ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ
НАВЧАЛЬНО–НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ
КАФЕДРА КОМП'ЮТЕРНОЇ ІНЖЕНЕРІЇ

КВАЛІФІКАЦІЙНА РОБОТА

на тему: **«ДОСЛІДЖЕННЯ ЗАСОБІВ ОРГАНІЗАЦІЇ БЕЗПЕКИ КОМП'ЮТЕРНИХ МЕРЕЖ З МЕТОЮ ОПТИМІЗАЦІЇ ЗАХИСТУ ВІД ВТОРГНЕНЬ»**

на здобуття освітнього ступеня магістра
зі спеціальності 123 Комп'ютерна інженерія
(код, найменування спеціальності)
освітньо-професійної програми Комп'ютерні системи та мережі
(назва)

Кваліфікаційна робота містить результати власних досліджень. Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело

_____ _____
(підпис) *(ім'я, ПРІЗВИЩЕ здобувача)*

Виконав: здобувач вищої освіти гр. КСДМ-61
Кувік Н.І.
(Ім'я, ПРІЗВИЩЕ)
Керівник: доктор філософії, доцент Лемешко А.В.
науковий ступінь, вчене звання (Ім'я, ПРІЗВИЩЕ)
Рецензент: _____
науковий ступінь, вчене звання (Ім'я, ПРІЗВИЩЕ)

Київ 2023

ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ

Навчально-науковий інститут інформаційних технологій

Кафедра Комп'ютерної інженерії
Ступінь вищої освіти «Магістр»
Спеціальність 123 Комп'ютерна інженерія
Освітньо-професійна програма Комп'ютерні системи та мережі

ЗАТВЕРДЖУЮ
Завідувач кафедрою Комп'ютерної інженерії
Н.О. Лащевська Ім'я, ПРІЗВИЩЕ
"____" _____ 2023 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

Кувік Назару Івановичу
(прізвище, ім'я, по батькові здобувача)

1.Тема кваліфікаційної роботи Дослідження засобів організації безпеки комп'ютерних мереж з метою оптимізації захисту від вторгнень
керівник кваліфікаційної роботи доктор філософії, доцент кафедри КІ Лемешко А.В.
(ім'я, ПРІЗВИЩЕ, науковий ступінь, вчене звання)
затверджені наказом Державного університету інформаційно-комунікаційних технологій від «19» жовтня 2023 р. № 145
2. Строк подання кваліфікаційної роботи: «____» _____ 2023 р.
3. Вихідні дані до кваліфікаційної роботи:
3.1 Технічна документація стосовно розробки методів запобігання та виявлення вторгнень;
3.2 Інтернет-ресурси стосовно засобів організації безпеки комп'ютерних мереж;
3.3 Науково-технічна література.

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити:
4.1 Загальна характеристика методів аналізу безпеки комп'ютерних мереж;
4.2 Методи виявлення вторгнень в комп'ютерні мережі;
4.3 Методи візуалізації результатів аналізу виявлення загроз;
4.4 Проектуваення системи виявлення вторгнень IDS.

5. Перелік графічного матеріалу: *презентація*

6. Дата видачі завдання «___» _____ 2023 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів кваліфікаційної роботи	Строк виконання етапів роботи	Приміпка
1	Підбір науково-технічної літератури	19.10.2023-23.10.2023	Виконано
2	Загальна характеристика методів аналізу безпеки комп'ютерних мереж	23.10.2023-27.10.2023	Виконано
3	Методи виявлення вторгнень в комп'ютерні мережі	28.10.2023-09.11.2023	Виконано
4	Методи візуалізації результатів аналізу виявлення загроз	10.11.2023-20.11.2023	Виконано
5	Проектуваення системи виявлення вторгнень IDS	21.11.2023-27.11.2023	Виконано
6	Вступ, висновки, реферат	28.11.2023-02.12.2023	Виконано
7	Розробка обов'язкових демонстраційних матеріалів	03.12.2023-06.12.2023	Виконано
8	Попередній захист роботи	07.12.2023	Виконано

Здобувач вищої освіти _____ Кувік Н.І.
(підпис) *(Ім'я, ПРІЗВИЩЕ)*

Керівник
кваліфікаційної роботи _____ Лемешко А.В.
(підпис) *(Ім'я, ПРІЗВИЩЕ)*

РЕФЕРАТ

Текстова частина кваліфікаційної роботи на здобуття освітнього ступеня магістра: 81 стор., 12 табл., 30 рис., 24 джерел.

Мета роботи - Дослідження та розробка оптимальних стратегій організації безпеки комп'ютерних мереж з акцентом на оптимізацію захисту від потенційних вторгнень.

Об'єкт дослідження – засоби організації безпеки комп'ютерних мереж.

Предмет дослідження – комп'ютерні мережі.

Короткий зміст роботи:

В дипломній роботі наведено аналіз методів запобігання вторгненням в комп'ютерній мережі, методів виявлення вторгнень в мережі. Також було зосереджено увагу на дослідженні методів візуалізації результатів аналізу виявлення загроз. Було розроблено системи виявлення вторгнень на мережеві ресурси з застосуванням нейромережевих технологій.

КОМП'ЮТЕРНІ МЕРЕЖІ, МЕТОДИ ЗАПОБІГАННЯ ВТОРГЕННЯМ, МЕТОДИ ВИЯВЛЕННЯ ВТОРГНЕНЬ, МЕТОДИ ВІЗУАЛІЗАЦІЇ, БЕЗПЕКА КОМП'ЮТЕРНИХ МЕРЕЖ, IDS

ABSTRACT

Text part of the master's qualification work:
81 pages, 30 pictures, 12 table, 24 sources.

The purpose of the work is research and development of optimal strategies for organizing the security of computer networks with an emphasis on optimizing protection against potential intrusions.

Object of research is means of organizing the security of computer networks.

Subject of research is computer networks.

Summary of the work:

The thesis provides an analysis of methods of preventing computer network intrusions, methods of detecting network intrusions. Attention was also focused on researching methods for visualizing the results of threat detection analysis. Systems for detecting intrusions on network resources using neural network technologies were developed.

KEYWORDS:

COMPUTER NETWORKS, INTRUSION PREVENTION METHODS, INTRUSION DETECTION METHODS, VISUALIZATION METHODS, COMPUTER NETWORK SECURITY, IDS

Зміст

ВСТУП

Зі швидким розвитком комп'ютерних технологій інформаційна мережа стає важливою гарантією соціального розвитку. Комп'ютерна мережа також будується та покращується з кожним днем. Цей розвиток приносить велику зручність суспільству та реалізує спільне використання ресурсів.

Певною мірою це значно підвищує ефективність роботи людей та, до певної міри, це також підвищує рівень модернізації великих підприємств. відкритість,

спільне використання та інтернаціональність Інтернету, створюючи великі проблеми для безпеки комп'ютерів та мережевих додатків Тому в процесі функціонування мережних додатків слід звертати увагу на мережеві загрози з різних аспектів, включаючи фізичні атаки на лінії передачі, комунікаційні протокольні атаки та атаки в режимі реального часу на апаратні та програмні вразливості.

Мережева безпека означає, що обладнання, програмне забезпечення та дані в мережній системі повинні бути захищені та приховані в кожній ланці процесу використання комп'ютера. Вони можуть бути захоплені трояном або іншими шкідливими вірусами під час завантаження, запуску та надсилання електронної пошти. Для користувачів викрадені дані можуть призвести до колапсу чи паралічу інформаційної системи та до інших втрат. Інформація може знищуватись змінюватись і просочуватись безпечно чи через зловмисні причини. Система працює безперервно та надійно, а мережеве обслуговування не переривається. В даний час популярні інтернет-знання все більше і більше створюють величезні перешкоди захисту інформації. Тому питання інформаційної безпеки під час безперервного розвитку комп'ютерної мережі є дуже важливим.

Фізична безпека відноситься до фізичного захисту різних комп'ютерних пристроїв та пов'язаних з ними пристроїв у мережній системі, уникаючи пошкодження та втрати тощо. Логічна безпека включає цілісність інформації, конфіденційність та доступність.

Завдяки постійному розвитку інформаційних технологій хакерські технології поширюються повсюдно, вони можуть за бажанням легко використовувати комп'ютерну мережу для отримання, придбання та розкриття інформації і за це складно знайти конкретний відповідальний персонал.

Для конкретної роботи операційної системи комп'ютерної мережі персонал та техніка повинні чітко розрізняти її різні типи, щоб сприяти сталому розвитку комп'ютерної мережі. Для забезпечення нормальної роботи комп'ютерної мережі операційну систему комп'ютерної мережі можна назвати серцем комп'ютера.

Операційна система комп'ютерної мережі може реалізувати нормальну роботу комп'ютера і може керувати мережевими ресурсами з наукової точки зору та ділитися інформацією. Є доречним технічні характеристики специфікацій, в тому числі і для забезпечення її фізичного середовища.

Безпечне мережеве середовище має збільшити виявлення інформацію про порт доступу до мережі для своєчасного виявлення та обробки недостовірної інформації. Так званий список контролю доступу відноситься до налаштування джерела доступу, яке може ефективно запобігти поганим відвідувачам, таким чином, щоб внутрішні дані підприємства могли бути захищені певною мірою.

Технічною основою вдосконалення структури даних є технологія злиття інформації, основана на знаннях та надмірних методах міркувань, і більшість роботи полягає в класифікації даних за допомогою зіставлення зі зразком, інтелектуального аналізу даних, вибору функцій та машинного навчання.

Процес забезпечення безпеки комп’ютерної інформації потребує спільного використання всіх видів стратегій і їх розумного розгортання.. Тільки таким чином можна мінімізувати ймовірність порушення прав на безпеку інформацію .та отримати гарантію безпеки. Саме цій актуальній темі присвячена магістерська робота.

РОЗДІЛ 1 ЗАГАЛЬНА ХАРАКТЕРИСТИКА МЕТОДІВ АНАЛІЗУ БЕЗПЕКИ КОМП’ЮТЕРНИХ МЕРЕЖ

Оцінювання методів аналізу безпеки сучасних комп’ютерних мереж здійснюється за такими критеріями:

- покриття;
- ефективність;
- продуктивність;
- застосовність до різних типів збору даних;
- можливість виявлення вторгнень у зашифрований трафік.

Перший критерій полягає у здатності методу виявляти загрози безпеці комп’ютерних мереж. Покриття є повним, якщо метод виявляє як відомі, і невідомі загрози.

Другий критерій означає точність виявлення, кількість хибно-позитивних результатів, отриманих за допомогою методу.

Третій критерій означає швидкість обробки трафіку у мережі за цим методом і має вирішальне значення для розгортання високошвидкісних мереж.

Четвертий критерій визначає, чи буде захоплення пакетів та/або (вибірка) потокових даних, що оцінюється. Останній критерій стає дедалі важливішим у сучасній мережі.

1.1 Системи виявлення вторгнень

Вторгнення в мережу – це будь-яка несанкціонована дія в комп’ютерній мережі. Мережеві вторгнення часто пов'язані з крадіжкою цінних мережевих ресурсів і зазвичай ставлять під загрозу безпеку мереж та їх даних.

Вторгнення ставить під загрозу комп'ютерну систему, порушуючи безпеку такої системи або перетворюючи її на небезпечний стан.

Мережі та кінцеві точки вразливі для вторгнень з ненавмисних джерел, які називають суб'єктами загроз. Зловмисник може бути буквально в будь-якій точці світу. Все, що їм потрібно, це доступ до Інтернету, мотив і метод або маршрут атаки, який зазвичай називають вектором загрози. Рис.1.1 демонструє точки комп'ютерної мережі, вразливі для вторгнень за векторами загроз.

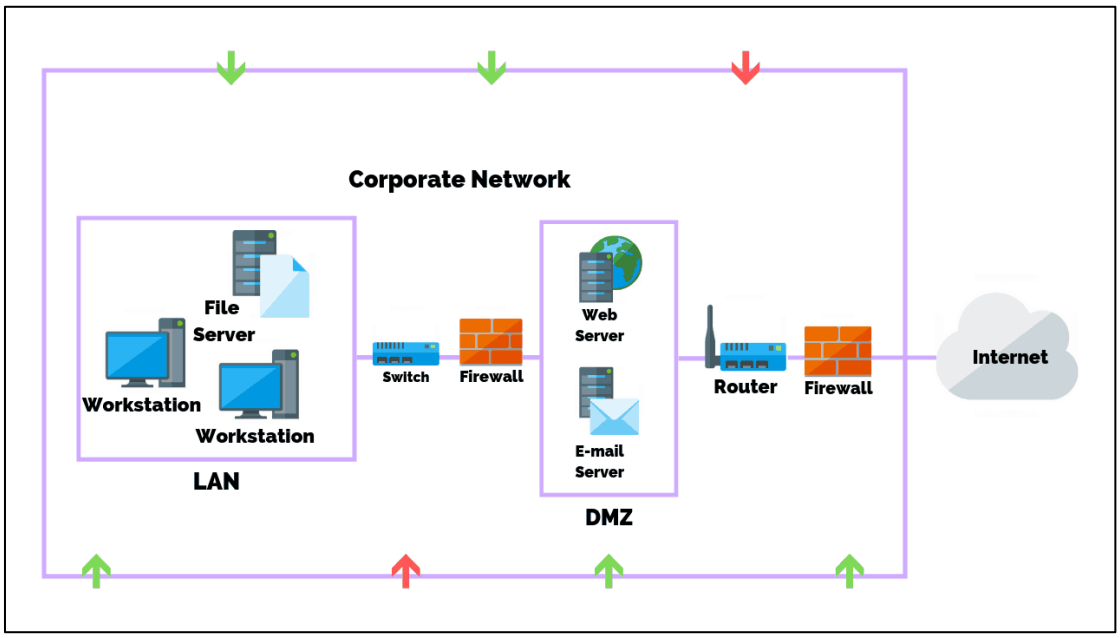


Рисунок 1.1 - Точки комп'ютерної мережі, вразливі для вторгнень за векторами загроз

Система виявлення вторгнень або IDS є технологією безпеки комп'ютерної мережі і призначена для виявлення шаблонів мережевого трафіку. IDS не розміщується безпосередньо на одному потоці з мережним трафіком.

Зазвичай вона вставляється у пристрій дзеркального відображення портів tap або span для моніторингу та сповіщення адміністратора про небезпеку, не впливаючи на потік даних.

Можна розділити системи виявлення вторгнень (IDS) на два основні класи в залежності від їхнього положення у мережі:

- хостові системи виявлення вторгнень (HIDS);

- мережеві системи виявлення вторгнень (NIDS); [1].

На рис. 1.2 представлено класи системи виявлення вторгнень (IDS).

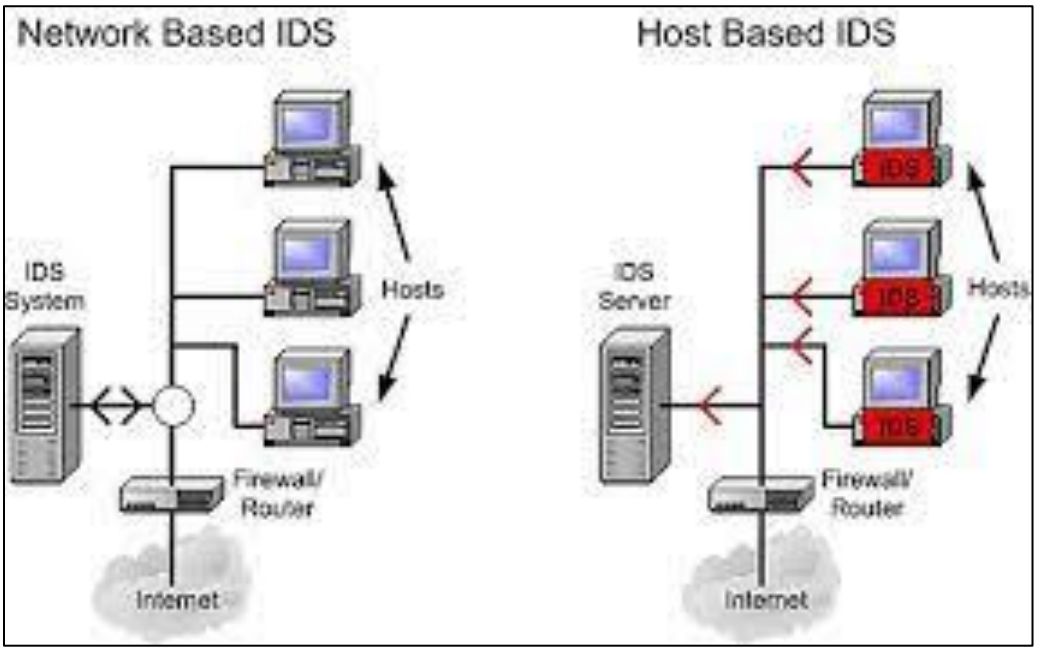


Рисунок 1.2 - Класи системи виявлення вторгнень (IDS)

Виявлення вторгнень на хості виконується на хост-комп'ютері у комп'ютерній мережі. Хост-система виявлення вторгнень зазвичай відстежує файли (наприклад, журнали брандмауера, журнали веб-сервера та системні журнали) та цілісність системних файлів (наприклад, цілісність ядра чи відкриті порти).

При мережевому виявленні вторгнень перевіряється весь вхідний або вихідний мережевий трафік на підозрілі шаблони. Шаблони можуть бути представлені у вигляді підпису, рядка символів, який описує певну атаку. Інший підхід, який застосовується при мережевому виявленні вторгнень – це визначення вторгнень на основі аномалій. При цьому спочатку створюється модель нормальної поведінки мережі, потім оцінюється різниця між поведінкою мережі і нормальною поведінкою моделі. Якщо ця різниця є більшою від визначеного значення (порога), це може вказувати на атаку.

Інші мережеві системи виявлення вторгнень (NIDS) використовують аналіз протоколу з відстеженням стану виявлення підозрілих, несподіваних чи неприпустимих послідовностей пакетів в залежності від конкретного протоколу.

Мережеві системи виявлення вторгнень є пасивними системами: вони "невидимі" для інших хостів і в основному для зловмисників.

Щодо IDS часто згадуються два наступні терміни: хибнопозитивний і хибнонегативний. Перше означає хибне сповіщення IDS: система класифікує безпечний трафік як зловмисний. Навпаки, останній вказує на шкідливий трафік, який не було розпізнано з ІДС. Звичайно, існує тенденція до мінімізації числа як хибних спрацьовувань, так і негативи. Наприклад, якщо IDS видає високий відсоток помилкових спрацьовувань, це турбує адміністратора про подальший ручний аналіз цих попереджень.

1.2 Система запобігання вторгненням

У порівнянні з системою виявлення вторгнень (IDS) система запобігання вторгненням (IPS) є реактивною системою, в якій IDS тісно пов'язана із брандмауером (і має бути частиною каналу зв'язку). На рис. 1.3 представлено відмінності між функціонуванням системи виявлення вторгнень (IDS) і системи запобігання вторгненням (IPS).

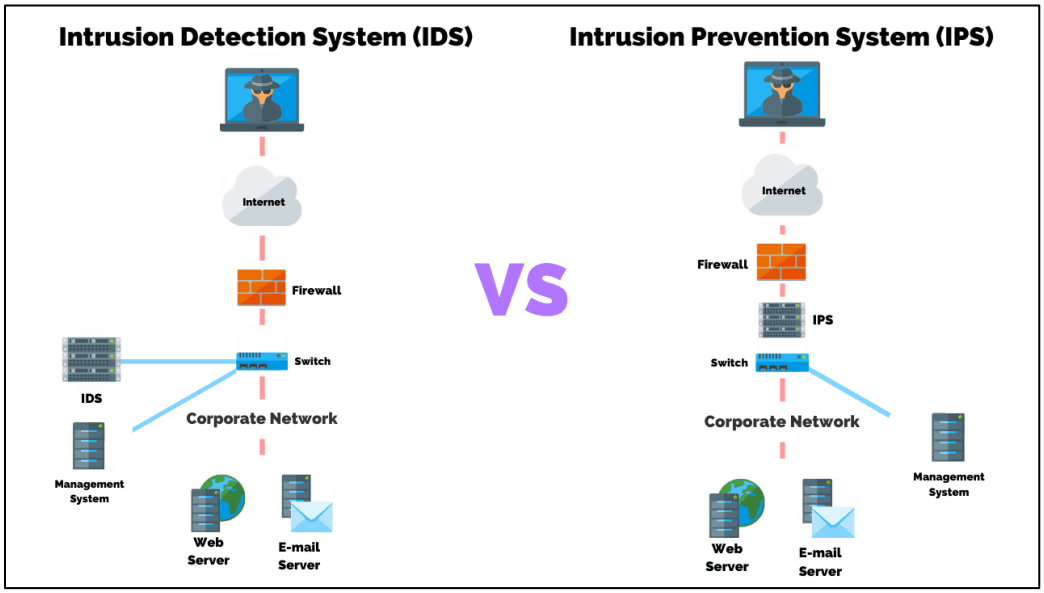


Рисунок 1.3 - Відмінності між функціонуванням системи виявлення вторгнень (IDS) і системи запобігання вторгненням (IPS)

Основна відмінність між системою виявлення вторгнень (IDS) і системою запобігання вторгненням (IPS) полягає в тому, що IDS використовується для

моніторингу мережі, яка потім надсилає попередження при виявленні підозрілих подій у системі або мережі. IPS реагує на поточні атаки з метою запобігти їх доступу до цільових систем та мереж. Хоча IDS і IPS здатні виявляти атаки, основна відмінність полягає у їх реакції на атаку.

