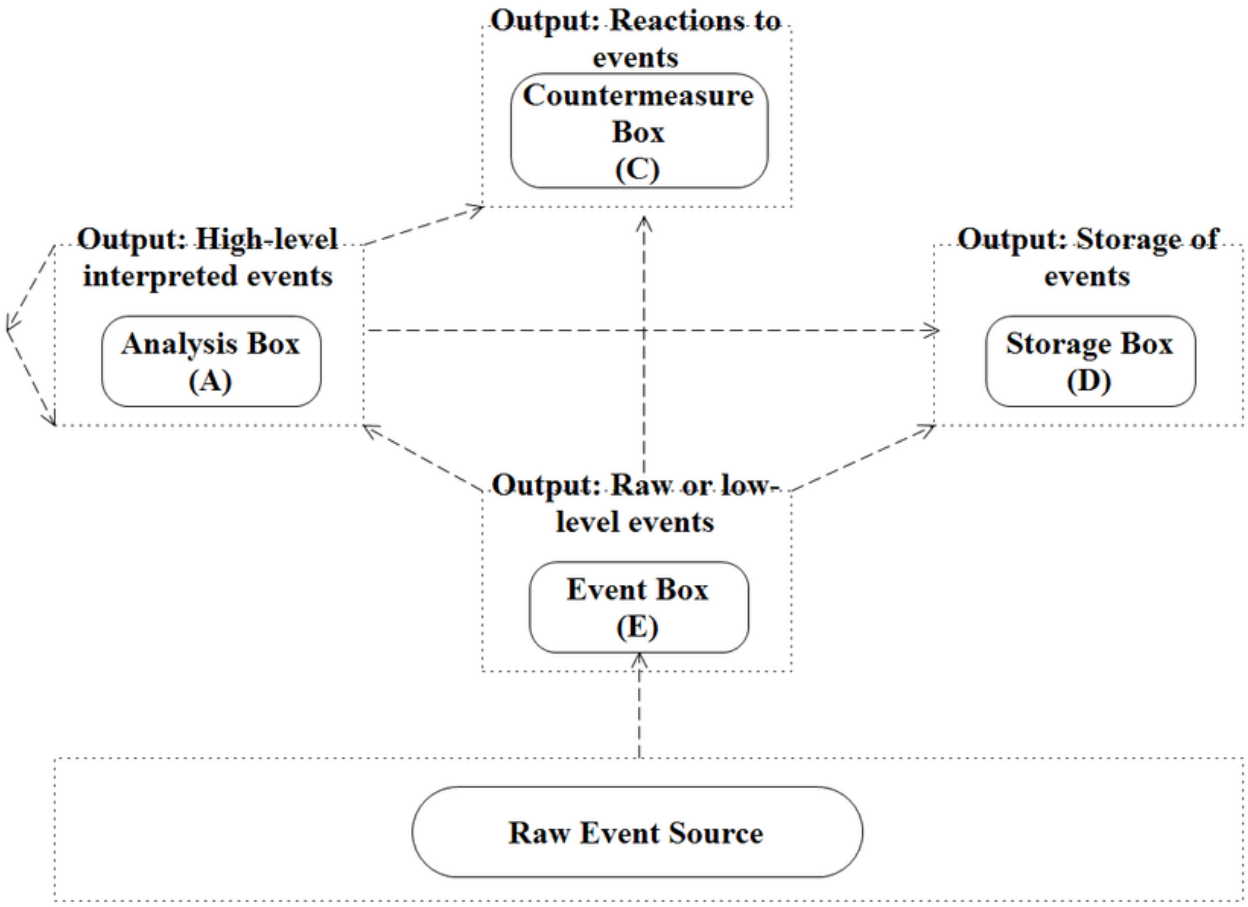

Рисунок 1.1 – Загальна схема роботи Системи виявлення вторгнень

Існує кілька способів класифікації СВВ в залежності від типу і розташування сенсорів, а також методів, використовуваних підсистемою аналізу для виявлення підозрілої активності. У багатьох простих СВВ всі компоненти реалізовані у вигляді одного модуля або пристрою.

Система запобігання вторгнень (англ. Intrusion Prevention System, IPS) - програмна або апаратна система мережевої та комп'ютерної безпеки, що виявляє вторгнення або порушення безпеки і автоматично захищає від них.

Системи IPS можна розглядати як розширення Систем виявлення вторгнень (IDS), так як завдання відстеження атак залишається однаковою. Однак, вони відрізняються в тому, що IPS повинна відстежувати активність в реальному часі і швидко реалізовувати дії щодо запобігання атак.

Різниця в поведінці СВВ/СЗВ систем та фаєрвола

Хоча і СВВ/СЗВ, і фаєрвол відносяться до засобів забезпечення інформаційної безпеки, фаєрвол відрізняється тим, що обмежує надходження на хост або підмережу певних видів трафіку для запобігання вторгнень і не відстежує вторгнення, що відбуваються всередині мережі. СВВ/СЗВ, навпаки, пропускають трафік, аналізуючи його і сигналізуючи при виявленні підозрілої активності (Рисунок 1.2).

Зазвичай поєднують ці дві системи в наступному порядку: зовнішня (незахищена) мережа, фаєрвол, СВВ/СЗВ, внутрішня мережа.

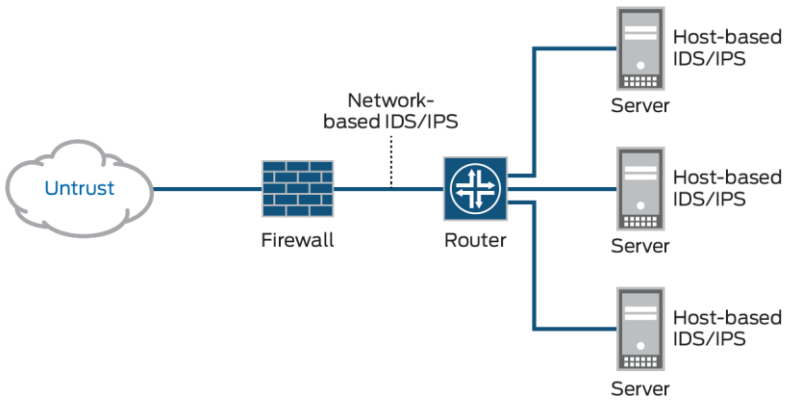

Рисунок 1.2 – Поєднання СВВ/СЗВ систем та фаєрвола

1.2 Історія розробки Систем виявлення вторгнень

Перша концепція СВВ з'явилася завдяки Джеймсу Андерсону і його статті з назвою ‘Моніторинг та спостереження загроз комп'ютерної безпеки’ [1]. У 1984 Фред Коен зробив заяву про те, що кожне вторгнення виявити неможливо і ресурси, необхідні для виявлення вторгнень, будуть рости пропорційно використанню комп'ютерних технологій.

Дороті Деннинг, за сприяння Пітера Неймана, опублікувала статтю з назвою ‘Модель виявлення вторгнень’ в 1986 [2]. В ній вона описала складові елементи моделі СВВ, таким чином сформувавши основу для більшості сучасних систем. Її модель використовувала статистичні методи для виявлення вторгнень і називалася IDES (Intrusion detection expert system - експертна система виявлення вторгнень). Система працювала на робочих станціях Sun і перевіряла як мережевий трафік, так і дані користувачів додатків [3]. IDES використовувала два підходи до виявлення вторгнень: 1. в ній використовувалася система на основі сигнатур для визначення відомих видів вторгнень; 2. система виявлення, яка використовувала статистичні методи для виявлення ‘аномальної’ поведінки, тобто потенційно небезпечної поведінки, аналізуючи профілі користувачів і профіль системи, що охороняється.

Тереза Лунт в статті ‘IDES: Інтелектуальна система для виявлення порушників ‘ [4] запропонувала в доповнення до існуючих двох методів захисту додати використання нейронної мережі як третього компоненту для підвищення ефективності виявлення. Слідом за IDES в 1993 вийшла NIDES

(Next-generation Intrusion Detection Expert System - експертна система виявлення вторгнень нового покоління).