Однак важливо зазначити, що і IDS, і IPS можуть реалізовувати одні й ті самі методи моніторингу та виявлення. Отже, головним завданням системи запобігання вторгненням IPS є пом'якшення (зупинка) виявленої атаки. На рис. 1.4 представлено схему функціонування системи запобігання вторгненням (IPS).

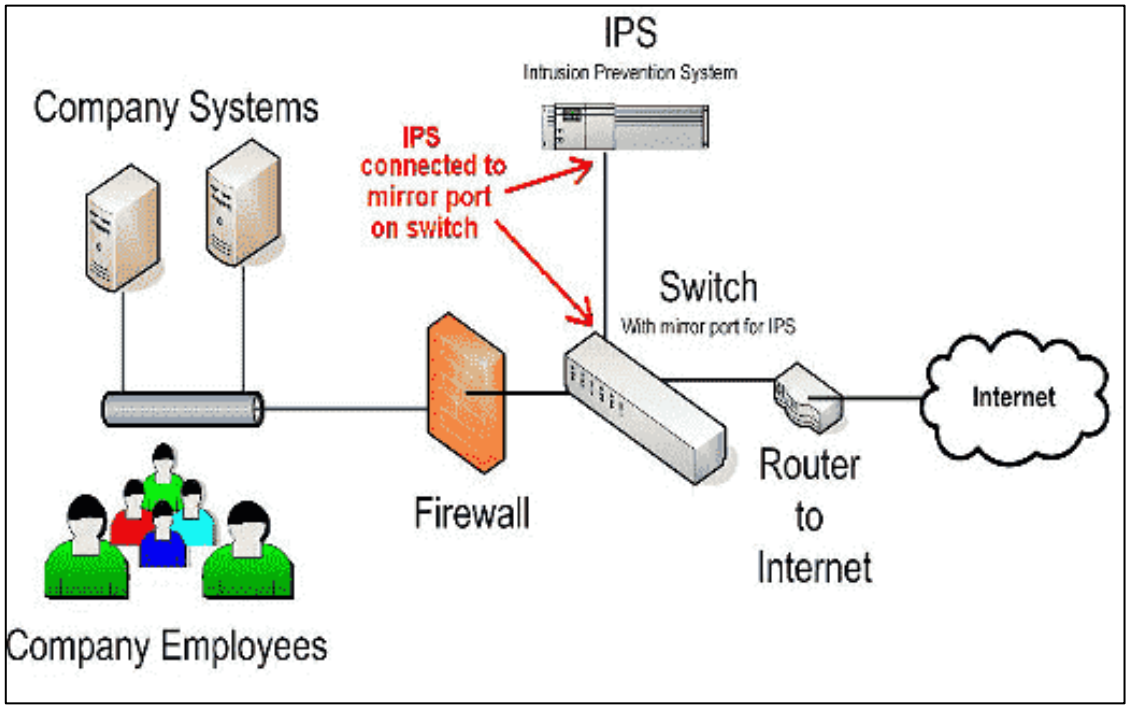


Рисунок 1.4 - Схема функціонування системи запобігання вторгненням (IPS)

IPS можна розділити на чотири класи:

- мережеві системи запобігання вторгненням;
- системи запобігання вторгненням у безпроводову мережу (WIPS);
- системи запобігання вторгненням на основі мережевої поведінки;
- хостові системи запобігання вторгненням.

1.2.1 Мережеві системи запобігання вторгненням (NIPS)

Мережеві системи запобігання вторгнень або NIPS виявляють і запобігають шкідливим діям, аналізуючи пакети протоколів по всій мережі. Їх часто називають системами IDS/IPS або системами виявлення та запобігання вторгненням.

Після встановлення NIPS збирає інформацію з консолі хоста та з мережі, щоб ідентифікувати дозволені хости, програми та операційні системи, які зазвичай використовуються в мережі.

Вони також реєструють інформацію про характеристики звичайного мережевого трафіку для виявлення будь-яких підозрілих змін у мережі.

NIPS може запобігати атакам різними способами, наприклад, відправляючи TCP-з'єднання для запобігання атакам, обмежуючи використання пропускної здатності або навіть блокуючи підозрілу мережну активність.

Сьогоднішні NIPS здатні навіть давати команду брандмауерам та маршрутизаторам блокувати підозрілу активність.

Недоліком NIPS є те, що ця система зазвичай не аналізує зашифрований мережевий трафік, не справляється з великими обсягами трафіку і не справляється з прямими атаками на IDS/IPS. На рис. 1.5 представлено схему застосування NIPS.

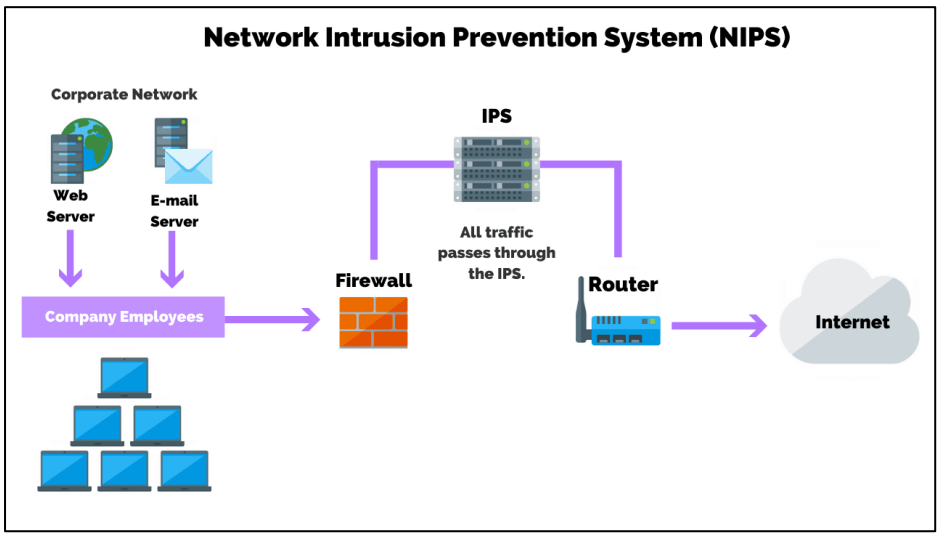


Рисунок 1.5 - Схема застосування NIPS

NIPS переважно використовує виявлення з урахуванням сигнатур виявлення загроз. Виявлення на основі сигнатур шукає шаблони або сигнатури розпізнаних раніше загроз для виявлення потенційних нових загроз.

1.2.2 Системи запобігання вторгненням у безпроводову мережу (WIPS)

Система запобігання вторгненням у безпроводову мережу (WIPS) працює на рівні 2 (рівень каналу передачі даних) моделі взаємодії відкритих систем.

WIPS може виявляти наявність шахрайських або неправильно налаштованих пристроїв і запобігати їх роботі в безпроводових корпоративних мережах, скануючи RF мережі щодо відмови в обслуговуванні та інших форм атак.

1.2.3 Системи запобігання вторгненням на основі мережевої поведінки

Цей метод в першу чергу включає виявлення на основі аномалій, яке шукає відхилення від того, що відомо як «нормальне» поведінка в системі або мережевої активності.

Після запуску виявлення аномалій потребує періоду навчання, протягом якого за певний період будується профіль того, що вважається нормальною поведінкою. Невідповідності з цим профілем позначаються як шкідливі.

Виявлення на основі аномалій відмінно підходить для виявлення нових загроз, але можуть виникнути проблеми, якщо мережа буде скомпрометована під час періоду навчання, оскільки зловмисна поведінка може реєструватися як завжди під час створення профілю.

Крім того, виявлення на основі аномалій також дає багато помилкових спрацьовувань через доброякісну активність, яка не була розпізнана протягом початкового періоду навчання.

Виявлення з аналізом протоколу з відстеження стану аналогічно виявленню на основі аномалій в тому сенсі, що воно шукає відхилення від нормальної поведінки мережі або системи.

1.2.4 Хост-системи запобігання вторгненням

Хост-системи запобігання вторгненням, або HIPS, аналізують активність усередині одного хоста, щоб виявляти та запобігати шкідливим діям.

HIPS насамперед аналізують поведінку коду, використовуючи як сигнатурний метод, так і метод виявлення з урахуванням аномалій виявлення підозрілої активності. Цим системам часто надають перевагу через запобігання атакам з використанням шифрування.

Подібно до HIDS, багато рішень для захисту від шкідливих програм надають HIPS як частину сімейства продуктів.

HIPS також може запобігти доступу до конфіденційної інформації, розташованої на хості, тим самим запобігаючи будь-якій потенційній шкоді, заподіяній руткітами або троянськими конями.

HIPS також може запобігти обробці хост-комп'ютером шкідливої активності в мережі.

Оскільки HIPS забезпечує безпеку тільки для одного хост-комп'ютера або сервера, його краще використовувати разом з IDS/IPS і WIPS, щоб забезпечити повне управління загрозами по всій мережі, надаючи комплексну програму запобігання.

Отже, режими роботи IPS та IDS різні, але близькі за своєю метою, яка полягає у захисті мережі від зловмисників та їх методів атаки.

Приблизно у 2005 році постачальники об'єднали дві технології, щоб скористатися їхньою функціональністю. Щоб IPS могла навіть заблокувати будь-який трафік, вона має спочатку виявити та перевірити трафік, що описує функціональність IDS.

Замість двох окремих пристроїв, що виконують дві окремі функції, виробникам брандмауерів та інших мережних продуктів було цілком розумно поєднати дві системи в один пристрій. Це відкрило можливість нового терміну, який зазвичай використовується сьогодні, «Наступне покоління», зокрема, системи брандмауера наступного покоління/NGFW і Unified Threat Management/UTM.

Хоча Gartner ввів термін «брандмауер наступного покоління» в 2003 році і передбачив, що вони включатимуть функції IPS і будуть пропонуватися в 2006 році, NGFW не набули широкого поширення до 2013 року.

На той час вони почали включати функції IDS/IPS, такі як використання сигнатур для виявлення відомих атак і пошук аномалій та відхилень протоколу в потоці пакетів.

Коли технології об'єднані в одному пристрої, адміністратор має варіанти розгортання як вбудовану систему запобігання вторгнень або виявлення тільки з датчиками, стратегічно розміщеними для пасивного моніторингу мережного трафіку.

Ця модель конфігурації надає адміністратору можливість використовувати пристрій у режимі запобігання або виявлення.

IDS/IPS нового покоління було розроблено у відповідь складні цільові загрози, які можуть обійти IDS/IPS першого покоління. Компанія також може захотіти впровадити рішення SIEM разом із IPS/IDS для прийому та агрегування даних журналу для виявлення подій, які можуть спонукати до потенційного використання системи.

При оцінці рішення для забезпечення безпеки корпоративної інфраструктури або будинку необхідно пам’ятати, що інтернет-загрози безпеки стають все більш тихими та небезпечними.

Багаторівнева система безпеки, що поєднує сигнатурні та поведінкові технології з іншими інструментами, дозволить зрівняти правила гри зі зловмисниками та їх методами атаки.

Об'єднана функціональність системи запобігання вторгненням, незалежно від того, мережева вона чи хост-система, є одним із інструментів, що заслуговують на увагу, і відіграє важливу роль у забезпеченні безпеки корпоративних мережевих інфраструктур та персональних комп'ютерів.

1.2.5 Збір трафіку з урахуванням потоку

Класичний підхід багатьох IDS чи IPS до збору даних полягає у захопленні всіх мережевих пакетів, які проходять через систему, найчастіше у форматі pcap1. Навпаки, багато маршрутизаторів зонди моніторингу виконують збір даних на основі потоку, зазвичай у форматі NetFlow.

1.2.6 NetFlow та IPFIX

Спочатку NetFlow був розроблений Cisco Systems, світовим лідером у галузі мережевих рішень. Багато комутаторів і маршрутизаторів Cisco здатні експортувати записи NetFlow. Використовуються дві версії: NetFlow версії 5 і 9. Перша є пропрієтарною версією Cisco формату, а останній стандартизувався як відкритий протокол IETF у 2006 році.

Потік визначається як односпрямована послідовність пакетів із деякими загальними властивостями, які проходять через мережевий пристрій. Ці зібрані потоки експортуються на зовнішній пристрій, колектор NetFlow. Мережеві потоки дуже деталізовані; наприклад, записи потоку включають такі відомості, як IP-адреси, кількість пакетів та байтів, тимчасові мітки, тип обслуговування (ToS),

РОЗДІЛ 2 МЕТОДИ ВИЯВЛЕННЯ ВТОРГНЕНЬ В КОМП'ЮТЕРНІ МЕРЕЖІ

2.1 Метод виявлення вторгнень на основі сигнатур

Метод виявлення на основі сигнатур - це один із найстаріших методів аналізу безпеки, що широко використовується багатьма комерційними та відкритими IDS.

Сигнатура — це шаблон, який відповідає відомій загрозі. Виявлення на основі сигнатур – це процес порівняння сигнатур зі спостережуваними подіями для виявлення можливих інцидентів. Це найпростіший метод виявлення, оскільки він просто порівнює поточну одиницю активності, таку як пакет або запис у журналі, зі списком підписів за допомогою операцій порівняння рядків, тобто, виявлення працює з «локальною» інформацією. Архітектуру методології виявлення вторгнень на основі сигнатур представлено на рис.2.1.

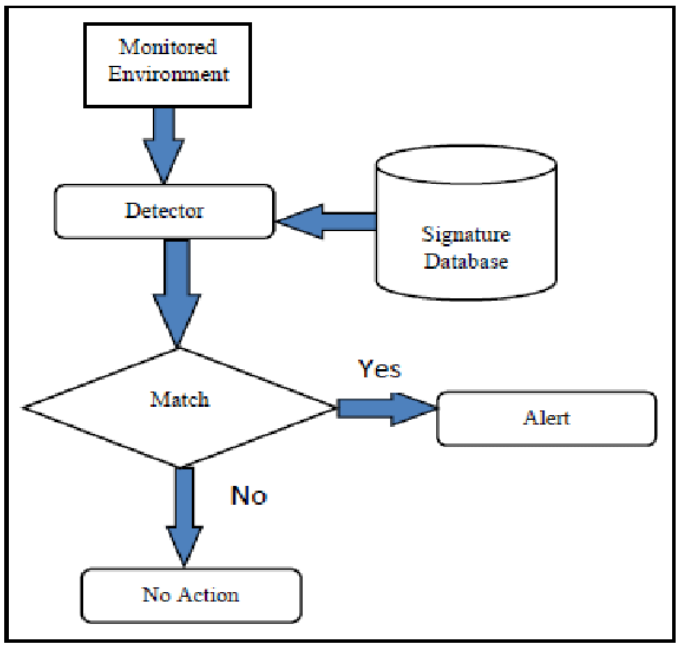


Рисунок 2.1 – Архітектура методології виявлення вторгнень на основі сигнатур

Цей метод дуже ефективний для виявлення відомих загроз, але значною мірою неефективний для виявлення раніше невідомих загроз, загроз,

замаскованих за допомогою методів ухилення, і багатьох варіантів відомих загроз.

Хоча виявлення на основі сигнатур обробляє в основному корисне навантаження пакетів, деякі сигнатури складаються з властивостей, отриманих шляхом збору даних на основі потоку. Тоді можливе граничне виявлення. Однак, якщо використовується вибірка, деякі пакети, що містять підписи, можуть бути втрачені, і, таким чином, ефективність буде нижчою.

Можна зробити висновок, що виявлення на основі сигнатур є низьким, оскільки цим методом виявляються лише відомі атаки, визначені сигнатурами. Ефективність залежить від якості підписів, ризик високого хибного спрацьовування є високим у звичайних мережах, навіть без використання деяких методів уникнення IDS.

Продуктивність методу виявлення на основі сигнатур прийнятна для високошвидкісних мереж лише в тому випадку, якщо вона підтримується апаратним прискоренням. Використання потокових даних як вхідних даних для цього методу є обмеженим. Як правило, IDS на основі підпису страждає від значної затримки під час розгортання абсолютно нового правила (підпису) у такій системі. І останнє, але не менш важливе: метод не може впоратися із зашифрованим корисним навантаженням.

Схему системи виявлення вторгнень на основі сигнатур представлено на рис. 2.2.

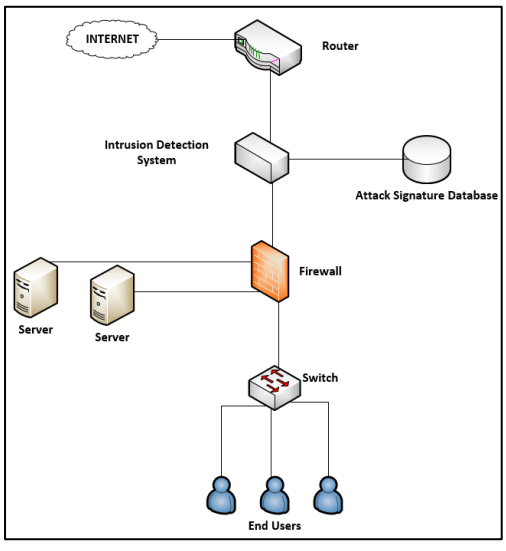


Рисунок 2.2 - Схема системи виявлення вторгнень на основі сигнатур

2.2 Аналіз протоколу з підтримкою стану

Іншим підходом до виявлення вторгнень є аналіз протоколів зі збереженням стану, який працює переважно на вищих рівнях моделі мережі TCP/IP. Аналіз протоколу з урахуванням стану (або глибока перевірка пакетів) — це процес порівняння попередньо визначених профілів загальноприйнятих визначень доброякісної активності протоколу для кожного стану протоколу з подіями, що спостерігаються, для виявлення відхилень. На відміну від виявлення аномалій, воно спирається на універсальні профілі, розроблені постачальником, які визначають, як слід і як не слід використовувати певні протоколи. Це означає, що IDS здатна розуміти та відстежувати стан мережі, транспорту та протоколів додатків, які мають поняття стану. Наприклад, коли користувач починає сеанс протоколу передачі файлів (FTP), сеанс спочатку знаходиться в неавтентифікованому стані. Неавтентифіковані користувачі повинні виконувати лише кілька команд у цьому стані, наприклад переглядати довідкову інформацію або надавати імена користувачів і паролі. Важливою частиною розуміння стану є поєднання запитів із відповідями, тому, коли відбувається спроба автентифікації FTP, IDS може визначити, чи була вона успішною, знайшовши код статусу у відповідній відповіді.

Після успішної автентифікації користувача сеанс перебуває в стані автентифікації і від користувачів очікується виконання будь-яких кілька десятків команд. Виконання більшості цих команд у неавтентифікованому стані вважатиметься підозрілим, але в автентифікованому стані виконання більшості з них вважається доброякісним.

Хоча існують деякі інструменти, що реалізують базовий аналіз протоколу з урахуванням стану, цей метод не є широко поширеним (наприклад, виявлення на основі сигнатур)з наступних причин:

- по-перше, це дуже ресурсомістке завдання, особливо у високошвидкісних мережах. Складність аналізу зростає зі збільшенням кількості (одночасних) сеансів;

- по-друге, він спирається на «знання» всіх проаналізованих протоколів.

Треба зауважити, що існують численні відмінності між реалізаціями різними постачальниками та визначеннями в RFC та інших стандартах. Крім того, можливий лише аналіз відомих протоколів. Далі атаки (наприклад, атаки на відмову в обслуговуванні), які використовують добре сформовані пакети і не порушують нормальну поведінку, не виявляються.

Нарешті, цей метод також впливає на корисне навантаження зашифрованих пакетів.

З іншого боку, метод в цілому забезпечує відносно високу точність. На відміну від методу на основі сигнатур, який шукає відомі шаблони в корисному навантаженні пакета, цей метод працює з сеансами. Цей метод може співвідносити інформацію, отриману з усього сеансу, і забезпечує кращий огляд мережевого трафіку. На рис. 2.3 представлено архітектуру методології виявлення вторгнень на основі аналізу протоколу з підтримкою стану.

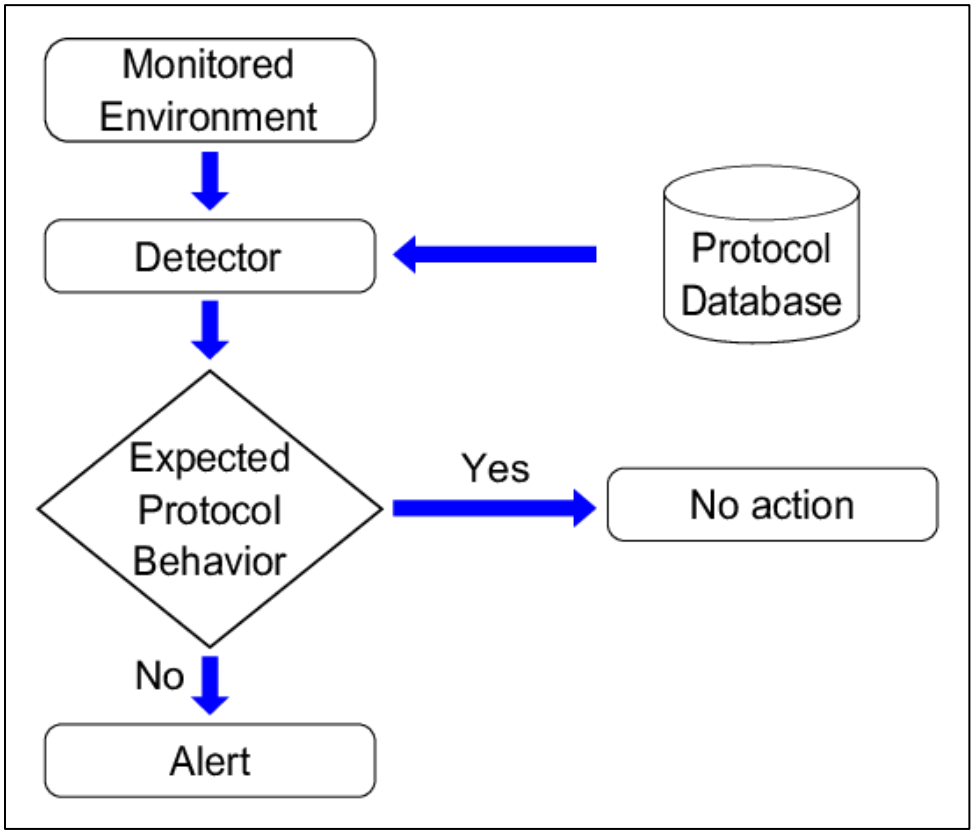


Рисунок 2.3 - Архітектура методології виявлення вторгнень на основі аналізу протоколу з підтримкою стану

Аналіз протоколу з підтримкою стану також може виявляти деякі загрози, які можуть бути пропущені іншими методами, які виконують класифікацію трафіку на основі портів.

Нарешті, але не менш важливо, обмежена підмножина аналізу також може обробляти потоки.

2.3 Метод виявлення на основі аномалій

Метод виявлення на основі аномалій - це процес порівняння визначень того, яка діяльність вважається нормальною, з подіями, що спостерігаються, для виявлення значних відхилень. IDS, що використовує виявлення аномалій, має профілі, які представляють нормальну поведінку таких речей, як користувачі, хости, мережеві підключення або програми. Архітектуру методології виявлення вторгнень на основі аномалій представлено на рис 2.4.

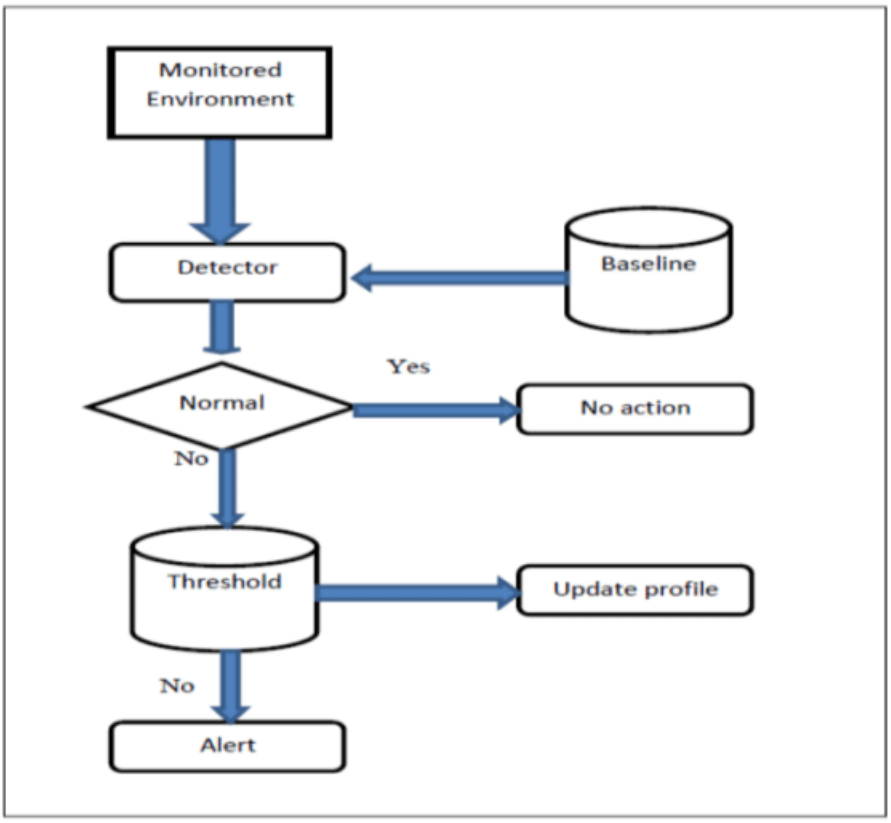


Рисунок 2.4 - Архітектуру методології виявлення вторгнень на основі аномалій

Профілі розробляються шляхом моніторингу характеристик типової діяльності протягом певного періоду часу. Основна перевага методів виявлення аномалій полягає в тому, що вони можуть бути дуже ефективними для виявлення раніше невідомих загроз. Наприклад, припустимо, що комп'ютер заражається новим типом шкідливого програмного забезпечення. Ймовірно, він виконуватиме поведінку, яка суттєво відрізнятиметься від встановлених профілів для комп'ютера. Схему системи виявлення вторгнень на основі аномалій представлено на рис. 2.5.

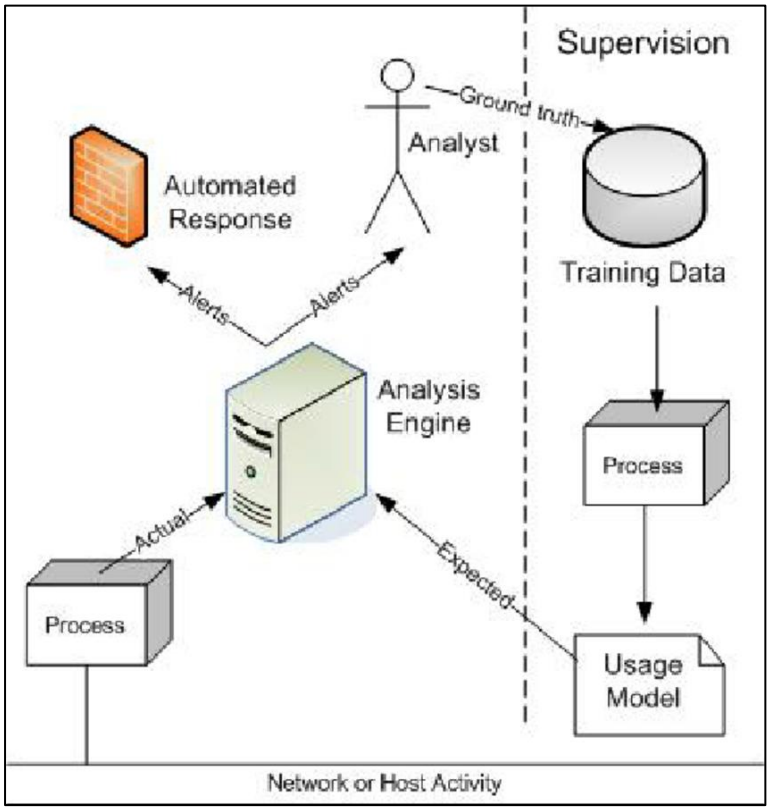


Рисунок 2.5 - Схема системи виявлення вторгнень на основі аномалій

2.3.1 Метод Холта-Уінтерса

Цей метод, також відомий як потрійне експоненціальне згладжування, протягом багатьох років був дуже корисним у багатьох ситуаціях прогнозування. Його вперше запропонував К. С. Холт у 1957 році, і цей метод мав використовуватися для несезонних часових рядів, які не демонструють тенденцій. Пізніше Холт запропонував процедуру (1958), яка враховує тенденції. Уінтерс (1965) узагальнив метод, щоб включити сезонність, звідси й назва

«метод Холта-Уінтерса». Ключові концепції, на яких ґрунтується експоненційне згладжування Холта-Уінтерса, предствлено на рис. 2.6.

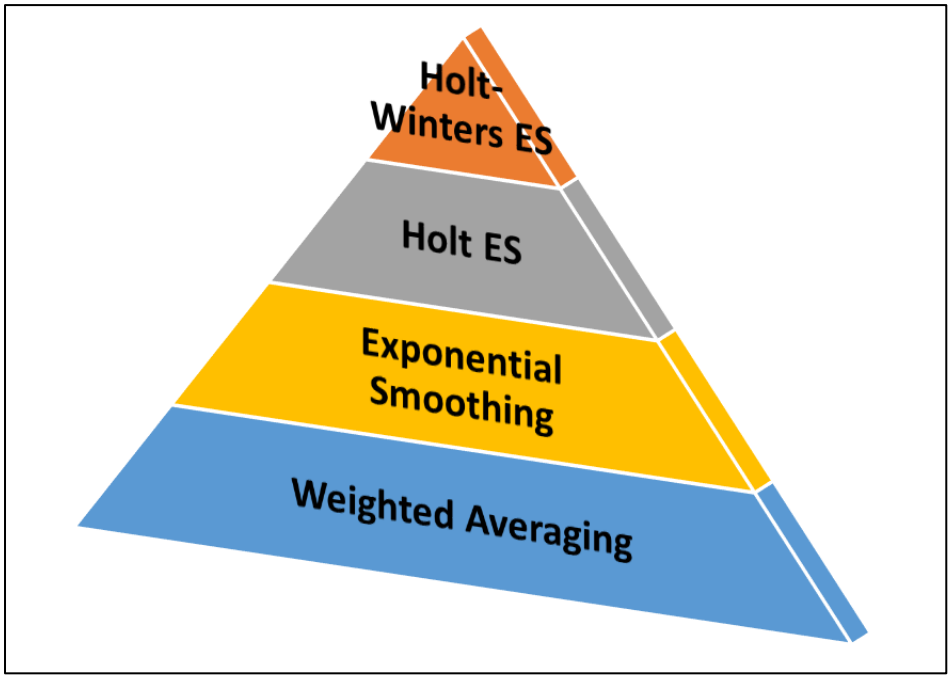


Рисунок 2.6 - Ключові концепції, на яких ґрунтується експоненційне згладжування Холта-Уінтерса

Багато змінних часових рядів мереж обслуговування демонструють такі закономірності (характеристики), які повинні бути враховані моделлю:

- тенденція через деякий час (тобто поступове збільшення запитів демона програми протягом двомісячного періоду через збільшення абонентського навантаження);

- сезонна тенденція або цикл (тобто щодня кількість байтів за секунду зростає вранці, досягає максимуму вдень і знижується пізно ввечері);

- сезонна мінливість (тобто запити на додатки різко коливаються щохвилини в години пік з 16:00 до 20:00, але о 1:00 запити на додатки майже не змінюються);

- поступова еволюція закономірностей (1) – (3) з плином часу (тобто добовий цикл поступово зміщується зі збільшенням кількості вечірніх денних годин із грудня до червня).

Нехай $y_1 ... y_{t-1}, y_t, y_{t+1} ...$ позначає послідовність значень для часового ряду, що спостерігається в деякому фіксованому часовому інтервалі. Позначимо через m період сезонного тренду (тобто кількість спостережень за день). Прогноз Холта-Вінтерса базується на передумові, що спостережуваний часовий ряд можна розкласти на три компоненти: базовий рівень, лінійний тренд і сезонний ефект. Алгоритм припускає, що кожен із цих компонентів змінюється з часом, і це досягається шляхом застосування експоненціального згладжування для поступового оновлення компонентів.

Прогноз є сумою трьох компонентів: $\hat{y}_{t+1} = a_t + b_t + c_{t+1-m}$.

Оновлення формули для трьох компонентів, або коефіцієнтів a, b, c:

$a_t = \alpha(y_t - c_{t-m}) + (1 - \alpha(a_{t-1}+b_{t-1})$, *базова лінія ("перехоплення"),*

$b_t = \beta(a_t - a_{t-1}) + (1-\beta)b_{t-1}$, лінійний тренд ("нахил"),

$c_t = \gamma(y_t - a_t) + (1-\gamma)c_{t-m}$, сезонний тренд.

Нова оцінка базової лінії є спостережуваним значенням, скоригованим на найкращу доступну оцінку сезонного коефіцієнта (c_{t-m}). Оскільки оновлена базова лінія потребує врахування змін через лінійний тренд, прогнозований нахил додається до базового коефіцієнта. Нова оцінка нахилу є просто різницею між новою та старою базовою лінією (оскільки часовий інтервал між спостереженнями фіксований, він не має значення). Нова оцінка сезонної складової є різницею між спостережуваним значенням і відповідним базовим рівнем. α, β, γ — параметри адаптації алгоритму, а $0 < \alpha, \beta < 1$. Більші значення означають, що алгоритм адаптується швидше, а прогнози відображають останні спостереження в часовому ряді; менші значення означають, що алгоритм адаптується повільніше, надаючи більше ваги минулій історії часового ряду.

Простий механізм виявлення аномалії полягає в тому, щоб перевірити, чи не виходить спостережуване значення часового ряду за межі довірчого діапазону. Більш надійним механізмом є використання рухомого вікна з фіксованою кількістю спостережень. Якщо кількість порушень (спостережень, які виходять за межі довірчого діапазону) перевищує вказаний поріг, тоді ініціюється сповіщення про відхилену поведінку.

Налаштування параметрів α, β, γ, довірчий діапазон і порогове значення нечіткі. Параметри моделі потрібно встановити та налаштувати таким чином, щоб модель працювала добре. Не існує єдиного оптимального набору значень, навіть обмеженого даними для однієї змінної. Це відбувається через взаємодію між декількома параметрами в моделі. [3] Автор також дає деякі пропозиції та, більш загально, автори [14]. Те, що тонке налаштування аналізу Холта-Вінтерса не є тривіальним завданням, підтверджується в [11]. Налаштування можуть впливати тренування зазвичай дає задовільні результати. Покриття досить хороше. Оскільки деякі загрози та атаки поводяться подібним чином, можна запропонувати «чутливу» змінну мережевого трафіку, а потім виявити навіть раніше невідомі загрози. Питання продуктивності тісно пов'язане зі збором даних. Теоретично можливі як захоплення пакетів, так і підхід на основі потоку. Останній надає агреговану інформацію в якості вхідних даних для цього методу. Це означає, що сам по собі метод не працює з величезною кількістю даних у (майже) реальному часі. Це забезпечує нижній шар: зонди та колектори на основі потоку. Метод «тільки» обчислює прогноз на основі історичних даних (як правило, останній тиждень [21]). Для розгортання цього методу в проекті GEANT [11] було обрано потоковий підхід (NfSen-HW спирається на дані NetFlow.

2.3.2 Міннесотська система виявлення вторгнень (MINDS)

Система виявлення вторгнень Міннесоті (MINDS) — це система на основі інтелектуального аналізу даних для виявлення мережевих вторгнень. Рисунок 2.7 ілюструє процес аналізу реальних даних мережевого трафіку за допомогою системи.

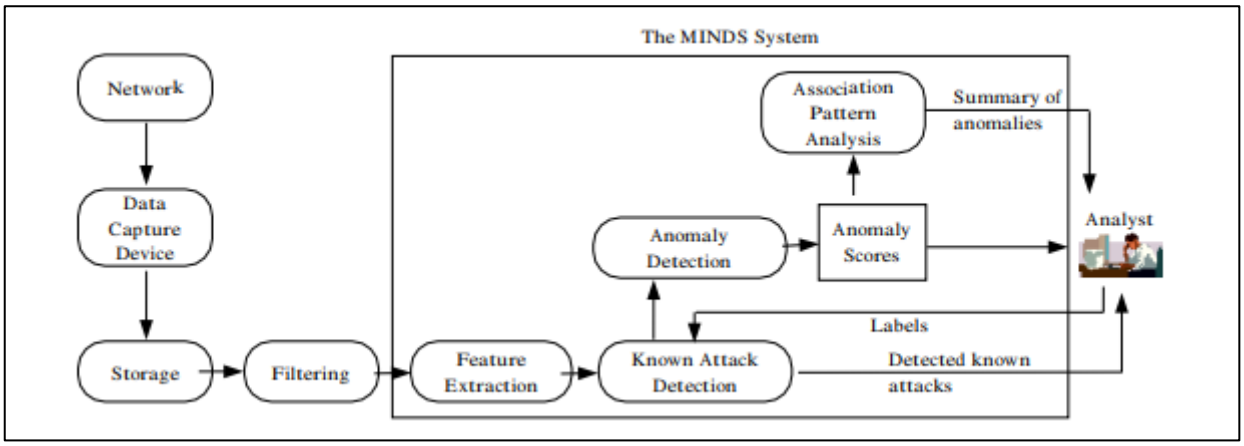


Рисунок 2.7 - Процес аналізу реальних даних мережевого трафіку за допомогою системи MINDS

Основою MINDS є метод виявлення аномалій, який призначає оцінку кожному мережевому з'єднанню, що відображає, наскільки аномальним є з'єднання, і модуль на основі аналізу шаблонів зв'язків, який узагальнює ті мережеві з'єднання, які модуль виявлення аномалій оцінює як дуже аномальні.

Вхідними даними для MINDS є зібрані дані Netflow за допомогою flow-tools. Flow-tools збирають лише інформацію заголовка пакета (тобто не фіксують вміст повідомлення), а створюють односторонні сеанси (потоки).

Дані Netflow для 10-хвилинних вікон, що зазвичай обробляють до 1–2 мільйонів потоків, зберігаються в плоских файлах. Аналітик використовує MINDS для аналізу цих 10-хвилинних файлів даних у пакетному режимі. Причина пакетної роботи системи полягає в тому, що режим налізу не пов'язаний із часом, який потрібен для аналізу цих файлів, але він зручний для роботи аналітика. Перед подачею даних у модуль виявлення аномалій аналітиком поступово виконується фільтрація даних для видалення мережевого трафіку, в аналізі якого він не зацікавлений. Наприклад, відфільтровані дані можуть включати трафік із надійних джерел або незвичайну/аномальну поведінку мережі, яка завідомо повинна бути безпечною.

Першим кроком у MINDS є виділення функцій, які використовуються в аналізі отриманих даних. Основні функції включають IP-адреси вихідного пункту та пункту призначення, порти вихідного пункту та пункту призначення,

протокол, прапори, кількість байтів і кількість пакетів. Похідні функції включають функції на основі часових вікон і вікон підключення. Функції на основі часових вікон створені для захоплення з'єднань із подібними характеристиками за останні T секунд. На рис. 2.8 представлено схему функціонування системи виявлення вторгнень Міннесоті (MINDS).

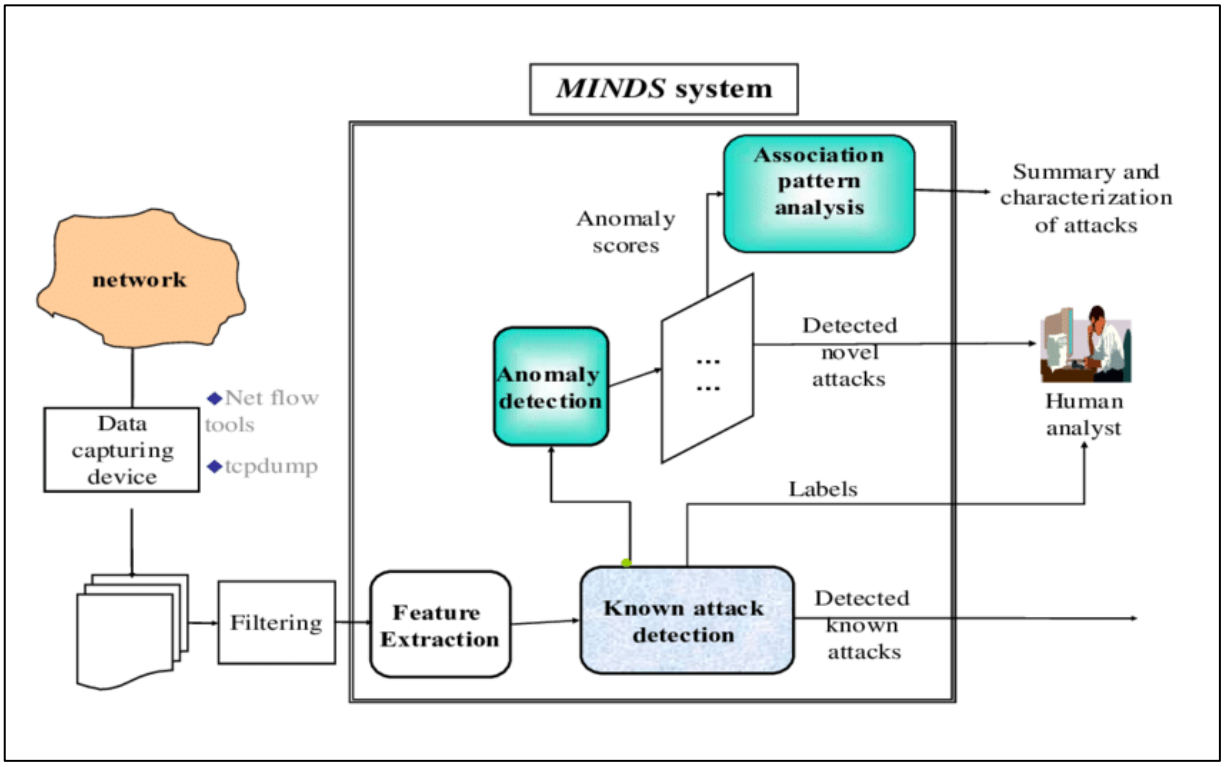


Рисунок 2.8 - Схема функціонування системи виявлення вторгнень Міннесоті (MINDS)

В таблиці 2.1 наведено функції на основі часових вікон.