MIDAS (Multics intrusion detection and alerting system), експертна система, написана з використанням мови програмування LISP, була розроблена в 1988 році на основі роботи Деннінга і Неймана. [5] У цьому ж році була розроблена система Haystack, заснована на статистичних методах. [6]

W & S (Wisdom & Sense - мудрість і почуття) - це заснований на статистичних методах детектор аномалій, був розроблений в 1989 році в Національної лабораторії Лос-Аламосу. [7] W & S створював правила на основі статистичного аналізу і потім використовував ці правила для виявлення аномалій.

У 1990, було розроблена система виявлення вторгнень з назвою TIM (Time-based inductive machine). В ній було реалізовано виявлення аномалій з використанням індуктивного навчання, на основі аналізу послідовних патернів користувача на мові Common LISP. Програма була розроблена для комп'ютера VAX 3500. Принципи роботи цієї системи було описані в статті 'Адаптивне виявлення аномалій в режимі реального часу з використанням індуктивно сформованих послідовних шаблонів' [8]. Приблизно в той же час був розроблений NSM (Network Security Monitor - монітор мережевої безпеки), що порівнює матриці доступу для виявлення аномалій на робочих станціях Sun-3/50. [9] У тому ж 1990 році був розроблений ISOA (Information Security Officer's Assistant), що містить в собі ряд стратегій виявлення,

включаючи використання статистики, перевірку профілю та експертну систему. ComputerWatch, розроблений в AT & T Bell Labs, використовував статистичні методи і аналіз сигнатур для скорочення даних аудиту і виявлення вторгнень.

Далі, в 1991, працівники Університету Каліфорнії розробили прототип розподіленої системи DIDS (Distributed intrusion detection system - Розподілена система виявлення вторгнень), яка також була експертної системою [10]. Також в 1991 співробітниками Національної Лабораторії Вбудованих обчислювальних мереж (ICN) була розроблена система NADIR (Network anomaly detection and intrusion reporter - Виявлення мережевих аномалій і сигналізування про вторгнення). На створення цієї системи дуже вплинула робота Деннінга і Люнт з назвою ‘Поетапний підхід до виявлення вторгнень в мережі’ [11]. NADIR використовував заснований на статистиці детектор аномалій і експертну систему.

У 1998 році Національна лабораторія ім. Лоуренса в Берклі представила Bro, що використовує власну мову правил для аналізу даних мережевого трафіку, які були захоплені за допомогою libpcap [12]. NFR (Network Flight Recorder), розроблений в 1999, також працював на основі libpcap [13]. У листопаді 1998 був розроблений APE, аналізатор трафіку, теж використовує libpcap. Через місяць APE був перейменований в Snort. [14]

У 2001 році була розроблена система ADAM IDS (Audit data analysis and mining IDS). Система використовувала дані tcpdump для створення правил. [15]

З цієї короткої історичної довідки випливає що перші СВВ використовували, так звані, статистичні методи виявлення вторгнень. Нижче наведення класифікація СВВ, в тому числі пояснення що означає словосполучення ‘статистичні методи’.

1.3 Загальна класифікація СВВ

Мережеві СВВ (Network-based IDS, NIDS) розташовуються у таких місцях мережі, де можливий контроль трафіку всіх пристроїв у мережі. Програмне забезпечення перехоплює весь мережевий трафік і аналізує вміст кожного пакета на наявність потенційно шкідливих компонентів. Прикладом мережевої СВВ є Snort.

Протокольні СВВ (Protocol-based IDS, PIDS) являє собою систему, яка відстежує і аналізує комунікаційні протоколи зі зв'язаними системами або користувачами. Для веб-сервера подібна СВВ зазвичай веде спостереження за HTTP і HTTPS протоколами. При використанні HTTPS СВВ повинна розташовуватися на такому інтерфейсі, щоб переглядати HTTPS пакети ще до їх шифрування та надсилання у мережу.

Заснована на **прикладних протоколах СВВ** (Application Protocol-based IDS, APIDS) - це система (або агент), яка веде спостереження і аналіз даних, що передаються з використанням специфічних для певних програм

протоколів. Наприклад, на веб-сервері з SQL базою даних СВВ буде відстежувати вміст SQL команд, що передаються на сервер.

Вузлова СВВ (Host-based IDS, HIDS) - система (або агент), розташована на хості, що відслідковує вторгнення, використовуючи аналіз системних викликів, логів додатків, модифікацій файлів (виконуваних, файлів паролів, системних баз даних), стану хоста і інших джерел . Прикладом є OSSEC. Прикладом системи контролю цілісності файлів, які перевіряють системні файли з метою визначення, коли в них були внесені зміни є Tripwire.

Гібридна СВВ поєднує два і більше підходів до розробки СВВ. Дані від агентів на хостах комбінуються з мережевою інформацією для створення найбільш повного уявлення про безпеку мережі. Як приклад гібридної СВВ можна привести Prelude.

1.4 Класифікація за методом виявлення вторгнень:

Метод аналізу сигнатур.

Цей метод був першим, який був використаний для розпізнавання вторгнень.

Він полягає в наступному: серед трафіку інтернет мережі іде пошук патернів, або рядків за допомогою який були виконані певні атаки. Ключовим в даному методі є те що він захищає лише від відомих атак, оскільки нові

атаки цим методом розпізнати неможливо. Програмні рішення на основі методу аналізу сигнатур оновлюються за принципом антивірусного забезпечення: існують оновлення самого програмного продукту, а крім цього є оновлення бази сигнатур. Від оновлення цієї бази даних безпосередньо залежить безпека системи, оскільки з оновленням система може виявляти нові, до цього невідомі види атак.

Також цей метод використовується в Протокольних СВВ, саме за допомогою нього відбувається перевірка на правильність використання синтаксису певного протоколу.

Метод машинного навчання (або виявлення аномалій).

Цей метод був створений для пошуку і виявлення нових видів атак, стимулом для його розвитку послужили швидкі темпи розвитку і створення нових видів атак.

Метод заснований на машинному навчанні, або більш детально: систему навчають розпізнавати правильну поведінку всередині певної системи; після навчання модель використовують для порівняння нової поведінки на схожість відомим та 'правильним' діям. Оскільки ці системи навчались на основі специфічного набору програмного забезпечення та конфігурації обладнання, ці системи мають більш 'глибокі' знання щодо дозволений дій в системі порівняно з універсальними наборами правил якими користуються системи на основі Методу аналізу сигнатур.

Попри значні переваги системи на основі методів машинного навчання страждають від хибних спрацювань: раніше невідомі, але дозволені дії можуть буть розпізнані як шкідливі, або іншими словами розпізнані як атака.

Також ці системи страждають від значного використання обчислювальних ресурсів, що зменшує їх ефективність. Налаштування чутливості даної системи перед її введенням в експлуатацію значно впливає на її надійність.

Нові види систем виявлення вторгнень що також відносяться до даної категорії є: Аналітика поведінки користувачів і суб'єктів (User and Entity Behavior Analytics (UEBA) (англ.)) та Аналіз мережевого трафіку (Network traffic analysis (NTA) (англ.)). Ці нові види систем вперше було розглянуто в дослідницькій компанії у сфері інформаційних технологій, яка має назву Gartner. Аналіз мережевого трафіку дозволяє виявити не лише зовнішні атаки, а й атаки які йдуть зсередини системи, наприклад від скомпрометованого акаунту користувача або комп'ютера. Аналіз мережевого трафіку поступово набуває все більшого визнання, і деякі організації вже віддають йому перевагу в порівнянні з більш традиційними (старими) СВВ.

1.5 Різниця між СВВ та СЗВ

Визначення СЗВ вже було дане, з метою повторення буде надане коротке визначення.

СЗВ (Intrusion prevention systems (IPS) (англ.)) - можна розглядати як розширення Систем виявлення вторгнень (IDS), які відстежують потенційні атаки в реальному часі і швидко виконують дії щодо запобігання атак.

Тобто, Система виявлення вторгнень (IDS - Intrusion Detection System (англ.)), її ще називають пасивною системою, при виявленні порушення безпеки, інформація про це порушення записується в лог додатку, а також сигнали небезпеки відправляються на консоль і / або адміністратору системи по певному каналу зв'язку.