Таблиця 2.1 - Функції на основі часових вікон

Назва функції	Опис функції
count-dest	Кількість потоків до унікальних IP-адрес призначення всередині мережі в останні *T* секунд з того самого джерела
count-src	Кількість потоків з унікальних IP-адрес джерела всередині мережі в останні *T* секунд до того самого пункту призначення
count-serv-src	Кількість потоків з IP-адреси джерела на той самий порт призначення в останні *T* секунд
count-serv-dest	Кількість потоків до IP-адреси призначення з використанням того самого порту джерела в останні *T* секунд

«Повільне» сканування, тобто таке сканування, при якому скануються хости (або порти). І яке використовує набагато більший часовий інтервал, ніж кілька секунд, напр. один дотик за хвилину або навіть один дотик за годину, не можна відокремити від решти трафіку за допомогою функцій на основі часових вікон.

Для цього необхідно також отримувати функції на основі вікон підключення, які фіксують подібні характеристики з'єднань, що й функції на основі часових вікон, але обчислюються за допомогою останніх N з'єднань, що надходять з(прибувають) з різних джерел до (пунктів призначення).

Функції на основі вікон підключення, наведено в таблиці 2.2.

Таблиця 2.2 - Функції на основі вікон підключення

Назва функції	Опис функції
count-dest-conn	Кількість потоків до унікальних IP-адрес призначення всередині мережі в останніх N потоках з того самого джерела
count-src-conn	Кількість потоків з унікальних IP-адрес джерела всередині мережі в останніх N потоках до того самого пункту призначення
count-serv-src-conn	Кількість потоків з IP-адреси джерела на той самий порт призначення в останніх N потоках
count-serv-dest-conn	Кількість потоків до IP-адреси призначення з використанням того самого порту джерела в останніх N потоках

Після етапу створення функції використовується відомий модуль виявлення атак на мережеві підключення, які відповідають атакам, для яких доступні сигнатури, а потім видалити їх із подальшого аналізу.

Далі дані надходять у модуль виявлення аномалій MINDS, який використовує алгоритм виявлення аномалій для призначення оцінки аномалії для

кожного мережевого підключення. Потім аналітику доводиться розглядати лише найбільш аномальні зв’язки, щоб визначити, чи вони є справжніми атаками, чи іншою цікавою поведінкою.

2.3.3 Модуль виявлення аномалій MINDS

Модуль виявлення аномалій MINDS підсумовує мережеві з’єднання, які модуль виявлення аномалій оцінює їх як високо аномальні. Аналітик забезпечує зворотний зв'язок після аналізу створених зведень і вирішує, чи ці зведення є корисними для створення нових правил, які можна використовувати у модулі виявлення атак. Модуль виявлення аномалій MINDS призначає кожній аномалії точку даних, яка називається фактором локальних викидів (LOF) [5]. Коефіцієнт викиду в точці даних є локальним в тому сенсі, що він вимірює ступінь викиду по відношенню до власного мережевого оточення.

Аналіз виконується на 10-хвилинних вікнах даних NetFlow.

По-перше, виконується вилучення функції. MINDS представляє два типи функцій, похідних від функцій стандартного NetFlow:

- на основі часових вікон, з’єднання з подібними характеристиками за останні *T* секунд;

- на основі вікон з’єднань, останніх N з’єднань, що походять (прибувають до) різних джерел (пунктів призначення).

Перші, очевидно, не включають шкідливі дії (такі як приховане сканування портів), які тривають більше T секунд. Отже, він доповнюється останнім. Функції на основі часових вікон: count-dest, кількість потоків до унікальних IP-адрес призначення всередині мережі за останні T секунд з того самого джерела, count-src, кількість потоків з унікальних IP-адрес джерела всередині мережі за останні T секунд до того самого пункту призначення, count-serv-src, кількість потоків з IP-адреси джерела на той самий порт призначення за останні T секунд count-serv-dest, кількість потоків до IP-адреси призначення з використанням того самого вихідного порту за останні T секунд.

По-друге, дані надходять в модуль виявлення аномалій MINDS, який використовує *алгоритм виявлення аномалій* для присвоєння *локального фактору аномалії*, оцінки аномалії кожному мережевому з'єднанню. Фактор точки виявлення аномалій є *локальним* у тому сенсі, що він вимірює ступінь відхилення по відношенню до її мережевого оточення. Для кожного прикладу даних спочатку обчислюється щільність мережевого оточення. LOF даних конкретного зразка p являє собою середнє значення співвідношень щільності зразка p і щільності його сусідів.

Для ілюстрації переваг підходу LOF розглянемо простий двовимірний набір даних, наведений на рис. 2.9. На рисунку видно, що щільність кластера C_2 значно вища, ніж щільність кластера C_1. Через низьку щільність кластера C_1 для більшості зразків q всередині кластера C_1 відстань між зразком q та його найближчим сусідом більша ніж відстань між зразком p_2 і його найближчим сусідом, який знаходиться всередині кластера C_2, і тому зразок p_2 не буде вважатися викидом.

Отже, простий підхід найближчого сусіда, оснований на обчисленні відстаней, не працює у цих сценаріях. Однак зразок p_1 може бути виявлений як викид за допомогою відстані до найближчих сусідів. З іншого боку, LOF здатний захопити і ті, і інші викиди через те, що враховується щільність навколо зразків.

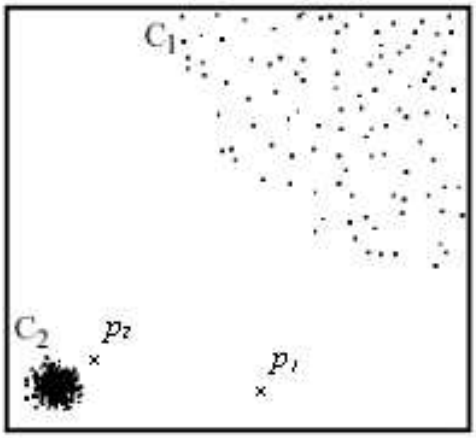


Рисунок 2.9 - Приклад 2-D викиду

Нарешті, модуль аналізу патернів асоціацій MINDS узагальнює мережеві зв'язки, які оцінюються модулем виявлення аномалій як дуже аномальні. У цьому модулі також використовуються деякі методи виявлення на основі сигнатур.

РОЗДІЛ 3 МЕТОДИ ВІЗУАЛІЗАЦІЇ РЕЗУЛЬТАТІВ АНАЛІЗУ ВИЯВЛЕННЯ ЗАГРОЗ

Основна проблема аналізу – це осмислення результатів всього процесу. Ми можемо отримувати дані, які (дійсно) відображають мережевий трафік, і обробляти їх різними методами. Однак, якщо ми не використовуємо жодної техніки інтелектуального аналізу даних, нам все одно доведеться інтерпретувати результати вручну. Це цілком можливо в невеликих мережах, але абсолютно немислимо у високошвидкісних мережах, оскільки людина не в змозі оцінити великий обсяг інформації. Візуалізація має допомогти нам і представити важливу інформацію в іншому та більш зручному вигляді.

Наприклад, tcpdump є найбільш використовуваним інструментом для моніторингу мережі та збору даних. Це інструмент командного рядка, який може читати пакети з мережевого інтерфейсу або файлу даних і відображати кожен пакет у новому рядку під час виведення. На відміну від цього, аналізатор мережевих пакетів Wireshark використовує графічний інтерфейс користувача (GUI) і, наприклад, «розфарбовує» відображення пакетів на основі фільтрів. Насправді інструмент класифікує процеси та результати представлені різними кольорами. Ми також можемо інтерактивно переглядати отримані дані, переглядати підсумкову та детальну інформацію для кожного пакета. Ми підтверджуємо, що такі (невеликі) покращення полегшують аналіз.

Однак візуалізацією є не тільки використання кольорів. Візуалізація є невід’ємною частиною сучасного аналізу безпеки. Ми окреслюємо деякі способи візуалізації в сучасних програмних засобах та оцінюємо їхній внесок у прискорення аналізу. Ми в основному зосереджуємось на програмному забезпеченні з відкритим кодом, яке візуалізує захоплений мережевий трафік у форматі pcap або NetFlow. У той час як дані у форматі pcap містять заголовки

пакетів і корисне навантаження, записи NetFlow навмисно пропускають корисне навантаження.

3.1 Діаграми

Основним інструментом візуалізації є діаграма. Існує багато інструментів, що розширюють базове програмне забезпечення, яке виконує лише збір даних. Ці інструменти часто будують двовимірні діаграми, які відображають часові ряди відстежуваних значень або їхніх сукупностей. Це простий і тому широко поширений метод візуалізації. Зокрема, NfSen інтегрує результати nfdump з різними діаграмами, які показують часові ряди загальної кількості пакетів, потоків і обсягу трафіку. На рис. 3.1 представлено діаграму обсягу мережевого трафіку в NfSen.

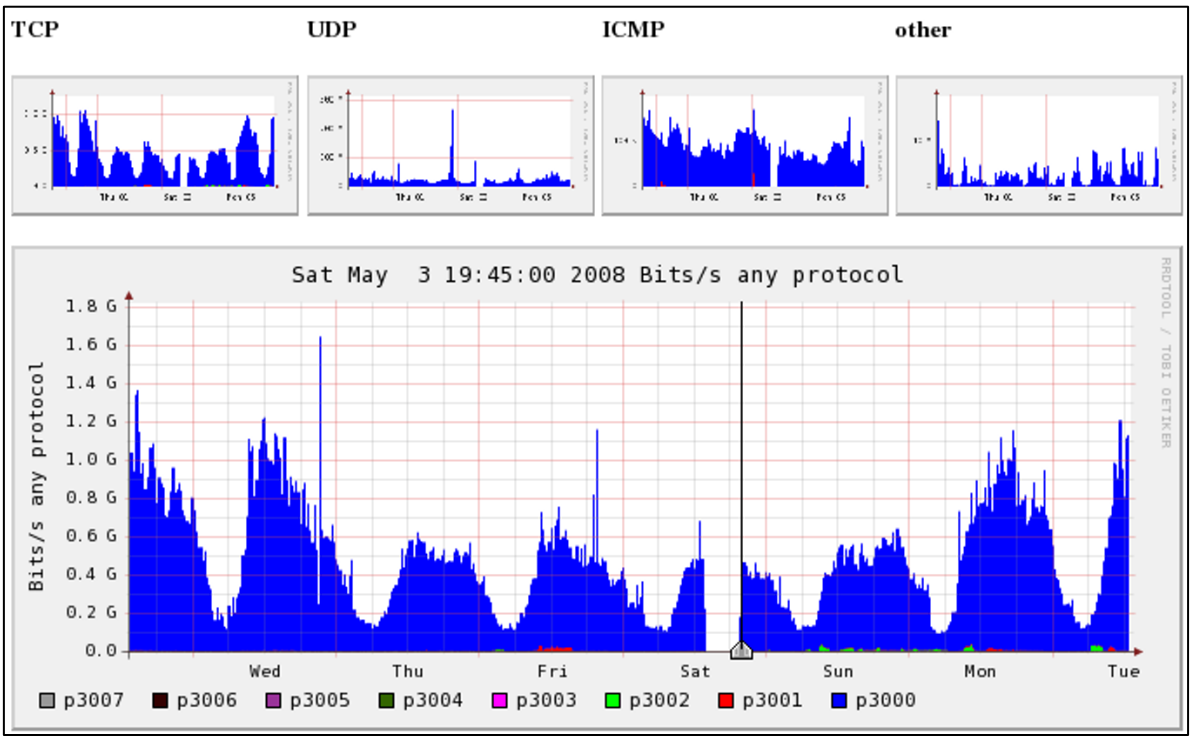


Рисунок 3.1 - Діаграма динамічного діапазону мережевого трафіку в NfSen

Діаграми також використовуються в інших інструментах, таких як FlowScan, Java Netflow Collect-Analyzer, ntop, nfstat, NetFlow Monitor, Caligare Flow Inspector або Stager.

Діаграми також використовуються для моніторингу мережі. Адміністратор мережі може легко переглянути відповідну діаграму та негайно прийняти рішення, якщо сталася мережева або аномальна небезпека. У таких випадках відповідна крива зазвичай різко зростає або падає.

3.2 Картографування в просторі

Цей метод візуалізації дозволяє малювати точки в дво- або квазі-тривимірному просторі, який відображається на екрані. Він використовує стереоскопічний зір людини та «перетворює» шаблони в отриманих даних на графічні візерунки у визначеному просторі.

Наприклад, *The Spinning Cube of Potential Doom* — це анімоване візуальне відображення мережевого трафіку. Кожна вісь куба представляє різні компоненти TCP-з'єднання: X — локальний простір IP-адрес, Z — глобальний простір IP-адрес, а Y — номери портів, які використовуються в з'єднаннях для пошуку служб і координації зв'язку (наприклад, 22 для SSH і 80 для HTTP). З'єднання TCP, як спроби, так і успішні, відображаються як окремі точки для кожного з'єднання.

Успішні TCP-з'єднання відображаються білими точками. Неповні TCP-з 'єднання відображаються у вигляді кольорових точок.

Неповні з'єднання - це спроби зв'язатися з неіснуючими системами, які більше не прослуховують цей конкретний номер порту. Куб розфарбовує неповні з'єднання за допомогою райдужної кольорової карти зі зміною кольору за номером порту.

Кольорове зіставлення допомагає спостерігачам знайти точку в 3D-просторі. [6]. Наприклад, сканування портів у захоплених даних створює лінію в кубі. як представлено на рис. 3.2. Це більш корисний і ефективний погляд на таку подію, ніж ручна перевірка tcpdump або навіть вихідних даних Wireshark. Продовженням куба є InetVis [16]. Аналогічний підхід також використовує Фламінго [9]. PortVis [29] і *tnv* [45] скоріше використовують двовимірний простір.

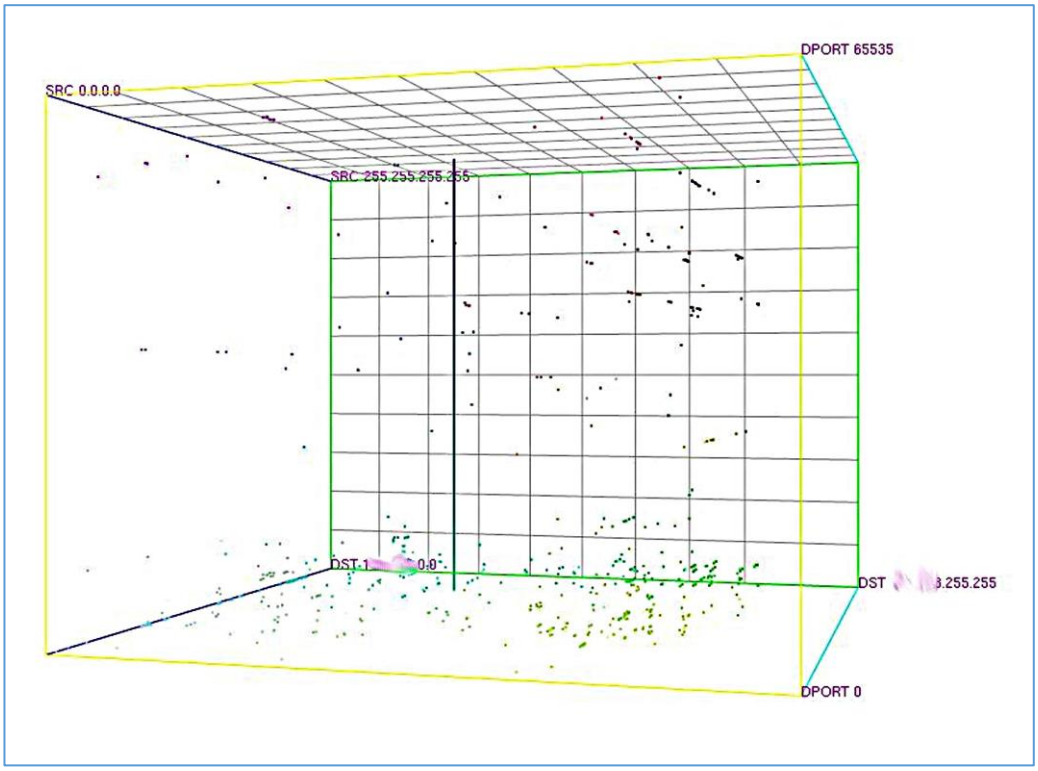


Рисунок 3.2 - Сканування портів у GPL Cube of Potential Doom, мережевому візуалізаторі GPL, основаному на обертовому кубі Potential Doom

3.3 Графи

Природним представленням мережевого трафіку є граф, де вершини відповідають хостам, а (орієнтовані) ребра - зв'язкам (потокам), захопленим між хостами (рис. 3.3). Ця структура зображує, *хто* з ким спілкується. Для порівняння, класичне представлення мережевого трафіку наведено на рисунку 3.4. NfVis10 розшифровується як NetFlow Visualizer і є *інструментом перевірки концепції*, заснованої на інструментарії *візуалізації pref11*. Представлення трафіку на основі графіка доповнено кількома важливими функціями. Користувач може перерахувати потоки та статистику трафіку, пов'язану з кожним краєм/хостом. Трафік можна фільтрувати та агрегувати за багатьма релевантними функціями.

Візуальні атрибути дисплея (такі як розмір вузла/краю та колір) також можуть адаптуватися до цих характеристик, полегшуючи орієнтацію користувача.

Інформація, надана «третіми сторонами» (DNS і whois), легко інтегрується у візуалізацію. Оскільки поточний мережевий трафік є безмасштабною мережею, особливо важливо обробляти візуалізацію *супервузлів*, тобто вузлів з великою кількістю з'єднань. Ці вузли характерні для багатьох сценаріїв атаки, а також для високоцінних цілей. Таким чином, візуалізатор замінює одноразові з'єднання до/з цих хостів спеціальним представленням «хмари» трафіку, і лише виділяє вузли, які також з'єднуються з іншими вузлами спостережуваної мережі. [5].

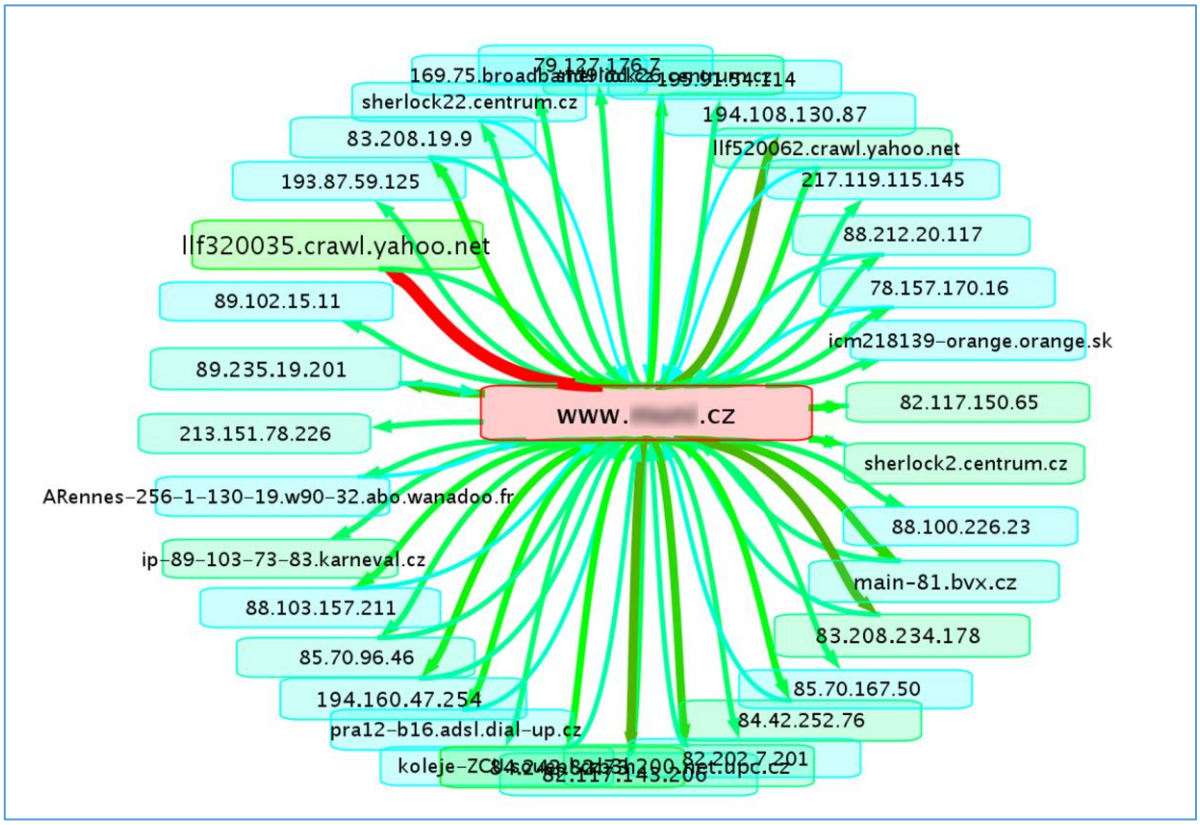


Рисунок 3.3 - Мережевий трафік у вигляді графа

MyNetScope12 — це мережева платформа візуальної аналітики, основана на стандартних даних NetFlow і гетерогенних джерелах даних. Вона розвиває NfVis двома важливими способами. По-перше, вона може включати інші зовнішні джерела даних, такі як роздільна здатність DNS, відповідь служби. дані відповіді щодо різних аномалій. методи виявлення і інформацію про топологію мережі.

117.50.86.71	148.228.205.218	TCP	315	37	80	66796
107.252.174.21	148.228.205.218	TCP	17	1	80	1135
148.228.205.218	107.252.174.21	TCP	16	1	42587	17382
96.207.181.5	148.228.205.218	TCP	6	1	80	684
148.228.205.218	96.207.181.5	TCP	6	1	1933	737
123.64.213.194	148.228.205.218	TCP	1	1	80	40
148.228.205.218	117.50.86.71	TCP	7	1	4632	990
148.228.205.195	121.47.105.162	TCP	196	2	1346	274595
148.228.205.218	117.50.86.71	TCP	18	1	4684	19235
148.228.205.218	117.50.86.71	TCP	6	1	4685	4335
148.228.205.218	117.50.86.71	TCP	1	1	4635	40
121.47.105.162	148.228.205.195	TCP	169	4	443	6834
148.228.205.195	121.47.105.162	TCP	79	2	1352	107215
148.228.205.218	117.50.86.71	TCP	9	1	4687	6867
148.228.205.218	117.50.86.71	TCP	6	1	4690	1792
148.228.205.218	117.50.86.71	TCP	8	1	4686	1666
148.228.205.218	117.50.86.71	TCP	18	1	4689	15058
148.228.205.218	117.50.86.71	TCP	8	1	4691	1804
148.228.205.218	117.50.86.71	TCP	5	1	4694	1868
121.37.81.102	148.228.205.195	TCP	93	9	443	8030
148.228.205.218	117.50.86.71	TCP	22	2	4695	21238
148.228.205.218	117.50.86.71	TCP	11	2	4696	8111
148.228.205.218	117.50.86.71	TCP	16	2	4692	5753
148.228.205.218	117.50.86.71	TCP	8	1	4688	4400
237.238.8.219	148.228.205.218	TCP	226	16	80	18888
148.228.205.218	116.204.159.253	TCP	14	1	2118	18957
148.228.205.218	116.204.159.253	TCP	4	1	2117	3762
116.204.159.253	148.228.205.218	TCP	18	4	80	3450
101.249.217.238	148.228.205.218	TCP	13	1	80	941
148.228.205.218	101.249.217.238	TCP	12	1	44531	13927
231.39.29.116	148.228.205.218	TCP	32	3	80	2148
148.228.205.218	116.135.87.175	TCP	13	2	4842	16840
148.228.205.218	254.166.192.97	TCP	26	1	1575	31400
254.166.192.97	148.228.205.218	TCP	15	8	80	4090
116.135.87.175	148.228.205.218	TCP	7	1	80	1026
107.252.188.190	148.228.205.218	TCP	1525	46	80	111012
148.228.205.218	107.252.188.190	TCP	15	1	59975	16354
148.228.205.195	121.37.81.102	TCP	64	2	1882	78651
148.228.205.218	117.50.86.71	TCP	7	3	4636	280

Рисунок 3.4 - Мережевий трафік як перелік потоків

По-друге, це масштабоване рішення навіть для широких мереж. Інтеграція зовнішніх джерел даних дуже вітається, оскільки зазвичай аналітик безпеки працює лише з первинними даними, такими як виходи *tcpdump* або записи NetFlow. Як правило, йому доводиться збирати додаткову інформацію з інших доступних джерел. В іншому випадку повна перевірка інциденту з безпекою неможлива.

Також відомі інші інструменти візуалізації на основі графів

VisFlowConnect-IP візуалізує мережевий трафік у вигляді графа паралельних осей з хостами як вузлами та потоками трафіку в якості ліній, що з'єднують ці вузли. Потім ці графи можна анімувати з часом, щоб виявити тенденції. Кооперативна асоціація з аналізу інтернет-даних (CAIDA) розробляє два цікаві інструменти.

LibSea13 є одночасно форматом файлу та бібліотекою Java для представлення великих орієнтованих графів на диску та в пам'яті. Масштабованість до графів з мільйоном вузлів була основною метою. Додатковими цілями були виразність, компактність і підтримка конвенцій і політик для конкретних додатків.

Walrus – інструмент для інтерактивної візуалізації великих орієнтованих графів15 у тривимірному просторі. Використовуючи спотворення, схоже на риб'яче око, він забезпечує дисплей, який одночасно показує локальні деталі та глобальний контекст. Незважаючи на те, що вони не є спеціалізованими додатками для візуалізації мережевого трафіку, було б корисно об'єднати їх для цієї мети, якщо існував інструмент, що забезпечує виведення у форматі LibSea.

Висновки до розділу 3

Візуалізація є важливою для аналізу безпеки, при цьому представлено три методи та інструменти, які вони використовують. Загально використовувані діаграми згодом були доповнені методами, які використовують відображення в просторі та представлення мережевого трафік у вигляді графів. Також підсумовано їхній внесок в аналіз безпеки. Природно, що прогрес засобів візуалізації пов'язаний з розробкою інструментів, які збирають та/або обробляють мережеві дані.

Хороший інструмент візуалізації повинен відображати складну картину мережевого трафіку, в ідеалі з поміченими актуальними інцидентами безпеки. Однак усі наявні відомості про хости та їх комунікацію також повинні бути. за вимогою. відображені в добре впорядкованих таблицях, діаграмах та списках.

РОЗДІЛ 4 ПРОЕКТУВАЕННЯ СИСТЕМИ ВИЯВЛЕННЯ ВТОРГНЕНЬ IDS

В попередніх розділах було оцінено кілька підходів до виявлення вторгнень, а також методи візуалізації мережевого трафіку. У цьому розділі розглянуто проектування системи виявлення вторгнень для великих мереж. Спочатку необхідно визначити та обґрунтувати вимоги до таких IDS, а потім розробити рішення, яке відповідає цим вимогам.

Перш за все, було вибрано систему виявлення вторгнень. На відміну від системи запобігання вторгнень (IPS), вона «лише» відстежує мережевий трафік і попереджає оператора в разі інциденту безпеки. Отже, вона аналізує інцидент і врешті-решт забезпечує його пом'якшення. Якщо ми розгорнемо IPS, і вона попередить про помилкове спрацьовування, вона негайно заблокує легітимне мережеве з'єднання. Ще одна причина вибору систему виявлення вторгнень полягає в тому, що IPS має бути вбудованою (частиною з’єднання. Коли IPS виходить з ладу, може вийти з ладу і вся мережа. Отже, вибір системи виявлення вторгнень грунтується через виникнення помилкових спрацьовувань та збоїв системи. Це є основною причиною розгортання IDS.

4.1 Вимоги до системи виявлення вторгнень IDS

4.1.1 Точність

Точність є фундаментальною вимогою будь-якої системи IDS. Однак виконати цю вимогу для нинішніх систем дуже складно. Вони страждають від високого рівня помилкових спрацьовувань. Крім того, існують деякі методи ухилення від IDS, такі як *пронизливий звук*. У зв'язку з цим, IDS не є широко прийнятими та розгорнутими мережевими адміністраторами. Високий відсоток помилкових спрацьовувань перевантажує адміністраторів, які все одно зайняті. Навпаки, хибнонегативні результати неможливо виявити в рутинній експлуатації. Таким чином, IDS створює «хибне відчуття безпеки».

4.1.1 Виявлення нових загроз

На сьогоднішній день існує багато IDS, здатних виявляти відомі загрози, особливо ідентифікатори на основі сигнатур, такі як Snort. Їх недоліком є те, що база правил таких IDS повинна підтримуватися адміністратором мережі або безпеки. Крім того, нові загрози включаються в базу правил вручну, часто сторонніми постачальниками. Нарешті, очевидно, що ці системи не здатні до виявлення нових загроз. Таким чином, запропонована IDS повинна виявляти навіть нові загрози за допомогою більш ефективного механізму виявлення.
Це необхідно, щоб уникнути деяких атак, таких як (розподілена) відмова в обслуговуванні (DDOS і DOS), коли зловмисник наповнює мережу пакетами, призначеними для IP-адреси IDS. По-друге, IDS не повинен помітно впливати на топологію мережі та мережевий трафік. А саме, затримка повинна зберігатися, а IDS не повинен даремно завантажувати мережеві з’єднаня.

4.1.2 Надійність безпеки

Зрозуміло, що IDS привертають увагу зловмисників. Сама система IDS повинна бути невразливою і стійкою до загроз безпеці. Можна запобігти деяким атакам, якщо виконувати попередню вимогу прозорості на рівні IP. Далі цілісність IDS повинна бути неушкодженою. Наприклад, якщо IDS складається з декількох компонентів, їх зв'язок може бути порушений або підслуханий . У будь-якому випадку адміністратор безпеки повинен отримати правдиві результати виявлення.

4.1.3 Виявлення аномалій у зашифрованому трафіку

Багато поточних IDS зазнають невдачі у виявленні загроз у зашифрованому мережевому трафіку. Такі системи покладаються на перевірку корисного навантаження. Запропоновані IDS повинні розпізнавати аномалії навіть у

зашифрованому трафіку, оскільки все більше мережевих сервісів використовують шифрування.

4.1.4 Зручний інтерфейс і добре продумана візуалізація

Останньою, але не менш важливою вимогою є інтерфейс користувача. Якщо IDS відповідає всім попереднім вимогам, але представлення результатів не є належним чином організованим, IDS не придатний для використання. З одного боку, інтерфейс повинен бути корисним для користувача і пропонувати всі доступні види даних. З іншого боку, він повинен забезпечувати підтримку повторюваних транзакцій і детальний перегляд. Інтерфейс повинен бути персоналізований користувачем.

4.2 Реалізація системи виявлення атак

Мережеві ідентифікатори (NIDS) повинні відповідати таким вимогам:

- масштабованість,
- простота в обслуговуванні,
- Надійність безпеки.

На відміну від Host-based IDS (HIDS), розгортання нового хоста в мережі не вимагає додаткових зусиль для моніторингу мережевої активності нового хоста. Немає необхідності встановлювати на хост будь-яке спеціалізоване програмне забезпечення. Слід звернути увагу, що мережа може складатися з деяких спеціалізованих хостів (крім звичайних серверів або робочих станцій). Отже, налаштування HIDS в такому випадку неможливе. Далі, NIDS – це пасивні пристрої, «невидимі» для зловмисників. Навпаки, HIDS покладаються на процеси, які працюють в операційній системі хоста. Далі також буде розглянуто розгортання, тестування та можливий апгрейд IDS. Як правило, оновити один компонент NIDS простіше, ніж багато компонентів HIDS на хостах.

В цій роботі запропоновано систему, яке складається з декількох компонентів і шарів. *Мережеві зонди* є «очима та вухами» запропонованої системи виявлення вторгнень. *Колектори* - це "пам'ять", MyNetScope *з джерелами даних - це "мозок і серце", а консоль MyNetScope* - це "рот" IDS. «Нервова система і кровообіг» представлена мережевими ланками, які з'єднують всі частини між собою. Адже архітектура (рисунок 4.1) схожа на архітектуру CAMNEP, зображену на рисунку 2.1.

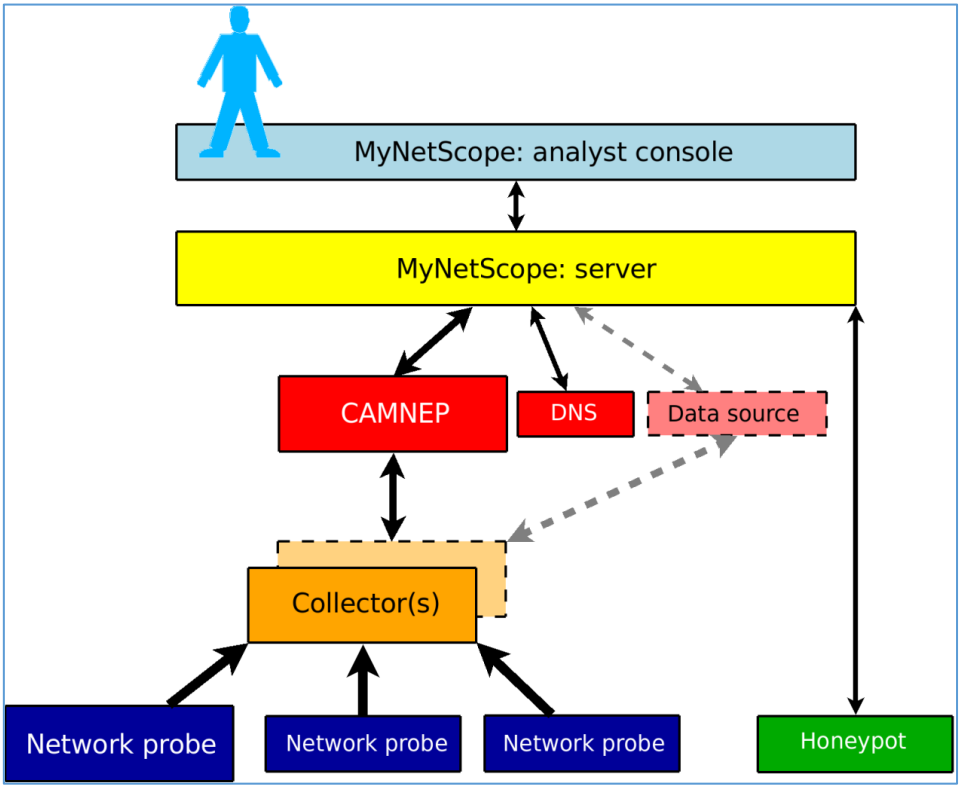


Рисунок 4.1 - Архітектура запропонованої системи

4.2.1 Мережеві зонди

Зонди створюють нижній шар нашої системи. Вони збирають мережевий трафік і обслуговують колекторів із захопленими даними. У цьому розділі описано можливості зондування та розгортання зондування у мережі, що адмініструється.

Мережеві зонди відстежують зв'язок і експортують отримані дані в формат NetFlow. Було вирішено, що цей формат відповідає вимогам щодо

роботи в мультигігабітних мережах. Було вирішено здійснювати реалізацію без використання SNMP-лічильників і трасування пакетів. Перший надає грубі дані, а другий дуже складний. Захоплення та зберігання пакетів на швидкості дроту практично неможливе навіть за допомогою спеціалізованого обладнання.

Було вирішено покладатися на дані NetFlow, які експортують деякі (периферійні) маршрутизатори Cisco, які можуть існувати в сучасній мережі. Не тільки даніі вимірювання показують, що маршрутизатори Cisco не правильно експортувати NetFlow за будь-яких обставин. Очевидно, що основним завданням маршрутизатора є маршрутизація мережевого трафіку. Треба враховувати, що експорт NetFlow є додатковою функцією. З іншого боку, дані NetFlow з маршрутизаторів можуть бути додатковим джерелом даних для запропонованої системи.

Рекомендується використовувати зонди за вартістю (комерційні готові комп'ютери) через їх вартість. існує дві альтернативи мережевих інтерфейсних карт (NIC), які використовуються в зондах. Перші використовують поширені NIC (наприклад, Intel), а другі покладаються на технологію COMBO, розроблену в проекті Liberouter2. Програмні зонди, які перехоплюють мережевий трафік NIC (наприклад, nprobe), недостатньо ефективні. [20] Тому необхідно розгортати Flow-Mon, зонд пасивного моніторингу мережі з апаратним прискоренням. [10] Як правило, зонди софт-ware є задовільними для малих мереж, апаратно-прискорені зонди для великих, багатогігабітних мереж. Обидва типи зондів відповідають вимозі прозорості, оскільки вони «невидимі» на рівні IP. Інтерфейсу, що виконує захоплення пакетів, не призначено IP-адресу. IPv6 підтримується завдяки використанню NetFlow версії 9.

Мережевий зонд відстежує трафік, що проходить через певний вузол мережі.

Таким чином, розташування мережевого зонда визначає, що відстежується. Це дуже важливо, оскільки запропонована система базується

на даних, наданих мережевими зондами. По суті, кожен пакет, який входить або виходить з керованої мережі, повинен проходити через місце, де знаходиться зонд. Ми обговорюємо це з мережевими адміністраторами мережі кампусу університету Масарика. Ми визначаємо, що зонди повинні бути розташовані "поблизу" периферійного маршрутизатора, враховуючи мережевий трафік від/до In- ternet.

На рис. 4.2 показано розташування основного зонда. Було зроблено вибір між двома альтернативними варіантами. При цьому припускається, що периферійний маршрутизатор також діє як брандмауер. Якби зонд було розміщено перед маршрутизатором/брандмауером, можна б було також відстежувати трафік, який не потрапив би в мережу, що адмініструється Серед двох варіантів було обрано другий варіант. Основний зонд знаходиться в мережі, що адмініструється, за маршрутизатором/брандмауером. Це гарантує, що зонд «побачить» лише трафік, який пройшов через брандмауер. Брандмауер зазвичай реалізує (частину) політики безпеки організації.

Як вже зазначалося вище, ми не будемо вставляти щуп в мережеве посилання, а тільки мережевий робочий кран. Це апаратний пристрій, який забезпечує спосіб доступу до даних, що протікають через комп'ютерну мережу3. Таким чином, ми фактично делегуємо відповідальність за безперервну роботу крана. Якщо ми використовуємо кран, який потребує живлення, ми повинні підключити його до джерела безперебійного живлення (ДБЖ). Також слід вибрати змішувач з подвійним блоком живлення на випадок виходу з ладу.

Основний зонд здатний фіксувати тільки ті атаки, які походять з мережі або призначені для неї. Щодо атак інсайдерів, пропонується розгорнути інші зонди всередині нашої мережі, особливо перед/за брандмауерами, які захищають певні сегменти мережі. Тоді можна виявити можливу шкідливу діяльність хостів у нашій мережі.

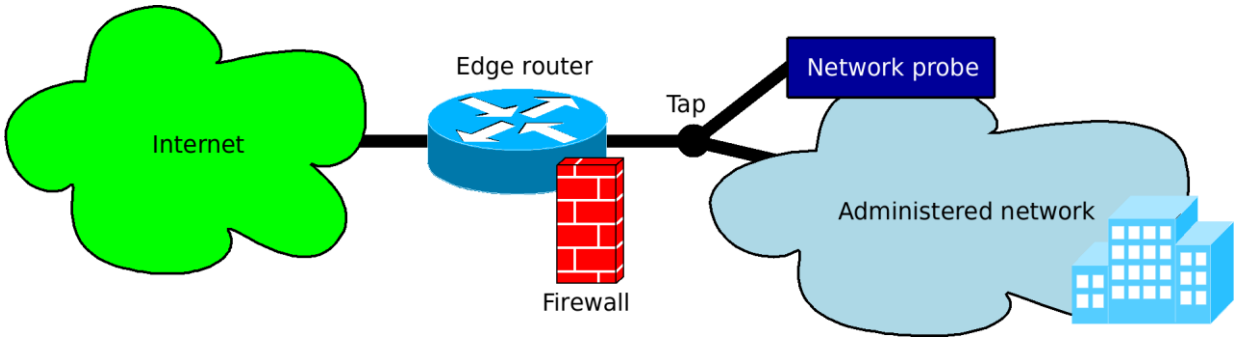


Рисунок 4.2: Розташування мережевого зонда

Для початку, на рисунку 4.3 зображено розгортання одного основного зонда та трьох всередині мережі, що адмініструється. Він може бути затребуваний в кампусі або корпоративних мережах. Існує один сегмент, який містить більш конфіденційні сервери, ніж інші, або організація достатньо велика, щоб контролювати мережевий трафік всередині організації.

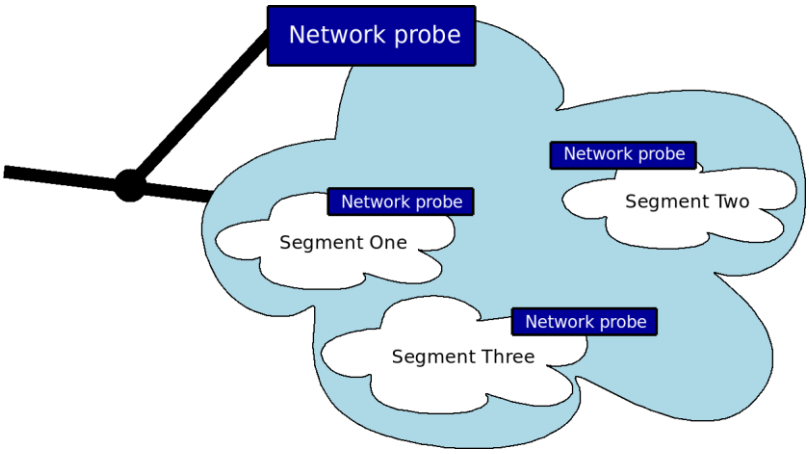


Рисунок 4.3: Зонди всередині мережі

Окрім зондів NetFlow пропонується розгорнути honeypots, щоб доповнити функціональність зондів. Це ресурс інформаційної системи, цінність якого полягає в несанкціонованому або незаконному використанні цього ресурсу. [13] Було обрано honeypot з низькою взаємодією для виконання пасивного, а не активного виявлення. Результатом honeypot повинен бути список хостів (ззовні і навіть всередині мережі), які

намагаються зв'язатися з уявними хостами в мережі, що адмініструється. Як правило, резервується кілька непризначених IP-адрес (або всієї підмережі) для honeypot. Якщо він спостерігає за спробою з'єднання з такою адресою, він реєструє хост, який ініціював з'єднання. Однак слід уникати передчасних висновків. Наприклад, для користувача, який ввів неправильну IP-адресу, неправильно налаштував хост і так далі.