В активній системі, також відомої як Система Запобігання Вторгнень (IPS - Intrusion Prevention System (англ.)), будуть виконані відповідні дії на порушення, спрямовані для його припинення, наприклад:

- відключення з'єднання;
- налаштування міжмережевого екрану для блокування трафіку від зловмисника;
- блокування IP адрес;
- блокування акаунтів користувачів від яких надходить потенційно небезпечна активність;
- відновлення лог файлів у випадку їх видалення;
- знищення процесів;
- запуск процесів;
- відключення систем;
- запис інформації про потенційну атаку і виконані дії для її припинення.

Як видно з цього списку, СЗВ мають широкий ‘арсенал’ дій та гарантують певний рівень захисту. Проте вони є лише одним зі складових елементів захисту мережі, та мають ряд дій які вони не можуть виконувати:

- запобігти копіювання документів на зовнішні носії;
- змінювати права доступу користувачів;
- контролювати оновленням програмного забезпечення.

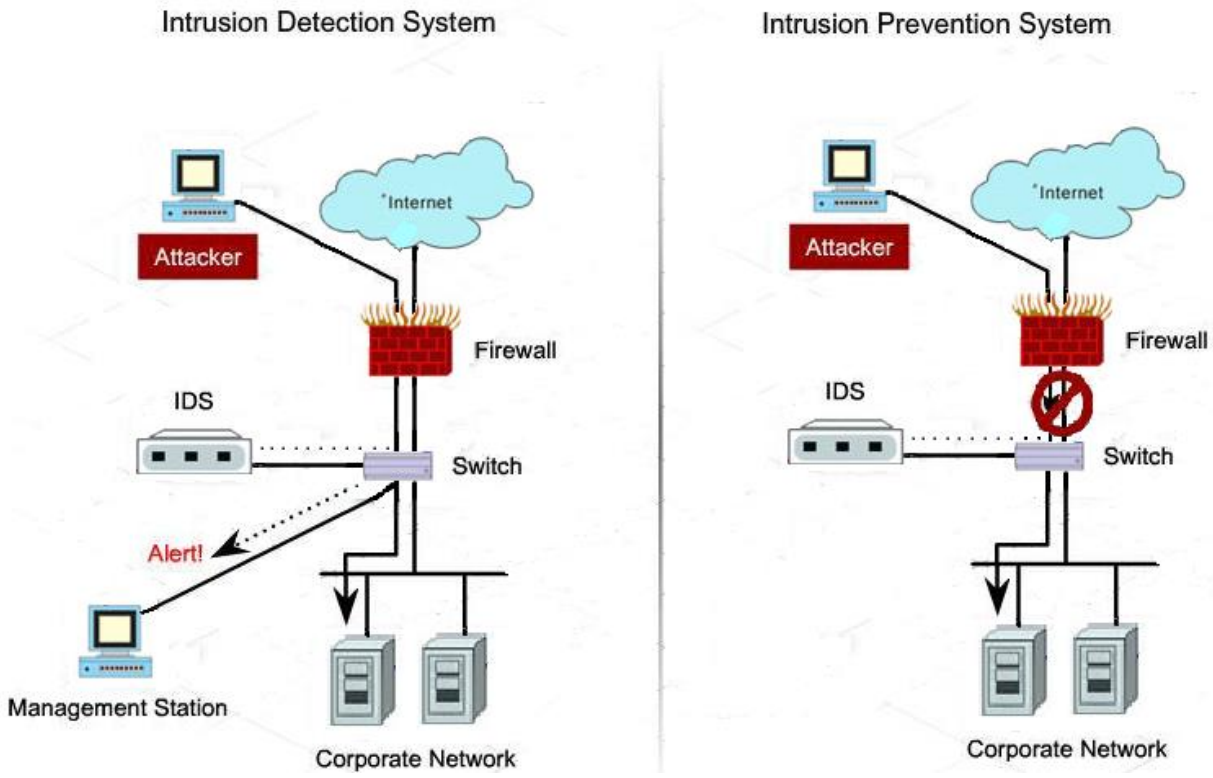

Рисунок 1.3 – Різниця в поведінці IDS та IPS систем

1.6 Класифікація СЗВ

- Мережевого рівня – аналогічно для СВВ, переглядає весь трафік що проходить через мережу та веде пошук потенційних атак, також перевіряє пакети на виконання умов протоколів;

- Рівень бездротової мережі – схоже на попередній пункт, але в цей раз нагляд відбувається за безпровідною мережею;

- Аналіз поведінки мережі – перевіряє мережу на наявність трафіку який не є звичним для даної мережі, наприклад надто сильний трафік може вказувати на DDos (distributed denial of service) атаку;

- Рівень хоста – слідкує за трафіком що проходить через окремий хост, може працювати в купі з мережевою СЗВ для додаткової надійності.

Способи виявлення загроз співпадають зі способами використаними в СВВ, оскільки СЗВ є ‘розширенням’ СВВ.

1.7 Обмеження СВВ та СЗВ

З плином часу можуть придумати нові атаки для певного виду програмного забезпечення. СВВ можуть дізнатись про ці атаки і почати їх виявляти лише отримавши нову базу даних відомих атак. Ефективність СВВ напряму залежить від своєчасного оновлення бази даних. Також необхідно вчасно оновлювати програмне забезпечення, оновлення можуть містити виправлення які забезпечують захист від певного виду атак.

У випадку СВВ на основі виявлення сигнатур існує певний проміжок часу між виявленням певної атаки та оновленням бази даних сигнатур, що дозволить виявити дану атаку. В цей проміжок часу СВВ не здатна виявити цю атаку.

'Шумна' мережа може сильно зменшити ефективність СВВ системи, через велику кількість хибних спрацювань. Під шумною мережею мається на увазі мережа в якій циркулюють пакети що не відповідають вимогам протоколів (такі пакети можуть з'явитись через помилки в програмному забезпеченні).

Зашифровані пакети не обробляються більшістю СВВ. Тобто зашифровані пакети можуть бути використані для атаки, і вона не буде виявлена.

Незважаючи на те що ВСС можуть виявити підозрілу активність, це не значить що вони захистять від слабких вимог до системи аутентифікації або від використання слабких та застарілих протоколів. У випадку отримання доступу через слабку систему аутентифікації, зловмисник може виконати багато шкідливих дій які не будуть розпізнані СВВ, наприклад він може скопіювати конфіденційну документацію до якої має доступ акаунт яким він скористався.

Висновок до розділу 1

В даному розділі наведені основні визначення що стосуються Систем виявлення вторгнень та Систем запобігання вторгненням; перераховані складові елементи з яких побудовано Систему виявлення вторгнень; наведена історична довідка про створення теоретичної бази для розробки СВВ, а також історія зародження та розвитку СВВ.

В даному розділі також наведено класифікації СВВ та СЗВ, як за видами інтернет трафіку який буде проаналізований, так і за способами виявлення потенційних атак в цьому трафіку.

Через значний розвиток атак, Системи виявлення та запобігання вторгненням є важливим елементом захисту для автоматизованих систем 3-го рівня. Вони дозволяють максимально швидко виявити атаку, фактично ще до того як вона починає виконуватися.

Для побудови методу оцінки СВВ необхідно розуміти які існують способи обходу СВВ і як СВВ можуть від них захиститися. Саме цій темі і присвячений наступний розділ.

2 СПОСОБИ ОБХОДУ СВВ ТА СЗВ

Атаки на СВВ в першу чергу спрямовані на їх:

- Доступність – задача згенерувати таку кількість потенційних атак, що система буде нездатна обробити їх, або вивести СВВ з ладу;
- Точність – генерування великої кількості хибних спрацювань;
- Надійність – не виявлення СВВ атаки.

Задача СВВ - це оцінка того, як певна команда може вплинути на систему, тому атаки полягають в генерації такої команди для якої СВВ не може оцінити точний вплив на стан системи.

Деякі команди є доволі гнучкими і їхніх вплив на систему може залежати від багатьох факторів, всі нюанси можуть не бути враховані в СВВ.

Можна використати незвичну комбінацію команд, яка також не була врахована СВВ.