Надійність безпеки дуже важлива для таких пристроїв, як мережеві зонди.

Сам зонд управляється через інтерфейс управління. При цьому використовується захищений канал (а саме SSH) і доступ надається тільки з вказаних IP-адрес. Ми використовуємо систему управління ідентифікацією, таку як RADIUS [31]. Це вигідно для розподілених систем, оскільки усуває проблеми з синхронізацією. І останнє, але не менш важливе: ми використовуємо NTP4 для синхронізації тактів обчислювальних пристроїв по мережі. Оскільки зонди позначають час потоків, використовуючи час хоста, необхідно встановити точний час.

Як правило, зонди є простими в обслуговуванні пристроями. Якщо ми помістимо їх в мережу і налаштуємо, вони будуть працювати і виконувати своє завдання. Однак, якщо вони не надсилають жодних даних колектору, ми не можемо визначити, чи відстежуване посилання або зонд вийшов з ладу. Тому ми використовуємо протокол конфігурації NETCONF поверх SSH для моніторингу стану зондування.

4.2.2 Колектори

Колектор NetFlow відповідає за правильний прийом і зберігання даних NetFlow, які експортуються мережевими зондами. При цьому використовуються існуючі інструменти та програмне забезпечення, яке добре перевірене та широко поширене. У випадку з колекторами NetFlow ми покладаємося на nf-дамп і набір інструментів NfSen. Колектори отримують і

зберігають записи NetFlow, а також виконують деякі завдання попередньої обробки, такі як періодичне виконання скриптів, які відстежують порушення політики. Колектори дотримуються вимог, описаних вище, а також інших частин запропонованого IDS.

Щоб відповідати вимогам безпеки, ми вказуємо IP-адреси зондів, які є авторизованимидля відправки даних NetFlow конкретному колектору. Слід звернути увагу, що сам колектор не обмежує прийом записів NetFlow. Це можна вважати загрозою безпеці, оскільки записи NetFlow передаються в пакетах UDP, які можна легко підробити. Якщо ми не хочемо передавати записи NetFlow через ту саму мережу, ми можемо підключити колектори безпосередньо до зондів через локальну мережу і таким чином значно посилити безпеку. Крім того, це може полегшити навантажені мережеві канали.

Хоча записи NetFlow вже агреговані (з точки зору мережевих потоків), вони займають відносно багато дискового простору. Наприклад, записи, які охоплюють один місяць мережевого трафіку великої мережі кампусу, займають близько 240 ГБ дискового простору5. Якщо ми не розгорнемо більше зондів, ми зможемо використовувати лише один колектор. Тим не менш, тривале зберігання даних вимагає достатнього місця на дисководах

Крім того, це може вимагатися якимось законом.

4.2.3 MyNetScope та джерела даних

У цьому підрозділі описується ядро системи виявлення вторгнень. Цей рівень вимагає даних від колекторів та інших джерел для своєї роботи.

Було використано платформу MyNetScope, яка була коротко описана в розділі 3.3.

Це не окрема програма, вона розроблена як клієнт-серверна архітектура. Сервер зчитує записи NetFlow від колекторів, виконує деякі завдання попередньої обробки потоків і відповідає на запити аналітика, які

надсилаються клієнтським додатком (консоллю аналітика). Знову ж таки, весь зв'язок між усіма частинами зашифрований. Ми використовуємо SSH-тунелі.

Сам CAMNEP MyNetScope не виконує виявлення вторгнень. Це дуже корисний інструмент візуалізації. Його перевага полягає в інтеграції зовнішніх джерел даних. Було вирішено розгорнути частину проекту CAMNEP як «мозок» системи виявлення вторгнень. Таким чином, можна задовольнити наступні вимоги:

- точність;
- виявлення нових загроз;
- функціонування у високошвидкісних мережах;
- раннє виявлення;
- виявлення аномалій у зашифрованому трафіку.

При цьому використовується в основному рівень кооперативного виявлення загроз CAMNEP, який поєднує в собі сучасні методи виявлення вторгнень. Таким чином, ми отримуємо кращу точність, ніж ми б застосовували методи виявлення партикулярних аномалій окремо. Методи здатні виявляти нові загрози та аномалії у випадку, якщо аномалія безпеки фіксується як аномалія мережевого трафіку. Наприклад, хробак, що розповсюджується або атакує відмову в обслуговуванні, «видно» в мережевих потоках. Навпаки, один пакет, який спричиняє переповнення буфера на хост-комп'ютері, не відображає аномалію мережевого трафіку. Далі методи були розроблені для високошвидкісних мереж з самого початку або вони були модифіковані під цю вимогу. Виявлення виконується в 5-хвилинні часові вікна. Це розумний інтервал через агрегацію потоків, яка зазвичай використовується у зв'язку з NetFlow. Нарешті, оскільки методи працюють виключно із заголовками пакетів, виявлення аномалій можливе навіть у разі зашифрованого корисного навантаження.

Рівень виявлення CAMNEP обчислює для кожного мережевого потоку його надійність. Потім це значення передається MyNetScope, і користувач

може переглядати підозрілі потоки та запитувати в MyNetScope іншу відповідну інформацію.

Також використовуються інші джерела даних, такі як DNS сервер, служба whois або спеціальні скрипти, які періодично перевіряють на предмет порушення політики. Їх вихідні дані також включаються в MyNetScope.

4.3 Розробка системи виявлення атак на мережеві ресурси з астосуванням нейромережевих технологій

4.3.1 Визначення компонентів системи

Так як для збору трафіку використовуватиметься CICFlowMeter, а його єдиним можливим висновком даних є постійно дозаписуваний csv файл, то є необхідність у створенні

інтерфейсу для його читання та видачі нових значень у зручному вигляді. Цим інтерфейсом буде служити веб-сервер, який у фоні зчитуватиме csv файл, переформатувати зчитані дані у формат json і видавати їх у веб-сторінки. Далі з веб-сторінки зчитуватимуться веб-клієнтом і передаватимуться на аналіз у нейронну мережу. За результатами аналізу вони будутьпозначені як нормальний, чи аномальний трафік. Аномальний трафік буде записано до бази даних.

Дані з бази даних формуватимуться і виводитимуться у вигляді невеликих звітів у спеціальному веб-додатку. Така структура системи виявлення атак дозволить створюватнеобмежену кількість аналізувальних модулів, що дозволить обробляти трафік у реальному часі. Аналізуючі модулі можуть бути розташовані на кількох серверах організації. Також при виході з ладу одного або декількох аналізованих модулів система продовжить свою роботу.

4.3.2 Розробка збирача даних

До початку розробки необхідно визначитись із форматом вхідних даних. Для цього запустимо програму CICFlowMeter і зробимо захоплення трафіку. Інтерфейс програми представлено на рис.4.4.

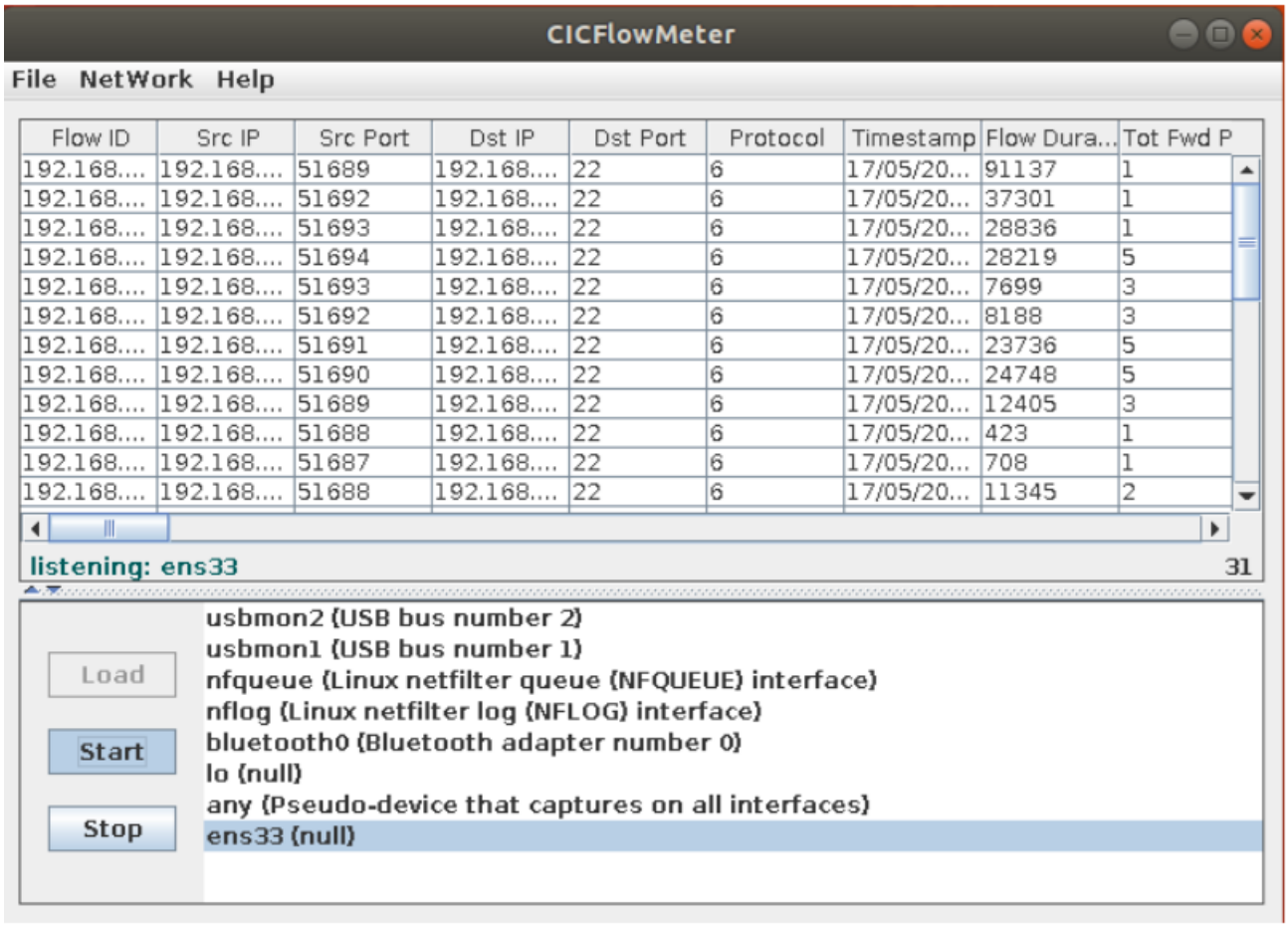


Рисунок 4.4 - Інтерфейс програми CICFlowMeter

Після початку захоплення трафіку програма створила файл, приклад полів у якому наведено в таблиці 4.1.

Таблиця 4.1 – Приклад даних, що надаються програмою CICFlowMeter

№ п/п	Назва параметра	Приклад даних
1	Flow ID	152.199.19.161-192.168.1.2-443-51472-6
2	Src IP	192.168.1.2
3	Src Port	51472
4	Dst IP	152.199.19.161
5	Dst Port	443
6	Protocol	6
7	Timestamp	17/05/2020 02:47:24 AM
8	Flow Duration	299482
9	Tot Fwd Pkts	1
10	Tot Bwd Pkts	1
11	TotLen Fwd Pkts	0
12	TotLen Bwd Pkts	0
13	Fwd Pkt Len Max	0
14	Fwd Pkt Len Min	0
15	Fwd Pkt Len Mean	0
16	Fwd Pkt Len Std	0
17	Bwd Pkt Len Max	0
18	Bwd Pkt Len Min	0
19	Bwd Pkt Len Mean	0
20	Bwd Pkt Len Std	0
21	Flow Byts/s	0
22	Flow Pkts/s	6,678198
23	Flow IAT Mean 299482	299482
24	Flow IAT Std	0
25	Flow IAT Max	299482
26	Flow IAT Max	299482

4.3.3 Опис програмних засобів

Програмний модуль реалізований за допомогою мови Python та середовища розробки PyCharm.

Додаткові бібліотеки:

- Pandas;
- Numpy;
- SciPy;
- Sklearn.

Pandas – програмна бібліотека для роботи з наборами даних, написана для мови програмування Python. Використовується для зручної роботи з наборами даних, зокрема, робота з таблицями, індексування даних, отримання зрізів даних за мітками, піднаборів з великого набору.

Numpy - програмна бібліотека, написана для мови програмування Python. Використовується для роботи з масивами, матрицями, багатовимірними масивами, операціями над ними. Дану бібліотеку часто порівнюють з MATLAB по функціоналу.

SciPy - програмна бібліотека наукових інструментів, написана для мови програмування Python. Може використовуватися для вирішення задач оптимізації, інтегрування, диференціювання, спеціальних функцій, обробки сигналів, обробки зображень тощо.

Sklearn - програмна бібліотека для машинного навчання, написана для мови програмування Python. За її допомогою можна створювати та тренувати різноманітні алгоритми нейронних мереж. Працює разом з Numpy та SciPy.

4.3.4 Опис вхідного набору даних

У даній роботі використовується набір даних NSL-KDD для навчання та

перевірки роботи нейронних мереж. Даний набір даних має ряд переваг у порівняні з базовим набором даних KDD'99 [18]:

- виключені повторювальні записи;
- рівномірний розподіл видів записів, завдяки чому можна більш ефективно навчити та оцінити нейронну мережу.
- має оптимальне співвідношення кількості записів для тренування та тестування, що дозволяє проводити одразу роботу з навчанням, без попереднього поділу на типи вибірок, пошук оптимального співвідношення тощо. Набір даних представлений декількома файлами. Опис кожного з них [18]:

- KDDTrain + .ARFF – повний комплект для тренування NSL-KDD із двійковими мітками у форматі ARFF;

- KDDTrain + txt – Повний набір для тренування NSL-KDD, що включає мітки типу атаки та рівень складності у форматі txt;

- KDDTrain + _20Percent.ARFF – 20% підмножини файлу KDDTrain + .arff;

- KDDTrain + _20Percent.TXT – 20% підмножини файлу KDDTrain + .txt;

- KDDTest + .ARFF – Повний набір тестових даних NSL-KDD із двійковими мітками у форматі ARFF;

- KDDTest + . txt – Повний набір тестових даних NSL-KDD, що включає мітки типу атаки та рівень складності у форматі txt;

- KDDTest-21.ARFF – Підмножина файлу KDDTest + .arff, яка не включає записи з рівнем складності 21 з 21;

- KDDTest-21. txt – Підмножина файлу KDDTest + .txt, яка не включає записи з рівнем складності 21 з 21.

Кожен з файлів має по 41 параметру, та 42-й параметр – це тип запису, тобто «нормальне» функціонування системи, чи відбувається атака, та в разі атаки записано саме вид атаки. [18]

У таблицях 4.2-4.4 представлено опис параметрів, що використовуються у наборі даних NSL-KDD.

Таблиця 4.2 – Базові параметри TCP з’єднання.

Назва параметру	Опис	Тип даних
duration	Час тривалості підключення	continuous
protocol_type	Протокол, який використовується при підключенні, наприклад, tcp, udp тощо	discrete
servic	Мережева служба, яка використовується підключенням	discrete
src_bytes	Кількість відправлених байт за одне з'єднання	continuous
dst_bytes	Кількість прийнятих байт за одне з'єднання	continuous
flag	Статус з'єднання - нормальне або з помилкою	discrete
land	Якщо ip-адреси хоста джерела і призначення рівні, і аналогічна ситуація з портами, то параметр приймає значення 1, інакше 0	discrete
wrong_fragment	Загальна кількість невірних фрагментів за це підключення	continuous
urgent	Кількість urgent-пакетів в цьому підключенні	continuous

Таблиця 4.3 – Функції вмісту в межах з'єднання, запропонованого даними домену

Назва параметру	Опис	Тип даних
hot	Кількість hot-індикаторів, наприклад таких як: вхід в системні директорії, створення програм, виконання програм.	continuous
num_failed_logins	Кількість невдалих спроб входу	continuous
logged_in	Логін статус. 1 - якщо успішно увійшли в систему, інакше 0	discrete
num_compromised	Число скомпрометованих станів	continuous
root_shell	1, якщо root-права отримані, інакше 0	discrete
su_attempted	1, якщо su root-права отримані, інакше 0	discrete
num_root	Число root-доступів	continuous
num_file_creations	Число операцій по створенню файлів під час з'єднання	continuous
num_shells	Число викликів shell-оболонки	continuous
num_access_files	Число операцій з отримання контролю доступу до файлів	continuous
num_outbound_cmds	Число вихідних команд в FTPсесії	continuous
is_hot_login	1, якщо логін належить hotлисту тобто якщо є root або адміністратором, інакше 0	discrete
is_guest_login	1, якщо логін є гостьовим, інакше 0	discrete

Таблиця 4.4 – Характеристики трафіку, обчислені з використанням двосекундного часового вікна.

Назва параметру	Опис
count	Кількість підключень до одного і того ж хосту призначення за останні дві секунди
serror_rate	Відсоток з'єднань з хостом з count з SYNпомилкам
rerror_rat	Відсоток з'єднань з хостом з count з REJпомилкам
same_srv_rat	Відсоток з'єднань з хостом з count використовують одні і ті ж служби
diff_srv_rate	Відсоток з'єднань з хостом використовуючи різні служби
srv_count	Число з'єднань з однієї і тієї ж службою за останні дві секунди.
srv_serror_rate	Відсоток з'єднань з SYNпомилками при з'єднанні по службі з srv_count
srv_serror_rate	Відсоток з'єднань з REJпомилками при з'єднанні по службі з srv_coun
srv_diff_host_rate	Відсоток з'єднань з різними хостами при з'єднанні по службі з srv_count
dst_host_count	Число з'єднань з тим же самим ip-адресою хоста призначення
dst_host_srv_count	Число з'єднань з тим же самим номером порту
dst_host_same_srv_rate	Відсоток з'єднань по тій же самій службі під час з'єднання з ip з dst_host_count
dst_host_diff_srv_rate	Відсоток з'єднань по різних служб під час з'єднання з ip з dst_host_count
dst_host_same_src_port_rate	Відсоток з'єднань до того ж самому хосту приймача під час зв'язок через порт з dst_host_srv_count
dst_host_srv_diff_host_rate	Відсоток з'єднань з різними хостами приймачами під час зв'язок через порт з dst_host_srv_count

Атаки діляться на чотири основні категорії:

- DOS: відмова в обслуговуванні, наприклад, syn flood;
- R2L: несанкціонований доступ з віддаленої машини, наприклад, guessing password;
- U2R: несанкціонований доступ до локальних привілеїв суперкористувача, наприклад, buffer overflow;
- probing: спостереження та інші зондування, наприклад, port scanning.

У таблиці 4.5 наведено різні типи атак, які представлено у наборі даних NSL-KDD і розподілено по категоріям.

Таблиця 4.5 – Типи атак за категоріями.

Категорія атаки	Тип атак
DoS	Back, Land, Neptune, Pod, Smurf, Teardrop
R2L	Guess_Password, Ftp_write, Imap, Phf, Multihop, Warezmaster, Warezclient, Spy
U2R	Buffer_overflow, Loadmodule, Rootkit, Perl
Probe	Satan, Ipsweep, Nmap, Portsweep

4.3.5 Програмна реалізація

Метою створення програми є побудова нейронних мереж, а також перевірка їх можливостей виявлення вторгнень до комп'ютерної системи. У результаті роботи програми відбувається навчання однієї з нейронних мереж на вибір та оцінка її роботоспроможності на тестових даних. У програмі реалізована

підготовка даних для навчання та тестування мережі, наприклад, для роботи відновлювальної нейронної мережі вимагається, аби параметри, які є категоріальними, були розділені на підмножини.

Під час реалізації нейронних мереж була використана бібліотека TensorFlow та Keras, для більш зручного моделювання мереж була використана бібліотека TensorFlow та Keras, для більш зручного моделювання мережі.

Також у програмі реалізований інтерфейс за допомогою бібліотеки Tkinter. Завдяки інтерфейсу можна обирати файл, що містить набір даних для обробки, а також обирати вид нейронної мережі та кількість нейронів на прихованих шарах мережі. Окрім цього, реалізована можливість обирати параметри, які будуть використанні при навчанні та тестуванні нейронної мережі. Загальна структура програми зображена на рис. 4.5.

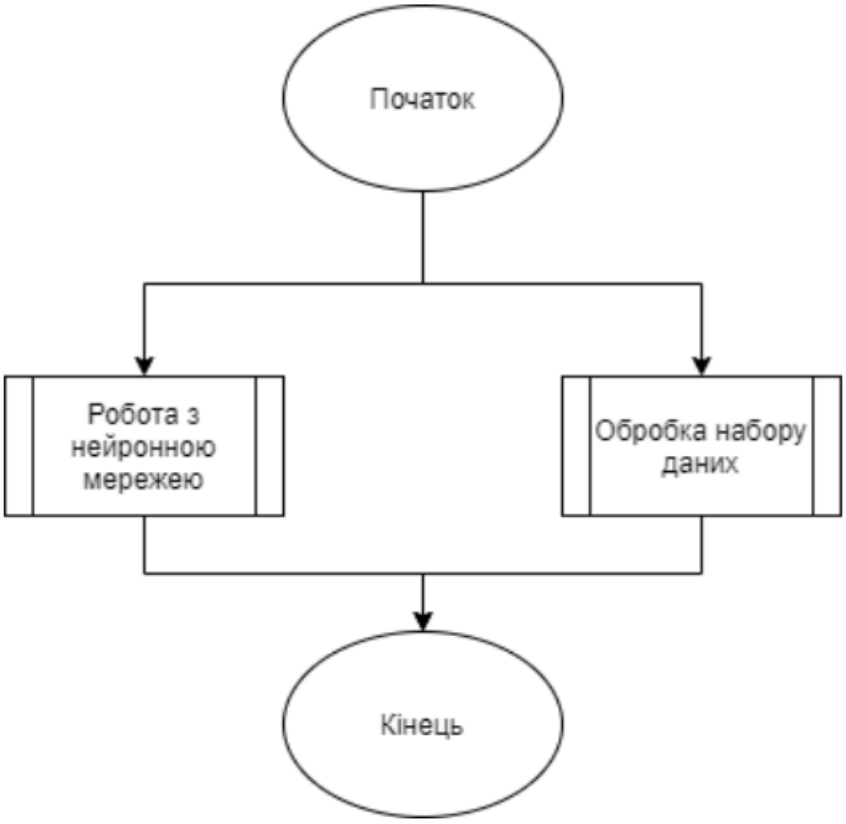


Рисунок 4.5 - Загальна структура програмного продукту

На рисунку 4.6 зображено алгоритм роботи програмного модулю для обробки набору даних, також для розділення категоріального параметру на підмножини.

Рисунок 4.6 - Алгоритм роботи програмного модулю для обробки початкового набору даних

У модулі для обробки початкового набору даних work_data_preparation() спочатку здійснюється зчитування даних з файлу, шлях до якого передається як параметр. Після цього, зчитується текст, у якому вказано, які стовбці даних необхідно залишити для подальшої обробки. Після зменшення кількості параметрів у наборі даних, відбувається логарифмування даних, аби «зтиснути» дані, що мають великі значення, оскільки нейронні мережі краще працюють, якщо дані знаходяться в проміжку близькому від 0 до 1 (допускаються відхилення, наприклад, від -1 до 1, або від 0 до 5). І одночасно з логарифмуванням, відбувається заміна текстових даних у полі «атака», на числові. На місце нормального функціонування записується 1, а в інших випадках – 0. Після цього, відбувається розподілення даних по п’яти підмножинам, в залежності від значення параметру «service»: http, smtp, ftp, ftp_data та other. А в кінці роботи модулю відбувається збереження даних до нових файлів. На рисунку 4.7 зображений загальний алгоритм роботи програмного модулю для роботи з нейронною мережею. З рисунку видно, що можна як навчати нейронну мережу, так одразу і тестувати її (якщо попередньо вона була побудована та збережена у пам’яті).

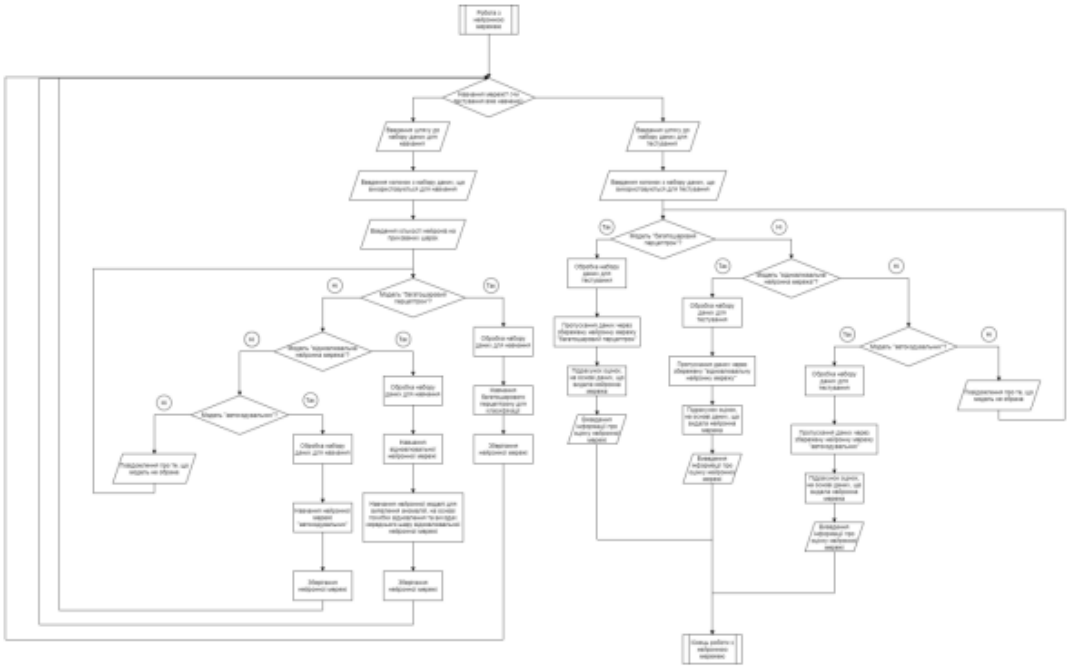


Рисунок 4.7 - Загальний алгоритм роботи програмного модулю для роботи з нейронною мережею.

Завдяки інтерфейсу, можна вибрати один із запропонованих видів нейронних мереж: «багатошаровий перцептрон», «відновлювальну нейронну мережу» або «автокодувальник». Після вибору типу мережі, можна модель навчити, або якщо модель є в пам’яті комп’ютера, то одразу протестувати її. Якщо моделі в пам’яті немає, то необхідно спочатку навчити нейронну мережу, а після чого можна протестувати.

Перевірка кожної моделі відбувається по-різному. «Багатошаровий перцептрон» встановлює чи є атака за допомогою вбудованого методу класифікації.

Для «відновлювальної нейронної мережі» будується додаткова нейронна мережа, що на вхід приймає похибку відновлення, а також значення на кожному з виходів прихованого середнього шару, а на виході додаткової мережі отримуємо результат, з якою ймовірністю відбувається нормальне функціонування. Приймаємо, що для значення, що більше 0.5 відбувається нормальне функціонування, в іншому випадку – відбувається атака.

Для «автокодувальника» вираховується похибка між початковими даними та «декодованими». Після цього необхідно встановити поріг для виявлення аномалій, тобто атак. Є декілька способів встановити цей поріг, наприклад, максимальне значення, що має похибка для нормального функціонування, або мінімальне значення похибки для даних, що містять атаку, або їх середнє арифметичне значення.

У даному програмному продукті реалізований останній алгоритм: знаходиться максимальне похибки для нормального функціонування та мінімальне значення похибки для атаки, після чого знаходимо середнє арифметичне значення і приймаємо це значення як поріг виявлення. Якщо значення похибки вище, за встановлений поріг – дані визнаємо такими, що мають атаку. Якщо похибка менше, ніж встановлений поріг – визначаємо як нормальне функціонування.

Для оцінювання якості роботи нейронної мережі були застосовані метрики, що представленні у таблиці 4.5.

Таблиця 4.5 – Метрики для оцінки нейронної мережі

Назва метрики	Формула для підрахунку або короткий опис
True positive (TP)	Правильно класифіковане нормальне функціонування
True negative (TN)	Правильно класифікована атака
False positive (FP)	Помилково класифіковане нормальне функціонування, коли була атака
False negative (FN)	Помилково класифікована атака, якої не було
True positive rate (TPR)	TP / (FN + TP
True negative rate (TNR)	TN / (FP + TN)
False negative rate (FNR)	FN / (FN + TP)
False positive rate (FPR)	FP / (FP + TN)
Positive predictive value (PPV)	TP / (TP + FP)
Negative predictive value (NPV)	TN / (TN + FN)
False discovery rate (FDF)	FP / (FP + TP)
False omission rate (FOR)	FN / (FN + TN)
Accuracy	(TP + TN) / (TP + TN + FP + FN)
Sensitivity	TP / (TP + FN)
Specificity	TN / (TN + FP

4.4 Аналіз вихідних результатів

4.4.1 Початкові дані

Набір даних NSL_KDD, зокрема файл KDDTrain+.txt, що використовується для навчання нейронних мереж містить 125973 записів, які були розподілені по файлам в залежності від значення параметру «service»: KDDTrain+_df_http.txt (40338 записів), KDDTrain+_df_smtp.txt (7313 записів), KDDTrain+_df_ftp.txt (1754 записи), KDDTrain+_df_ftp_data.txt (6859 записів), KDDTrain+_df_other.txt (69708 записів). Для навчання нейронних мереж було використано наступні параметри: duration, service, src_bytes, dst_bytes.

4.4.2 Результати роботи програми

На рисунках 4.8 – 4.10 зображено приклад виведення метрик якості роботи «відновлювальної нейронної мережі» для даних, що містять параметр «service» - «http», кількість нейронів на прихованих шарах – 35, 3, 35.

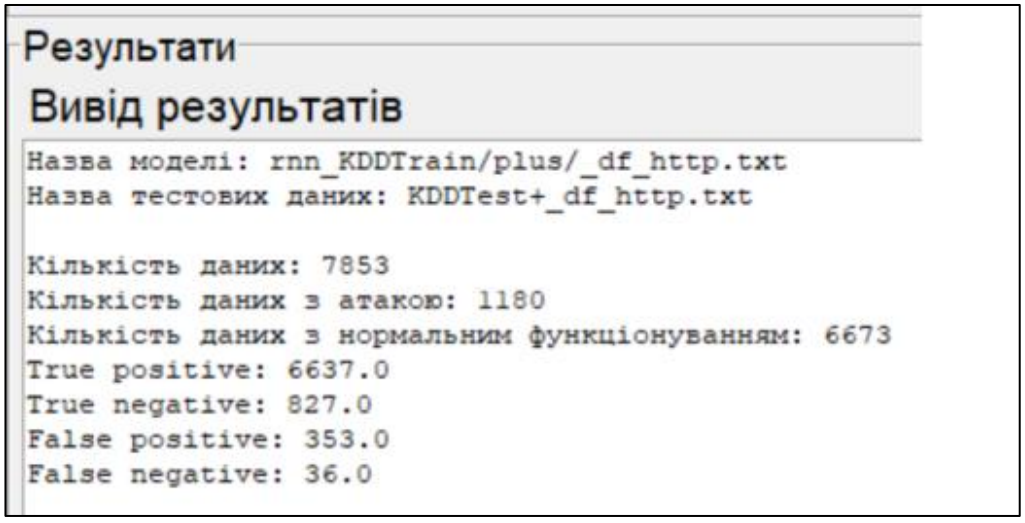


Рисунок 4.8 - Результат тестування «відновлювальної нейронної мережі» (частина 1)

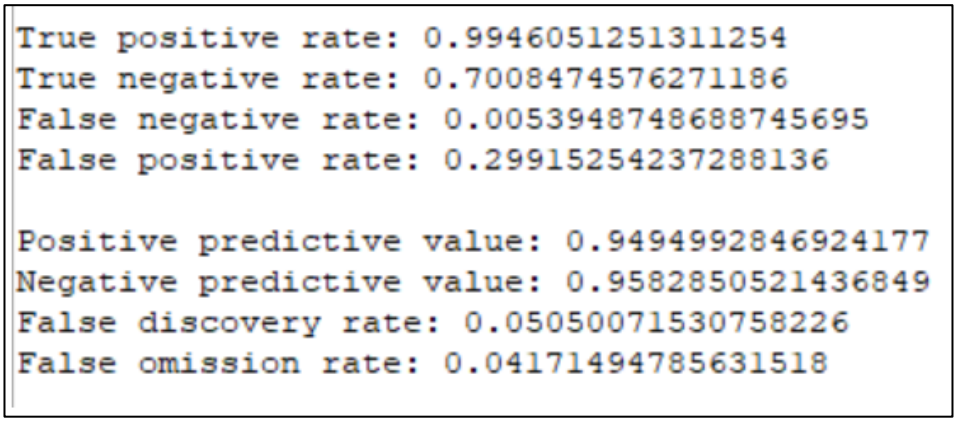


Рисунок 4.9 - Результат тестування «відновлювальної нейронної мережі» (частина 2)

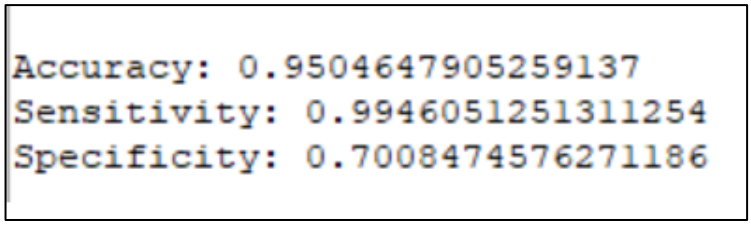


Рисунок 4.10 - Результат тестування «відновлювальної нейронної мережі» (частина 3)

У таблицях 4.6-4.8 зображено порівняння оцінок, в залежності від типу моделі нейронної мережі.

Таблиця 4.6 – Метрики якості роботи нейронної мережі «багатошаровий перцептрон»

Назва метрики	Багатошаровий перцептрон
1	2
Дані з атакою	12832
Дані з нормальним функціонуванням	9711
True positive (TP)	9700
True negative (TN)	3

Закінчення таблиці 4.6

1	2
False positive (FP)	12829
False negative (FN)	11
True positive rate (TPR)	0,998867
True negative rate (TNR)	0,000234
False negative rate (FNR)	0,001133
False positive rate (FPR)	0,999766
Positive predictive value (PPV)	0,430556
Negative predictive value (NPV)	0,214286
False discovery rate (FDF)	0,569444
False omission rate (FOR)	0,785714
Accuracy	0,430422
Sensitivity	0,998867
Specificity	0,000234

З таблиці 4.6 видно, що нейронна мережа «багатошаровий перцептрон» має досить низьку точність виявлення загроз, тому в подальшому аналізі вона не розглядається.

В таблиці 4.7 наведено метрики якості роботи «відновлювальної нейронної мережі».

Таблиця 4.7 – Метрики якості роботи «відновлювальної нейронної мережі»

Назва метрики	http	smtp	ftp	ftp_data	other
Дані з атакою	1180	316	644	531	10161
Дані з нормальним функціонуванням	6673	618	48	320	2052
True positive (TP)	6644	606	35	312	1723
True negative (TN)	840	16	624	293	7190
False positive (FP)	340	300	20	238	2971
False negative (FN)	29	12	13	8	329
True positive rate (TPR)	0,995654	0,980583	0,729167	0,975	0,839669
True negative rate (TNR)	0,711864	0,050633	0,968944	0,551789	0,707608
False negative rate (FNR)	0,004346	0,019417	0,270833	0,025	0,160331
False positive rate (FPR)	0,288136	0,949367	0,031056	0,448211	0,292392
Positive predictive value (PPV)	0,951317	0,668874	0,636364	0,567273	0,367064
Negative predictive value (NPV)	0,966628	0,571429	0,979592	0,973422	0,956244
False discovery rate (FDF)	0,048683	0,331126	0,363636	0,432727	0,632936
False omission rate (FOR)	0,033372	0,428571	0,020408	0,026578	0,043756
Accuracy	0,953012	0,665953	0,952312	0,710928	0,729796
Sensitivity	0,995654	0,980583	0,729167	0,975	0,839669
Specificity	0,711864	0,050633	0,968944	0,551789	0,707608

Таблиця 4.8 - Метрики якості роботи нейронної мережі «автокодувальник»

Назва метрики	http	smtp	ftp	ftp_data	other
Дані з атакою	1180	316	644	531	10161
Дані з нормальним функціонуванням	6673	618	48	320	2052
True positive (TP)	6661	610	46	317	2021
True negative (TN)	793	2	629	228	6
False positive (FP)	387	314	15	303	10155
False negative (FN)	12	8	2	3	31
True positive rate (TPR)	0,998202	0,987055	0,958333	0,990625	0,984893
True negative rate (TNR)	0,672034	0,006329	0,976708	0,429379	0,00059
False negative rate (FNR)	0,001798	0,012945	0,041667	0,009375	0,015107
False positive rate (FPR)	0,327966	0,993671	0,023292	0,570621	0,99941
Positive predictive value (PPV)	0,945091	0,660173	0,754098	0,51129	0,165982
Negative predictive value (NPV)	0,985093	0,2	0,99683	0,987013	0,162162
False discovery rate (FDF)	0,054909	0,339827	0,245902	0,48871	0,834018
False omission rate (FOR)	0,014907	0,8	0,00317	0,012987	0,837838
Accuracy	0,949191	0,655246	0,975434	0,640423	0,165971
Sensitivity	0,998202	0,987055	0,958333	0,990625	0,984893

Specificity	0,672034	0,006329	0,976708	0,429379	0,00059

Використовуючи дані з таблиць 4.7 та 4.8 побудуємо оцінку для кожної з мереж, просумувавши перші 6 рядків між собою. Інші оцінки вирахуємо за формулами, що були описані вище. Отримані дані зобразимо в таблиці 4.9.

Таблиця 4.9 - Метрики якості роботи нейронної мережі «автокодувальник»

Назва метрики	Відтворювальна нейронна мережа	Нейронна мережа «автокодувальник»
Дані з атакою	12832	12832
Дані з нормальним функціонуванням	9711	9711
True positive (TP)	9320	9655
True negative (TN)	8963	1658
False positive (FP)	3869	11174
False negative (FN)	391	56
True positive rate (TPR)	0,959736	0,994233
True negative rate (TNR)	0,698488	0,129208
False negative rate (FNR)	0,040264	0,005767
False positive rate (FPR)	0,301512	0,870792
Positive predictive value (PPV)	0,706649	0,463536
Negative predictive value (NPV)	0,9582	0,967328
False discovery rate (FDF)	0,293351	0,536464
False omission rate (FOR)	0,0418	0,032672
Accuracy	0,811028	0,501841
Sensitivity	0,959736	0,994233

Specificity	0,698488	0,129208

На рисунку 4.11 зображено гістограму, що побудована на основі таблиць 4.6 та 4.7.

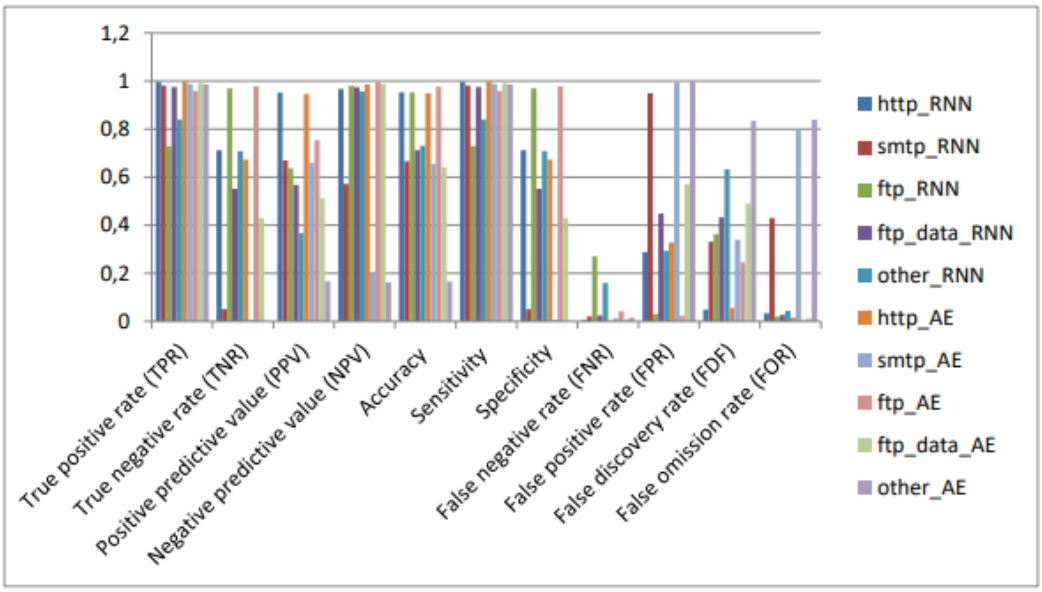


Рисунок 4.11 - Гістограма порівняння «відтворювальної нейронної мережі» та «автокодувальника» за типами мережевої служби, яка використовується підключенням

На рисунку 4.11 позначено «відтворювальну нейронну мережу» як «RNN», а «автокодувальник» як «AE».

На рисунку 4.12 зображено гістограму, що побудована на основі таблиці 4.8.