2.1 Короткий перелік основних способів обходу СВВ:

(Більш детально буде йтися нижче)

- Insertion – вставка
- Evasion – ухилення
- Denial of service (Dos) – відмова в обслуговуванні
- Pattern matching weakness - слабкість порівняння зразків
- Encryption and tunneling - Шифрування та тунелювання
- Fragmentation - фрагментація

- Protocol violation – порушення правил протоколу
- File location and integrity - Розташування та цілісність файлів
- Application hijacking - Захоплення програми

2.2 Класифікація атак:

2.2.1 Вставка (Insertion)

Атака при якій СВВ обробляє пакети і пропускає їх далі, в той час як отримувач їх не обробляє. Це може бути спричинено тим що пакет не дійде до одержувача оскільки буде вичерпано час життя пакету, або пакет буде отримано та відкинуто.

На картинці нижче наведено приклад такої атаки.

Перший рядок це початкове повідомлення в необробленому вигляді.

Другий – СВВ розташувала пакети в порядку нумерації і не виявила потенційно небезпечного змісту, який може містити цей набір пакетів.

Третій – система-отримувач відкинула 4 пакет (з певної причини) і результуючий набір пакетів склав слово attack (атака).

Рисунок 2.1 – Приклад insertion атаки

2.2.2 Evasion (ухилення)

Атака ухилення полягає в побудова такого ланцюжка пактів який не буде оброблений СВВ але буде оброблений кінцевою системою-отримувачем.

Цього можна досягти використовуючи фрагментацію пакетів, і СВВ може бути нездатна обробити таку велику кількість пакетів як одну команду і пропустить їх, і адресат отримає їх.

2.2.3 Відмова в обслуговуванні - Denial of service (Dos)

Запуск атаки спрямованої на відмову в обслуговуванні системи IDS є можливим методом обходу одного з механізмів захисту даної мережі. Атакуючий може досягти цього, використовуючи помилки в IDS, споживаючи всі обчислювальні ресурси в IDS, або навмисно запускаючи велику кількість повідомлень, щоб замаскувати фактичну атаку. Різновиди цієї атаки наведені нижче:

Виснаження процесора: пакети, захоплені IDS, зберігаються в буфері ядра, поки CPU не буде готовий до їх обробки. Якщо процесор знаходиться під високим навантаженням, він не може обробляти пакети досить швидко і цей буфер заповнюється. Після цього нові (і, можливо, шкідливі) пакети не можуть бути поміщені у вже заповнений буфер.

Зловмисник може вичерпати ресурси процесора IDS у декілька способів. Наприклад, системи виявлення вторгнень на основі сигнатур використовують алгоритми зіставлення вмісту пакетів з сигнатурами вже відомих атак. Порівняння вмісту з певними сигнатурами є більш дорогою з точки зору обчислень, ніж з іншими. Ця атака заснована на алгоритмічній складності може значно зменшити ефективність IDS відносно невеликою кількістю трафіку.

IDS, який також контролює зашифрований трафік, може витратити велику частину ресурсів ЦП на розшифровку вхідних даних.

Об'єднання цих двох факторів дозволить максимально збільшити використання ЦП, що може призвести до переповнення буферу.

Виснаження пам'яті: для того, щоб виконати перевірку на відповідність певним сигнатурам, IDS зобов'язаний зберігати стан, пов'язаний з підключеннями, які він контролює. Наприклад, IDS має підтримувати "контрольні блоки TCP" (TCBs), фрагменти пам'яті, які містять інформацію, таку як порядкові номери і стани з'єднання (ESTABLISHED, RELATED, CLOSED і т.д.), для кожного з'єднання TCP, що контролюється IDS системі. Як тільки вся оперативна пам'ять IDS (RAM) буде використана, вона буде змушена використовувати файл підкачки - віртуальну пам'ять на жорсткому диску, яка працює набагато повільніше, ніж оперативна пам'ять. Це може

спричинити проблеми з продуктивністю і пропускання потенційно небезпечних пакетів. Ефект подібний до ефекту виснаження процесора.

Якщо IDS не зберігає TCBs правильно і ефективно, зловмисник може вичерпати пам'ять IDS, запустивши велику кількість TCP з'єднань дуже швидко. Подібні атаки можуть бути зроблені шляхом фрагментації великої кількості пакетів у більшу кількість менших пакетів або відправлення великої кількості сегментів TCP в неправильному порядку.

Втома оператора: сповіщення, створені за допомогою IDS, повинні попереджати про початок атаки. Зловмисник може зменшити "доступність" IDS, перевантаживши людського оператора необмеженим числом сповіщень, посилаючи велику кількість "шкідливого" трафіку, призначеного для створення попередження від IDS. Після цього зловмисник може виконати фактичну атаку, використовуючи шум попереднього попередження як прикриття. Для цього були розроблені утиліти "Stick" та "Snot".

2.2.4 Обхід системи розпізнавання шаблонів

Система перевіряє вміст кожного з пакетів на співпадіння з шаблонами атак, у випадку співпадіння генерується попередження.

Цей підхід є проблематичним, оскільки не всі вхідні дані повинні бути однаковими, щоб використати одну й ту саму вразливість. Або іншими словами одна й те сама атакуюча команда може мати різні представлення.

Приклад безкоштовна система виявлення вторгнень Snort має шаблон що перевіряє чи TCP пакет містить в собі рядок "/etc/passwd", тобто чи намагаємось ми якимось чином отримати доступ до файлу що містить паролі:

alert tcp $EXTERNAL_NET any -> $HTTP_SERVERS $HTTP_PORTS (msg:"WEB-MISC /etc/passwd"; flow:to_server,established; content:"/etc/passwd"; nocase; classtype:attempted-recon; sid:1122; rev:5;)

Ми бачимо що цей шаблон є дуже простим, і метод його обходу криється в використанні кодування utf-8, за допомогою якого кожен символ можна представити декількома способами, наприклад:

“/” може бути представлений як U+005C, 0x5C, C191C, E0819C

“A” - U+0041, 0x41, U+100, U+0102, U+0104, U+01CD, U+01DE

Існує 30 унікальних способів представити літеру ‘A’,

34 унікальних способів представити літеру ‘E’,

83,060,640 способів представити рядок “AEIOU”.

Таким чином ми можемо згенерувати велику кількість запитів які матимуть однакове значення для сервера, але всі вони будуть виглядати по різному для СВВ і якийсь із них може бути пропущений нею.

Наприклад рядок "/etc/passwd" буде виглядати наступним чином в utf-8: \x2F\x65\x74\x63\x2F\x70\x61\x73\x73\x77\x64

Також можна використати обхід файлової системи, наприклад зайти в певну папку, потім завдяки команді ‘..’, вийти з неї і після цього зайти в потрібну папку.

Приклад, який є еквівалентним до ‘"/etc/passwd’:

/etc//\/passwd

/etc/rc.d/../../.\passwd

2.2.5 Виконання поліморфного Shellcode

СВВ на основі сигнатур шукають потенційні атаки шляхом пошуку попередньо визначених шаблонів.

Поліморфні атаки - це атаки, в яких кожен екземпляр шкідливого програмного забезпечення відрізняється, але виконує ту ж саму зловмисну дію, намагаючись уникнути виявлення за допомогою сигнатур. Шифрування, перестановка байтів і 'заплутування' коду - це методи перетворення коду, які використовуються для забезпечення того, щоб трафік атаки не збігся з сигнатурою [17].

Поведінкові системи виявлення вторгнень розробляють статистичні та евристичні профілі, які визначають нормальну поведінку під час періоду навчання системи. Розподіл частоти байтів, виникнення послідовності байтів, довжини пакетів і розподіл символів є прикладами статистичних характеристик, включених до профілю СВВ. Після того, як захищувана мережа є вивченою, СВВ сповіщатиме про те, коли характеристики трафіку будуть відхилятися від 'норми' на певне значення [18].

Різновиди поліморфного коду часто виявляються СВВ на основі машинного навчання. Розподіл частот байтів часто буде однаковим для різних випадків атаки. Значення байтів можуть бути різними через заміну або шифрування, але часто існує загальна крива розподілу для одної і тої самої атаки, яка є видозміненою.

Поліморфні атаки видозміни є поліморфними атаками, які створюються відповідно до профілю нормальної поведінки для певної мережі. Шкідливе програмне забезпечення, яке генерує атаку, досліджує мережу для розробки її профілю. Після того, як зловмисне програмне забезпечення розробило

'профіль' для мережі, ПО створює поліморфні екземпляри атакуючих програм, характеристики яких відповідають 'профілю' мережі. Методи змішування дозволяють включати всередину атакуючого ПО вставки, які не будуть виконані, для забезпечення співпадіння кількості байтів і частотної статистики. Однією з таких вставок може бути NOP - асемблерна команда яка фактично нічого не робить. Відкрите ПО 'CLET' є так званим поліморфним двигуном, який використовує байтові вставки і шифрування з ключами різної довжини в спробі збільшити складність виявлення[18].

Змішування нормального і атакуючого трафіку збільшує кількість хибних спрацювань[19]. Хибні спрацювання витрачають ресурси СВВ, що також є позитивним явищем з точки зору атакуючого.

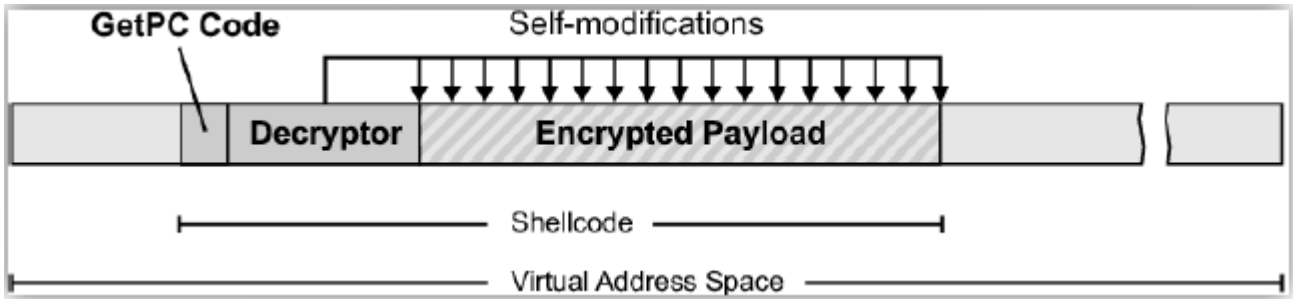

Рисунок 2.2 – Приклад атаки на основі поліморфного Shellcode

2.2.6 Шифрування і тунелювання

Якщо атакуючій стороні вдається встановити зашифроване з’єднання (тунель) з системою, то СВВ цієї системи не буде перевіряти трафік що йде через тунель.

Приклад зашифрованого з’єднання (тунелю):

- SSH
- SSL
- IPSec
- RDP

Два комп’ютера можуть спілкуватися за допомогою IPSec або інших форм зашифрованих тунелів, і NIDS не матиме засобів для перевірки трафіку, якщо не буде надано ключів дешифрування [20].

Деякі зловмисні програми використовують ключ і побітову операцію XOR для шифрування. Враховуючи достатні ресурси, в деяких випадках NIDS може розшифрувати пакети. До переваг шифрування за допомогою операції XOR можна віднести те, що вона є швидкою та простою з обчислювальної точки зору, а найпоширеніші архітектури забезпечують цю функцію на апаратному рівні. Недоліки полягають у тому, що шкідливі програми, зашифровані за допомогою операції XOR, можуть бути виявлені з використанням статистичного аналізу, наприклад частоти байтів. Також, трафік IPSec може бути виявлений за допомогою вимірювань ентропії, хоча архівовані дані можуть мати аналогічні рівні ентропії і призвести до хибних спрацювань [21].

Зважаючи на вище сказане шифрування трафік значно збільшує шанси на виконання атаки, яка не буде виявлена СВВ.

2.2.7 Використання невеликих пакетів

Одним з основних методів є розділення корисного навантаження на кілька невеликих пакетів, так що IDS повинен зібрати потік пакетів для виявлення атаки. Простий спосіб розбиття пакетів полягає в їх фрагментації, але атакуючий може також створювати пакети з невеликими корисними навантаженнями.

Один метод виконання даної атаки - це пауза між відправленнями частин атакуючої команди. Другий метод - відправлення пакетів в неправильному порядку, що може обманути прості IDS системи.

Сама по собі атака фрагментацією є малоефективною, але зазвичай її використовують в поєднанні з іншими атаками.

Приклад Snort для виявлення даної атаки (корисне навантаження пакету має розмір (dsize)=1 і пакет містить лише пробіл):

alert tcp $EXTERNAL_NET any -> $HTTP_SERVERS $HTTP_PORTS (msg:"WEB-MISC whisker space splice attack"; flow:to_server,established; dsize:1; content:" "; reference:arachnids,296; reference:url,www.wiretrip.net/rfp/pages/whitepapers/whiskerids.html; classtype:attempted-recon; sid:1104; rev:11;)

2.2.8 Фрагментація пакетів

Ця атака схожа на атаку з використанням невеликих пакетів, але основна відмінність полягає в тому, що ids повинна зберігати послідовність пакетів в

пам’яті перш ніж зможе порівняти їх з шаблонами. Також система повинна розуміти в якому порядку пакети мають бути зібрані системою-адресатом.

На основі цієї техніки розрізняють наступні атаки:

Накладання фрагментів: використовує принцип того що один фрагмент перезаписує дані з попереднього фрагмента, тим самим чином ухиляючись від розпізнавання IDS системою.

Перезапис фрагменту: ця атака схожа на попередню, відмінність полягає в тому що деякі фрагменти повністю переписують попередні.

Тайм Аут фрагментів: використовує факт що деякі IDS системи утримують незавершену послідовність пакетів в пам’яті певну кількість часу перш ніж відкинути її. В більшості простих систем це 60 секунд. Тобто можна затримати відправлення частини послідовності на цей час і IDS не розпізнає атаку.

Один з спосіб захисту від даної атаки за допомогою Snort це фільтрація пакетів невеликого розміру:

alert ip $EXTERNAL_NET any -> $HOME_NET any (msg:"MISC Tiny Fragments"; dsize:< 25; fragbits:M; classtype:bad-unknown; sid:522; rev:3;)

Висновок до розділу 2

Система виявлення вторгнень є одним з перших бар'єрів, що захищає систему від атак, тобто на СВВ накладаються значні вимоги, такі як: виявлення значної кількості атак, ефективність роботи, зручність використання.

Завдяки тому, що були розглянуті основні способи обходу СВВ, з’явилася можливість побудувати метод, який дозволяє оцінювати Системи виявлення вторгнень.

3 ПОБУДОВА МЕТОДУ ОЦІНКИ IDS СИСТЕМ

3.1 Система оцінки

Для оцінки захищеності та ефективності певної системи можна використати інформацію про наявні атаки, спрямовані на обхід даної системи, методи захисту від цих атак, а також документацію, яка описує аспекти безпеки системи. В даному випадку тип систем що оцінюється, є системи виявлення вторгнень.

Проте, неможливо провести оцінку певної системи, не маючи методу оцінювання, який заснований на чіткому наборі критеріїв.

В даному розділі будуть проаналізовані основні критерії, що мають вплив на захищеність та ефективність СВВ. Буде виділений набір критеріїв що мають найбільший вплив. На їх основі буде побудований метод оцінки Систем виявлення вторгнень. Також даний метод буде апробований на готових програмних рішеннях.

3.2 Критерії для оцінювання

Завдання створення списку критеріїв, на основі яких буде проведена оцінка захищеності та ефективності Систем виявлення вторгнень є доволі складним завданням через значну кількість можливих критеріїв.

Першим і основним критерієм є можливість виявлення більшості атак, спрямованих на обхід систем виявлення вторгнень. Наступний етап - це виділення ряду критеріїв, крім першого, на основі яких буде виконана оцінка систем виявлення вторгнень.

В попередніх розділах були наведені способи обходу СВВ, а також критерії що впливають на їх ефективність. В таблиці 3.1 були виділені найбільш важливі критерії. Також даним критеріям була присвоєна певна оцінка в залежності від рівня важливості кожного з критеріїв. Залежно від рівня того, наскільки добре певний критерій реалізований у програмному рішенні, йому ставиться відповідна оцінка.

Таблиця 3.1 — Перелік критеріїв та їх оцінок

Назва критерію	Максимальна оцінка критерію
1. Виявлення потенційних атак	20
2. Можливість використання сигнатур спроектованих для використання в інших СВВ	5
3. Підтримка роботи в багато поточному режимі	8
4. Кількість рівнів моделі OSI доступних для перевірки	7
5. Наявність документації	5
6. Можливість зв'язатися з розробниками продукту напряму	5
Підсумок (максимальний)	50

Для більш точного порівняння захищеності різних СВВ було введено поняття “оцінки повноти реалізації критерію у продукту” для кожного критерія.

Оцінка повноти реалізації критерію “ 1. Виявлення потенційних атак ” (3.1) визначається як сума оцінок всіх механізмів захисту, що реалізовані у продукті, стандартизована згідно максимальної оцінки критерію.

$$O_1 = \sum_{i=1}^{N} V_i * \frac{O_{1\,max}}{\sum_{i=1}^{N} V_{i\,max}} \quad (3.1)$$

де O_1 — виставлена оцінка критерію, $O_1 \in [\,0\,, O_{1\,max}\,]$;

$O_{1\,max}$ — максимальна оцінка критерію;

V_1 — виставлена оцінка наявності механізму захисту, $V_1 = \{0, V_{1\,max}\}$;

$V_{1\,max}$ — максимальна оцінка наявності механізму захисту;

N — загальна кількість механізмів захисту;

Оцінка повноти реалізації критерію “3.2. Можливість використання сигнатур спроектованих для використання в інших СВВ” (3.2) визначається як експертна оцінка.

$$O_2 \in [0\,, O_{2\,max}] \quad (3.2)$$

де O_2 — виставлена оцінка критерію;

$O_{2\,max}$ — максимальна оцінка критерію;

Оцінка повноти реалізації критерію “3. 3. Підтримка роботи в багато поточному режимі ” (3.3) визначається як експертна оцінка повноти використання багато поточного середовища.

$$O_3 \in [0\ ,0_{3\ max}] \tag{3.3}$$

де O_3 — виставлена оцінка критерію;

$O_{3\ max}$ — максимальна оцінка критерію;

Оцінка повноти реалізації критерію “ 3.4. Кількість рівнів моделі OSI доступних для перевірки ” (3.4) визначається абсолютним значенням кількості рівнів доступних для перевірки IDS.

$$O_4 \in [0\ ,0_{4\ max}] \tag{3.4}$$

де O_4 — виставлена оцінка критерію;

$O_{4\ max}$ — максимальна оцінка критерію;

Оцінка повноти реалізації критерію “Наявність документації” (3.5) визначається як експертна оцінка наявної документації.

$$O_5 \in [0\ ,0_{5\ max}] \tag{3.5}$$

де O_5 — виставлена оцінка критерію;

$O_{5\ max}$ — максимальна оцінка критерію;

Оцінка повноти реалізації критерію “ Можливість зв'язатися з розробниками продукту напряму” (3.6) визначається як експертна оцінка наявних контактів.

$$O_6 \in [0\ , O_{6\ max}] \tag{3.6}$$

де O_6 — виставлена оцінка критерію;

$O_{6\ max}$ — максимальна оцінка критерію;

Підсумкова оцінка захищеності Систем виявлення вторгнень згідно представленого методу (3.7) визначається як сума оцінок повноти реалізації кожного критерію.

$$O = O_1 + O_2 + O_3 + O_4 + O_5 + O_6 \tag{3.7}$$

3.2.1 Виявлення потенційних атак

При проектуванні систем виявлення вторгнень, їх розробники використовують сигнатури, які виявляють максимально велику кількість атак з усіх можливих. При швидкій появі нових атак розробники можуть не встигати створювати відповідні сигнатури, для виявлення цих атак. Також наявні сигнатури можуть не виявляти всі модифікації певного типу атак.

Тому даний критерій є важливим оскільки він дозволяє оцінити відсоток атак що були виявлені системою виявлення вторгнень.

Деякі з цих атак можуть вважатися зайвими через “застарілість” та інші причини, але зловмисники не один раз демонстрували свою винахідливість та використовували під час атак й “найстаріші” методи. З цього можна зробити

висновок, що чим більша кількість типів атак може бути виявлена системою виявлення вторгнень, тим краще.

Для оцінки даного критерію запропоновано поділити атаки на наступні групи:

1. Атаки на основі фрагментованих пакетів
2. Атаки на основі використання поліморфного Shellcode
3. Інші основні методи обходу систем виявлення вторгнень

Кожна з перерахованих груп загроз включає в себе велику кількість індивідуальних типів атак, що має свої особливості.

Таблиця 3.2 – Типи атак для кожної групи

Група атак	Типи атак в групі
Атаки на основі фрагментованих пакетів	*Ping of death *Nestea Attack 1/3 *Nestea Attack 2/3 *Nestea Attack 3/3
Атаки на основі використання поліморфного Shellcode	*SHELLCODE ** sparc setuid 0 *SHELLCODE x86 setgid *SHELLCODE IRIX SGI + NOOP *SHELLCODE x86 setgid 0 && SHELLCODE x86 setuid 0 *OVERFLOW attempt *SHELLCODE x86 setuid 0 *win32_bind_dllinject - EXITFUNC=she DLL=c:\ LPORT=44 *win32_bind_dllinject - EXITFUNC=seh DLL=c:\ LPORT=44 *win32_bind - EXITFUNC=seh LPORT=4444 Size=709 Encode *db "cmd.exe /c net user USERNAME PASSWORD /ADD && n *Cisco: Creates a new VTY, allocates a password then *Rothenburg Shellcode *Mainz/Bielefeld Shellcode

Кінець Таблиці 3.2

Інші основні методи обходу систем виявлення вторгнень	*Nmap decoy test (6th position) *Nmap decoy test (7th position) *Hex encoding *Nmap scan with fragmentation *Nikto Random URI encoding *Nikto Directory self-reference *Nikto Premature URL ending *Nikto Prepend long random string *Nikto Fake parameter *Nikto TAB as request spacer *Nikto Change the case of the URL *Nikto Windows directory separator *Nikto Carriage return as request spacer *Nikto Binary value as request spacer *JavaScript Obfuscation

Таблиця 3.3 – Важливість захисту від кожної групи загроз

Група атак	Оцінка важливості
Атаки на основі фрагментованих пакетів	2
Атаки на основі використання поліморфного Shellcode	5
Інші основні методи обходу систем виявлення вторгнень	8

3.2.2 Можливість використання сигнатур спроектованих для використання в інших СВВ

Ефективність роботи системи виявлення вторгнень на основі сигнатур значно залежить від якості самих сигнатур. Компанії які створюють СВВ самі пропонують певний набір сигнатур, які встановлені за замовчуванням. Більшість СВВ дозволяють користувачам самим писати сигнатури, тестувати їх і ділитися ними з іншими користувачами. Деякі комерційні компанії також розробляють набори сигнатур, які можна отримати через щомісячну або щорічну підписку.

Тобто, основні типи сигнатур створенні для певної СВВ, це:

1. Сигнатури створені розробником СВВ
2. Створені комерційними компаніями, що спеціалізовані на розробці сигнатур
3. Сигнатури створені користувачами СВВ

Значну вигоду отримує СВВ яка може використовувати сигнатури розроблені для іншої СВВ, оскільки фактично це означає, що вона збільшує кількість доступних для роботи сигнатур в рази.

3.2.3 Підтримка роботи в багато поточному режимі

Системи що надають можливість використовувати багатоядерні та багатопроцесорні системи високої продуктивності мають значні переваги.

Кількість інтернет трафіку зростає дуже швидкими темпами. З покращенням технологій передачі, обробки і зберігання даних, ця швидкість буде зростати все сильніше. А в поєднанні зі збільшенням кількості атак, на відповідність яким треба буде перевірити трафік, це значно збільшує

важливість використання доступних апаратних ресурсів системами виявлення вторгнень. СВВ повинні брати до уваги ці тенденції, і реагувати відповідно до них вони можуть двома наступними способами:

1. Збільшення ефективності. В свою чергу цей спосіб можна розбити на два підпункти. Перший – це збільшення ефективності роботи основних елементів СВВ: систему збору подій, систему аналізу, систему логування і систему реагування на виявлені атаки. Другий – СВВ на основі сигнатур перевіряють інтернет трафік на відповідність сигнатурам атак. І ця перевірка займає різну кількість часу в залежності від складності і ефективності цього правила. Тобто можна оптимізувати самі сигнатури.
2. Полягає в використанні максимальної кількості доступних обчислювальних ресурсів. Сучасні процесори можуть збільшувати свою потужність за допомогою широкого набору способів. Але основними залишаються два способи, це збільшення потужності ядра і збільшення кількості ядер. На даний момент процес збільшення потужності окремих ядер відходить на другий план і сучасні процесори нарощують потужність завдяки збільшенню кількості ядер.

Відповідно до перерахованих способів системи виявлення вторгнень можуть збільшити швидкодію завдяки збільшенню ефективності. Але збільшення ефективності має межі, і якщо на початкових етапах можна збільшити ефективність доволі сильно, то з плином часу оптимізація буде приносити все менше вигоди.

Тому використання всіх, або майже всіх, наявних обчислювальних потужностей є основним способом збільшення ефективності в рази. Особливо

збільшити ефективність можна завдяки використанню всіх доступних ядер, кількість яких сильно зросла за останні 10 років.

Також необхідно враховувати що деякі системи не тільки використовують багатоядерні процесори, а пішли ще на крок далі і дозволяють використовувати обчислювальні потужності відеокарт. Кількість ядер сучасних відеокарт обчислюються сотнями, що разом з великою пропускною здатністю надає великі обчислювальні потужності.

Основні переваги багато потокового дизайну полягають у тому, що він пропонує підвищену швидкість і ефективність аналізу мережевого трафіку, а також може допомогти розділити навантаження на IDS / IPS, залежно від навантаження на різні підсистеми.

3.2.4 Кількість рівнів моделі OSI доступних для перевірки

Насамперед вхідний трафік розбивається на TCP, UDP або інші транспортні потоки, після чого парсери маркують їх і розбивають на високо рівневі протоколи і їх поля – нормалізуючи їх, якщо потрібно. Отримані декодовані і нормалізовані поля протоколів потім перевіряються наборами сигнатур, які виявляють, чи є серед мережевого трафіку спроби мережевих атак або пакети, властиві шкідливій активності.

Можливість розкодувати і проаналізувати пакети декількох рівнів моделі OSI, вимагає використання додаткових системних ресурсів, як процесора так і оперативної пам’яті. Тому система яка може використовувати багатоядерні процесори і/або відеокарти, матиме значний виграш в потужності.

Аналіз багатьох рівнів моделі OSI, особливо з підтримкою роботи в багато поточному режимі, дає системі виявлення вторгнень більші можливості для виявлення атак. А значить, робить цю систему більш ефективною і надійною.

3.2.5 Наявність документації

Документація є дуже зручним інструментом як для людини, яка тільки починає використовувати певне програмне рішення, так і для людини, яку можна вважати експертом з використання цієї програми.

Саме до документації будуть звертатися як до першоджерела при виникненні будь-яких питань щодо використання програмного продукту.

Тому наявність розгорнутої, добре структурованої документації є необхідною і свідчить про високий рівень продукту.

3.2.6 Можливість зв’язатися з розробниками продукту напряму

Контакти з розробниками є важливими з точки зору можливості отримання відповіді на важливі питання, відповідь на які не було знайдено в документації.

Також, наявність контактів дозволяє зв'язатися з розробниками для того щоб повідомити про виявлену атаку або вразливість. Якщо залишити інформацію про нову вразливість в мережі Інтернет у відкритому доступі, існує велика ймовірність що ця інформація буде використана зловмисниками.

Можливість зв’язатися з розробниками продукту напряму свідчить про 'серйозність' ПО, якщо в команді розробників виділяються людський ресурс для спілкування з клієнтами.

Отже, Можливість зв’язатися з розробниками продукту напряму я важливим критерієм оцінки програмного рішення. Це дозволяє задати питання або повідомити про знайдену вразливість.

3.3 Результати оцінювання Систем виявлення вторгнень

Для оцінювання Систем виявлення вторгнень згідно вищенаведеного методу були використані офіційні інформаційні ресурси, а також фреймворк pytbull, який дозволяє виконати ряд атак на СВВ та проаналізувати кількість виявлених атак.

Таблиця 3.4 – Результати оцінювання Систем виявлення вторгнень

Критерій оцінки	Snort	Suricata
Атаки на основі фрагментованих пакетів	1	1
Атаки на основі використання поліморфного Shellcode	3	3
Інші основні методи обходу систем виявлення вторгнень	5	7
Підсумок	9	11
Наявність механізмів виявлення потенційних атак	12	15

Кінець Таблиці 3.4

Критерій оцінки	Snort	Suricata
Можливість використання сигнатур спроектованих для використання в інших СВВ	0	5
Підтримка роботи в багато поточному режимі	3	8
Кількість рівнів моделі OSI доступних для перевірки	6	7
Наявність документації	5	5
Можливість зв'язатися з розробниками продукту напряму	5	4

Підсумкові оцінки що виставляються для систем виявлення вторгнень на основі створеної методики. Максимальна оцінка при успішному виконанні всіх критерії – 50 балів.

Snort = 12+0+3+6+5+5 = 31.

Suricata = 15+5+8+7+5+4 = 44.

Далі буде наведена більш детальна інформація щодо оцінювання кожної системи виявлення вторгнень.

Snort

Дана система виявлення вторгнень була розроблена в 1998. Вона є настільки вдалою, що досі є однією з провідних СВВ і довгий час була золотим стандартом СВВ.

Має захист від всіх основних методів обходу СВВ, що позитивно впливає на її оцінку.

Не може використовувати сигнатури спроектовані для використання в інших системах виявлення вторгнень, оскільки даний функціонал не був закладений в неї під час проектування і з плином часу в ньому не виникало необхідності.

Оскільки перша версія snort була розроблена в 1998 році, вона не мала підтримки багато поточної роботи. Це через певний проміжок часу стало значним недоліком, оскільки розмір трафіку значно зріс і потужності одного ядра може не вистачити для аналізу всього трафіку. Згодом придумали спосіб використання Snort в ‘псевдо - багато поточному’ режимі, коли запускається декілька snort в різних потоках, але це доволі не зручно і вимагає додаткової обчислювальної потужності для узгодження роботи декількох Snort через між процесну взаємодію. Версія snort 3.0 beta повинна підтримувати багатопоточний режим роботи, але це лише версія для тестування роботи і пошуку можливих багів, також ще не пройшло достатньо часу, щоб зрозуміти наскільки це рішення є ефективним. Через причини описані вище snort отримав низьку оцінку роботи в багато поточному режимі.

З точки зору кількості рівнів моделі OSI доступних для аналізу, snort не підтримує application рівень, на що розробникам вказували з ранніх днів існування. Тому Snort отримав оцінку 6 з 7.

Має значну базу користувачів в усьому світі, про неї було написано багато матеріалу та книг, розгорнута документація з питань безпеки написана добре. Це позитивно вплинуло на оцінку даного критерію.

Має окремі контакти для питань з безпеки, а також значну кількість форумів з великою кількістю користувачів, на яких швидко можна знайти відповіді на багато питань.

Підсумкова оцінка = 31, що свідчить про добрий рівень захищеності та надійності серед оцінюваних систем виявлення вторгнень.

Suricata

Перша версія даної система виявлення вторгнень була випущена в 2010 році, тобто в неї увійшли більшість нових розробок і були взяті до уваги помилки попередніх систем.

Має захист від більшості способів обходу СВВ, через що отримала високий бал по даному критерію.

Suricata була спроектована як система що може використовувати сигнатури Snort, що дало її значну перевагу на ринку, оскільки вона з першої версії могла використовувати сигнатури Snort, які розроблялись вже 12 років. По даному критерію система отримала максимальну оцінку.

Оскільки система була розроблена в 2010 році вона підтримує роботу в багато поточному режимі що дозволяє використовувати значні потужності сучасних багатоядерних процесорів та навіть відеокарт. Через це Suricata отримала максимальний бал по даному критерію.

Система підтримує аналіз рівня application по моделі OSI, тому вона отримала 7 балів з 7 по заданому критерію.

Suricata має розгорнуту документацію з питань безпеки яка дуже добре оформлена, тому вона отримала максимальний бал з даного критерію.

Система має контакти для питань з безпеки, але має меншу кількість користувачів та форумів порівняно зі Snort. Тому система отримала 4 з 5 балів по даному критерію.

Висновки до розділу 3

В даному розділі був запропонований метод оцінки захищеності систем виявлення вторгнень, що базується на системі оцінки згідно розроблених критеріїв.

Виділені критерії дозволяють комплексно оцінити СВВ. Для підтвердження ефективності методу були проаналізовані такі фреймворки: Snort та Suricata. За результатами оцінювання обраних СВВ згідно розробленого методу, можна зробити висновок, що з точки зору безпеки Suricata виявився більш захищеним. На це слід звертати увагу адміністраторам, що знають про можливі обмеження проектних рішень та обирають СВВ для побудови захищеної автоматизованої ситуації 3 рівня.

ВИСНОВКИ

В даній роботі у першому розділі був виконаний огляд теоретичної інформації стосовно СВВ; наведені необхідні визначення та принцип роботи СВВ; також було надано класифікацію СВВ.

В другому розділі були наведені способи обходу СВВ.

В третьому розділі, спираючись на інформацію в перших двох розділах, був побудований метод оцінки СВВ. Даний метод був апробований на готових програмних рішеннях.

Тобто, результатом даної роботи є метод оцінки захищеності систем виявлення вторгнень. За допомогою даного методу був оцінений стан захищеності двох найбільш популярних систем виявлення вторгнень Snort та Suricata. Представлені критерії, що дозволяють оцінити захищеність конкретного СВВ для порівняння з результатами оцінки інших СВВ.

Для апробації методу були обрані дві СВВ. Кожна система отримала підсумкову оцінку захищеності, що є сумою оцінок повноти реалізації таких критеріїв:

1. Наявність механізмів виявлення потенційних атак
2. Можливість використання сигнатур спроектованих для використання в інших СВВ
3. Підтримка роботи в багато поточному режимі
4. Кількість рівнів моделі OSI доступних для перевірки
5. Наявність документації
6. Можливість зв'язатися з розробниками продукту напряму

Підрахунок оцінки повноти реалізації здійснювався за допомогою математичних методів, що були розроблені для кожного критерію.

В результаті оцінки, обрані фреймворки отримали наступні числові показники оцінки, при максимальній оцінці 50:

1. Snort — 31;
2. Suricata — 44;

За даною методикою “Suricata” має більшу оцінку і є рекомендованою СВВ.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ

1. Computer Security Threat Monitoring and Surveillance [Електронний ресурс]. – 1980. – Режим доступу до ресурсу: https://csrc.nist.gov/csrc/media/publications/conference-paper/1998/10/08/proceedings-of-the-21st-nissc-1998/documents/early-cs-papers/ande80.pdf.

2. An Intrusion Detection Model [Електронний ресурс]. – 1986. – Режим доступу до ресурсу: https://www.academia.edu/8674514/An_Intrusion-Detection_Model.

3. IDES: An Intelligent System for Detecting Intruders [Електронний ресурс]. – 1990. – Режим доступу до ресурсу: https://www.researchgate.net/profile/Teresa_Lunt/publication/242383334_Ides_an_intelligent_system_for_detecting_intruders/links/552dacae0cf29b22c9c4f95f/Ides-an-intelligent-system-for-detecting-intruders.pdf.

4. Detecting Intruders in Computer Systems [Електронний ресурс]. – 1993. – Режим доступу до ресурсу: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.43.7289.

5. Expert Systems in Intrusion Detection: A Case Study [Електронний ресурс]. – 1988. – Режим доступу до ресурсу: https://www.researchgate.net/publication/243557654_Expert_systems_in_intrusion_detection_A_case_study.

6. Haystack: An Intrusion Detection System [Електронний ресурс]. – 1988. – Режим доступу до ресурсу: http://people.scs.carleton.ca/~soma/id/readings/smaha-haystack.pdf.

7. Detection of Anomalous Computer Session Activity [Електронний ресурс]. – 1989. – Режим доступу до ресурсу: http://people.scs.carleton.ca/~soma/id-2007w/readings/vaccaro-wisdom+sense.pdf.

8. Adaptive Real-time Anomaly Detection Using Inductively Generated Sequential Patterns [Електронний ресурс]. – 1990. – Режим доступу до ресурсу: https://ieeexplore.ieee.org/abstract/document/63857.

9. A Network Security Monitor [Електронний ресурс]. – 1990. – Режим доступу до ресурсу: http://seclab.cs.ucdavis.edu/papers/pdfs/th-gd-90.pdf.

10. DIDS (Distributed Intrusion Detection System) — Motivation, Architecture, and An Early Prototype [Електронний ресурс]. – 1991. – Режим доступу до ресурсу: http://seclab.cs.ucdavis.edu/papers/DIDS.ncsc91.pdf.

11. A Phased Approach to Network Intrusion Detection [Електронний ресурс]. – 1991. – Режим доступу до ресурсу: https://permalink.lanl.gov/object/tr?what=info:lanl-repo/lareport/LA-UR-91-0334.

12. Bro: A System for Detecting Network Intruders in Real-Time [Електронний ресурс]. – 1998. – Режим доступу до ресурсу: https://www.usenix.org/legacy/publications/library/proceedings/sec98/full_papers/paxson/paxson.pdf.

13. Intrusion Detection: An Introduction to Internet Surveillance, Correlation, Trace Back, Traps, and Response [Електронний ресурс]. – 1999. – Режим доступу до ресурсу: https://www.amazon.com/Intrusion-Detection-Introduction-Surveillance-Correlation/dp/0966670078.

14. Snort IDS and IPS Toolkit [Електронний ресурс]. – 2007. – Режим доступу до ресурсу: https://www.amazon.com/Snort-Toolkit-Beales-Source-Security/dp/1597490997.

15. ADAM: Detecting Intrusions by Data Mining [Електронний ресурс]. – 2001. – Режим доступу до ресурсу: https://pdfs.semanticscholar.org/d69a/e114a54a0295fe0a882d205611a121f981e1.pdf.

16. Polymorphic Blending Attacks [Електронний ресурс]. – 2006. – Режим доступу до ресурсу: https://www.usenix.org/legacy/event/sec06/tech/full_papers/fogla/fogla.pdf.

17. Multi-Layer Integrated Anomaly Intrusion Detection System for Mobile Adhoc Networks [Електронний ресурс]. – 2007. – Режим доступу до ресурсу: http://xanadu.cs.sjsu.edu/~drtylin/classes/cs157A/Project/PDF-files/CS157B_Team14/4_Eric_Nam%20-%20IEEE%20Website/4156645.pdf.

18. Spectrogram: A Mixture-of-Markov-Chains Model for Anomaly Detection in Web Traffic [Електронний ресурс]. – 2009. – Режим доступу до ресурсу: https://academiccommons.columbia.edu/doi/10.7916/D8891G6G.

19. A fuzzy Intrusion Detection System based on categorization of attacks [Електронний ресурс]. – 2014. – Режим доступу до ресурсу: https://www.semanticscholar.org/paper/A-fuzzy-Intrusion-Detection-System-based-on-of-Varshovi-Rostamipour/8dc771ce3584a6daafeb2023b752b2e99e03f5d8.

20. Tuning Intrusion Detection to Work with a Two Encryption Key Version of IPsec [Електронний ресурс]. – 2007. – Режим доступу до ресурсу: https://www.semanticscholar.org/paper/Tuning-Intrusion-Detection-to-Work-with-a-Two-Key-Studer-McLain/18b0565076913ef05589bdae01f0e44a80c39db0.

21. Improving the Detection of Encrypted Data on Storage Devices [Електронний ресурс]. – 2015. – Режим доступу до ресурсу: https://www.semanticscholar.org/paper/Improving-the-Detection-of-Encrypted-Data-on-Thurner-Grun/66e278a6ddc67a4d1a72b5729d6396a1f5ac40b1.