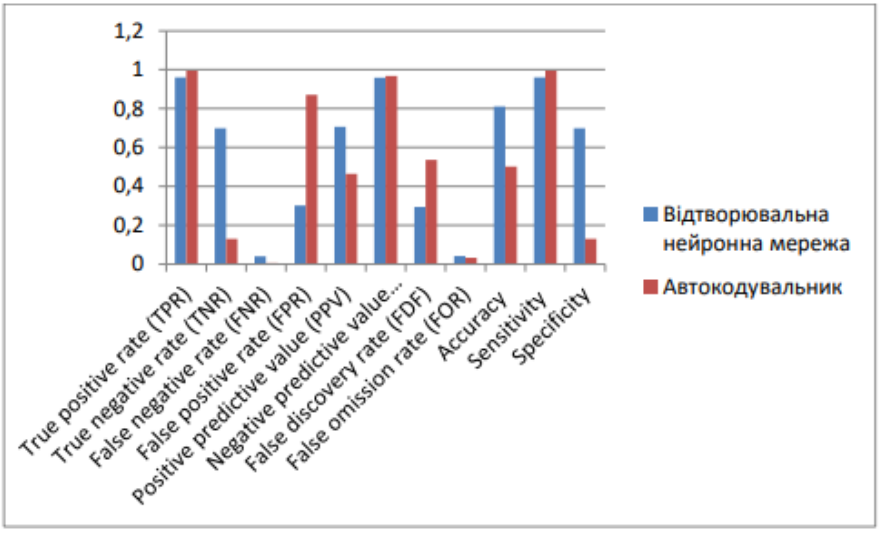


Рисунок 4.12 - Гістограма для порівняння відтворювальної нейронної мережі та автокодувальника.

З рисунку 4.11 видно, що серед «відтворювальної нейронної мережі» та нейронної мережі «автокодувальник» - найкращі показники точності «Accuracy» мають підмножини http та ftp. А в загальному плані, «відтворювальна нейронна мережа» показала кращу точність та вище значення «True negative rate», що у випадку виявлення загроз має важливе значення, оскільки це показник того, наскільки гарно нейронна мережа виявляє саме атаки.

Отже, в цьому розділі були реалізовані наступні моделі: багатошаровий перцептрон, «відтворювальна нейронна мережа» та нейронна мережа «автокодувальник». Був розглянутий початковий набір даних, а також результати роботи програмного продукту. Також був проведений аналіз оцінок якості роботи нейронних мереж.

В результаті аналізу було виявлено, що «відтворювальна нейронна мережа» має більшу точність, ніж інші нейронні мережі серед побудованих Набір даних для навчання був взятий NSL-KDD і на основі цих даних, найкращі результати з розпізнання атаки на комп'ютерну систему показала «відтворювальна нейронна мережа».

Дану систему розпізнання загроз можна застосовувати на реальних комп'ютерних системах, оскільки дані що приймаються на вхід до нейронної мережі є легкодоступними для моніторингу. Для практичного застосування варто

лише змінити вихідні дані, тобто на вихід отримувати актуальну інформацію про стан комп’ютерної системи та у разі атаки приймати відповідні дії для забезпечення безпеки системи.

ВИСНОВКИ

Метою цієї магістерської роботи є дослідження засобів оптимізації захисту комп’ютерних мереж від вторгнень, для чого було здійснено аналіз сучасного стану методів виявлення та запобігання вторгненням.

В першому розділі магістерської роботі розглянуто технології безпеки комп'ютерних мереж, такі як системи виявлення вторгнень та системи запобігання вторгненням. Аналіз систем виявлення вторгнень дозволяє зазначити, що вони розподіляються на два основні класи в залежності від їхнього положення у мережі:

- хостові системи виявлення вторгнень (HIDS);
- мережеві системи виявлення вторгнень (NIDS.

Системи запобігання вторгненням можна розділити на чотири класи:

- мережеві системи запобігання вторгненням;
- системи запобігання вторгненням у безпроводову мережу (WIPS);
- системи запобігання вторгненням на основі мережевої поведінки;
- хостові системи запобігання вторгненням.

У порівнянні з системою виявлення вторгнень система запобігання вторгненням є реактивною системою, в якій система виявлення вторгнень тісно пов'язана із брандмауером (і має бути частиною каналу зв'язку).

Під час дослідження було проаналізовано методи виявлення вторгнень і визначено, що основними методами виявлення вторгнень є наступні:

- метод виявлення вторгнень на основі сигнатур;
- метод виявлення вторгнень на основі аналізу протоколу з підтримкою стану;
- метод виявлення на основі аномалій
- метод Холта-Уінтерса;
- метод виявлення вторгнень на основі міннесотської системи виявлення вторгнень (MINDS).

При цьому було зосереджено увагу на сучасних методах, які працюють на рівні IP, оскільки вони ефективні у високошвидкісних гігабітних мережах.

Навпаки, аналіз протоколу зі збереженням стану або виявлення на основі сигнатур, що виконуються на вищих рівнях моделі TCP/IP, є завданнями, які потребують ресурсів. Отже, деякі статистичні методи перевіряють не весь пакет, а лише його заголовки. Вони працюють з даними NetFlow, отриманими від маршрутизаторів (зазвичай від пристроїв Cisco) або трасами пакетів, які пізніше «перетворюються» в мережеві потоки. Хоча ці методи працюють лише із заголовками пакетів, вони здатні виявити деякі аномалії в поведінці мережі.

Далі було визначено основні вимоги до системи виявлення вторгнень. Потім розроблено розподілену систему, яка відповідає цим вимогам. Система виявлення вторгнень складається з кількох різних компонентів. При цьому було об'єднано деякі існуючі підсистеми та розроблено інтеграційну платформу. Під час розробки інтеграційної платформи було використанои зонди NetFlow з апаратним прискоренням, honeypots, колектори NetFlow, платформу MyNetScope та інші джерела даних, такі як DNS, whois та вихідні дані інших сценаріїв, які (попередньо) обробляють отримані дані.

Наукова новизна даної роботи полягає в тому, що на основі цієї роботи може бути розроблено новий метод виявлення атак, оснований на нових напрямках збору даних. Зокрема, використання формату IPFIX отримало б доступ до цікавих функцій у корисному навантаженні пакетів для методів виявлення аномалій.

Також цінною була б більш тісна інтеграція інших джерел даних, таких як honeypots.

Крім того, в магістерській роботі розроблено програмний модуль системи виявлення атак на мережеві ресурси з застосуванням нейромережевих технологій. Були реалізовані наступні моделі: багатошаровий перцептрон, «відтворювальна нейронна мережа» та нейронна мережа «автокодувальник». Був розглянутий початковий набір даних, а також результати роботи програмного продукту. Також був проведений аналіз оцінок якості роботи нейронних мереж.

В результаті аналізу було виявлено, що «відтворювальна нейронна мережа» має більшу точність, ніж інші нейронні мережі серед побудованих Набір даних для навчання був взятий NSL-KDD і на основі цих даних, найкращі результати з

розпізнання атаки на комп’ютерну систему показала «відтворювальна нейронна мережа».

Дану систему розпізнання загроз можна застосовувати на реальних комп’ютерних системах, оскільки дані, що приймаються на вхід до нейронної мережі є легкодоступними для моніторингу. Для практичного застосування варто лише змінити вихідні дані, тобто на вихід отримувати актуальну інформацію про стан комп’ютерної системи та у разі атаки приймати відповідні дії для забезпечення безпеки системи.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Northcutt, S. and Frederick, K. and Winters, S. and Zeltser, L. and Ritchey, R.: Inside Network Perimeter Security: Definitive Guide для Firewalls, VPNs, Routers, and Intrusion Detection Systems , New Rider's Publishing, 2013, 978-0735712324. 2.1

2. Paxson, V.: Bro: A System for Detecting Network Intruders in Real-Time , 2020, < http://www.icir.org/vern/papers/bro-CN99.html > .

3. Brutlag, J.: Aberrant behaviour Detection in Time Series for Network Monitoring , 2018, <http://www.usenix.org/events/lisa00/full_papers/brutlag/ brutlag_html/index.html > .

4. Rehák, M. and P e ˇ chou c ˇ ek, M. and Bartoš, K. and Grill, M. and C ˇ eleda, P. _ and Krmí c ˇ ek, V.: CAMNEP: An intrusion detection system for high-speed networks , 2018, < http://www.nii.ac.jp/pi/n5/5_65.pdf >

5. Rehák, M. and P e ˇ chou c ˇ ek, M. and C ˇ eleda, P. _ and Krmí c ˇ ek, V. _ and Novotný, J. and Minark , P.: CAMNEP: Agent-Based Network Intrusion Detection System (Short Paper) , 2018.

6. Lau, S.: The Spinning Cube of Potential Doom , 2014.

7. Senie, D. і Sullivan, A.: Консультації для використання DNS Re- verse Mapping, 2018, < http://www.ietf.org/internet-drafts/ draft-ietf-dnsop-reverse-mapping-considerations-06.txt > .

8. Graham, I.: Achieving Zero-loss Multi-gigabit IDS Results from Testing Snort on Endace Accelerated Multi-CPU Platforms , 2016, < http://www.touchbriefings. com/pdf/2259/graham.pdf > .

9. Oberheide, J. and Goff, M. and Karir, M.: Flamingo: Visualizing Internet Traffic , 2016.

10. C ˇ eleda, P. _ and Ková c ˇ ik, M. and Koní r ˇ , T. _ and Krmí c ˇ ek, V. _ and Žádník, M.: CESNET technical report number 31/2021: FlowMon Probe , 2021, < http://www.cesnet. cz/doc/techzpravy/2006/flowmon-probe/flowmon-probe.pdf > .

11. Malagon, C. and Molina, M. and Schuurman, J.: Deliverable DJ2.2.4: Findings of the Advanced Anomaly Detection Pilot , 6. 9. 2017, < http://www.geant2.

net/upload/pdf/GN2-07-218v2-DJ2-2-4_Findings_of_the_Advanced_ Anomaly_Detetion_Pilot.pdf > .

12. Scarfone, K. and Mell, P.: Guide to Intrusion Detection and Prevention Systems (IDPS) , 2017, <_http://csrc.nist.gov/publications/nistpubs/800-94/.

13. Spitzner, L.: Honeypots , 2023, < http://www.tracking-hackers.com/papers/ honeypots.html >

14. Chatfield, C. and Yar, M.: Holt-Winters Forecasting: Some Practical Issues , 2018.

15. Brauckhoff, D. and Tellenbach, B. and Wagner, A. and Lakhina, A. and May, M.: Impact of Packet Sampling on Anomaly Detection Metrics , 2018, < http: //cs-people.bu.edu/anukool/pubs/anomalymetrics-sampling-imc06. pdf > .

16. van Riel, J. and Irwin, B.: InetVis, a visual tool for network telescope traffic analysis , 2016, < http://www.cs.ru.ac.za/research/g02v2468/publications/ vanRiel-Afrigraph2026.pdf > .

17. Brockwell, P. i Davis, R.: Introduction to Time Series and Forecasting, Second Edi- tion , 2022, Springer-Verlag New York, Inc., 0-387-95351-5. 2.6.1

18. Zhang, X. and Li, C. and Zheng, W.: Intrusion Prevention System Design , 2014.

19. Deering, S. and Hinden, R.: RFC 2460: Internet Protocol, Version 6 (IPv6) 2018, < http://www.ietf.org/rfc/rfc2460.txt > .

20. Ivanko, J.: One-way throughput Test - 20070715-F-0001 , 2022, < http://www. liberouter.org/flowmon/reports/report-20070715-F-0001.pdf > .

21. Kiss, G.: NfSen-HW project site , 2016, http://bakacsin.ki.iif.hu/~kissg/project/nfsen-hw <http://bakacsin.ki.iif.hu/~kissg/project/nfsen-hw/ > .

22. Lakhina, A. and Crovella, M. and Diot, C.: Mining Anomalies Using Traf- fic Feature Distributions _ 2015, < http://cs-people.bu.edu/anukool/pubs/ sigc05-mining-anomalies.pdf > .

23. Lakhina, A. and Crovella, M. and Diot, C.: Diagnosing Network-Wide Traffic Anomalies , 2014, < http://cs-people.bu.edu/anukool/pubs/ subspacemethod-sigc04.pdf > .

24. Lakhina, A. and Papagiannaki, K. and Crovella, M. and Diot, C. and Kolaczyk, E. and Taft, N.: Structural Analysis of Network Traffic Flows , 2014, < http://cs-people.bu.edu/anukool/pubs/odflows-sigm04.pdf > .

ДЕМОНСТРАЦІЙНІ МАТЕРІАЛИ

ДЕРЖАВНИЙ УНІВЕРСИТЕТ

ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ
НАВЧАЛЬНО–НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

МАГІСТЕРСЬКА РОБОТА

«Дослідження засобів організації безпеки комп'ютерних мереж з метою оптимізації захисту від вторгнень»

виконав студент: **Кувік Н.І.**
керівник: **Лемешко А.В.,** доктор філософії, доцент

1

Мета магістерської роботи:

дослідження та розробка оптимальних стратегій організації безпеки комп'ютерних мереж з акцентом на оптимізацію захисту від потенційних вторгнень

Об’єкт дослідження:

засоби організації безпеки комп’ютерних мереж

Предмет дослідження:

комп’ютерні мережі

2

Актуальність:

Зростаюча кількість кіберзагроз та вдосконалення методів вторгнень ставлять під сумнів ефективність існуючих засобів безпеки. Оптимізація захисту від вторгнень у комп'ютерних мережах є важливим завданням для забезпечення конфіденційності, цілісності та доступності інформації.

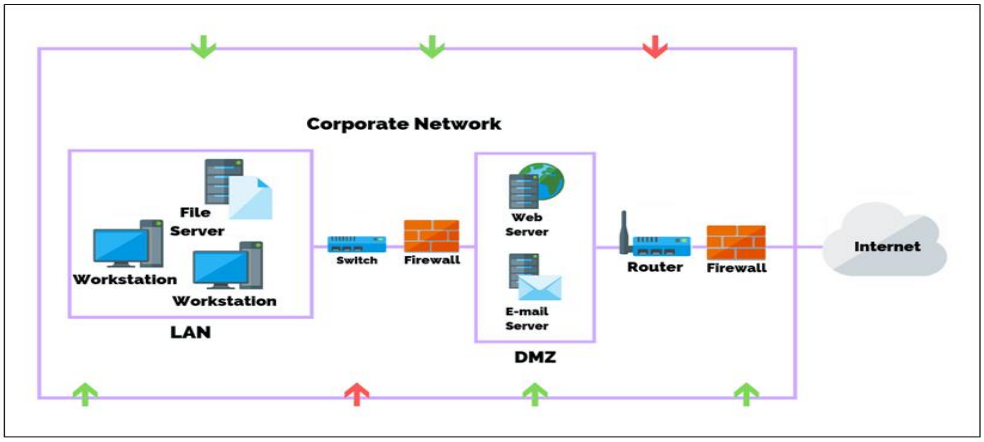
3

Наукова новизна:

Наукова новизна даної роботи полягає в тому, що на основі магістерської роботи може бути розроблено новий метод виявлення атак, оснований на нових напрямках збору даних. Зокрема, використання формату IPFIX отримало б доступ до цікавих функцій у корисному навантаженні пакетів для методів виявлення аномалій.

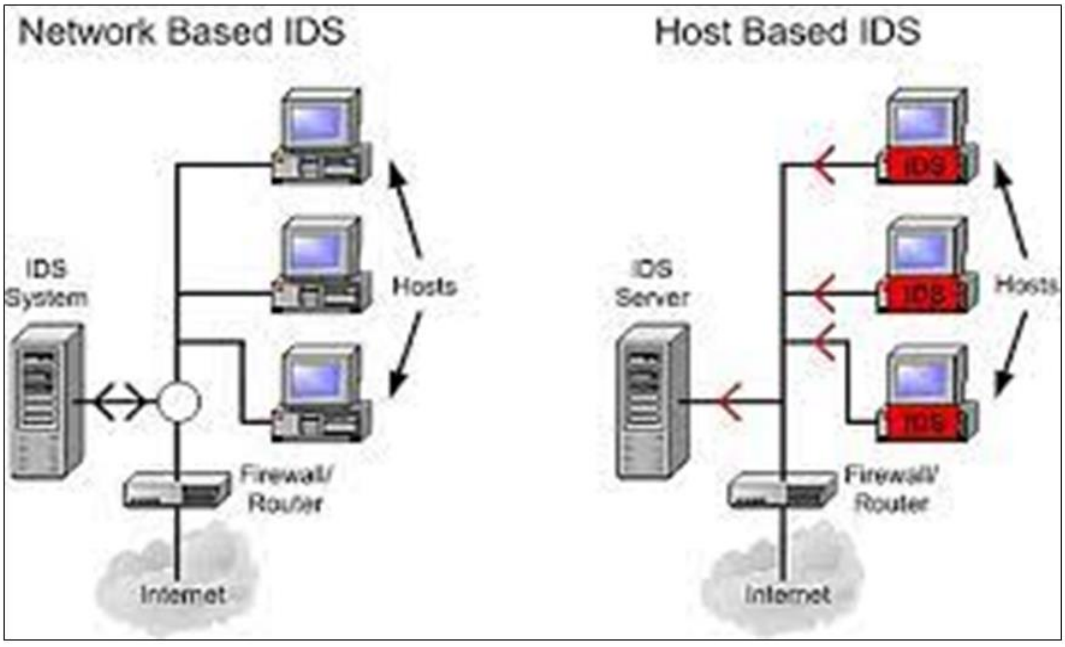
4

Точки комп'ютерної мережі, вразливі для вторгнень за векторами загроз



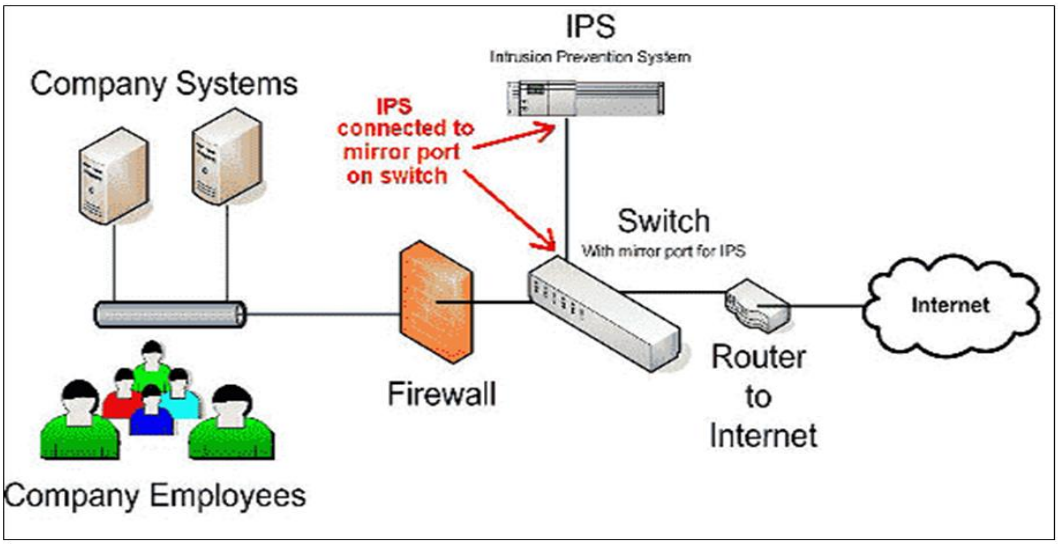
5

Класи системи виявлення вторгнень (IDS)



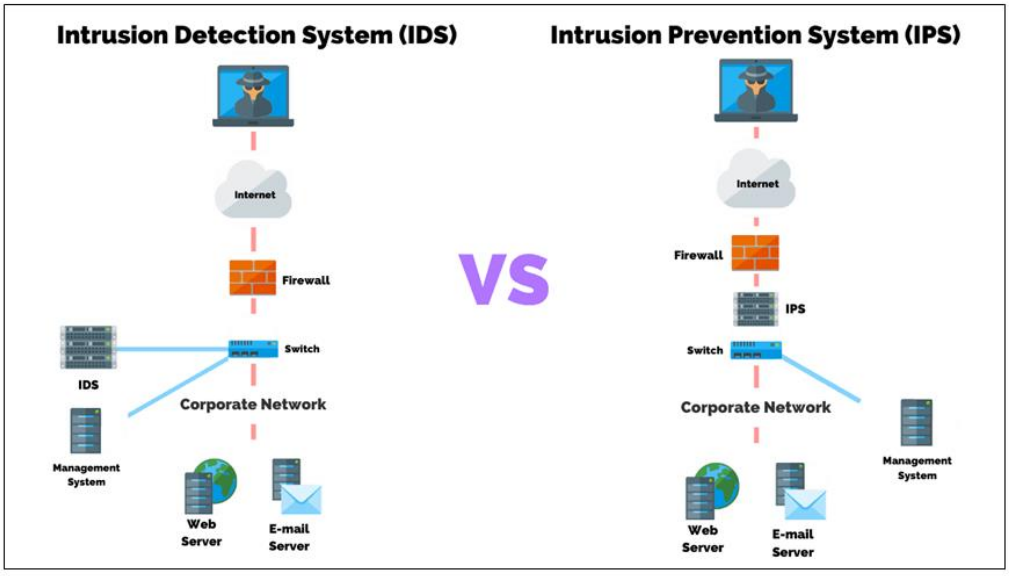
6

- Схема функціонування системи запобігання вторгненням (IPS)



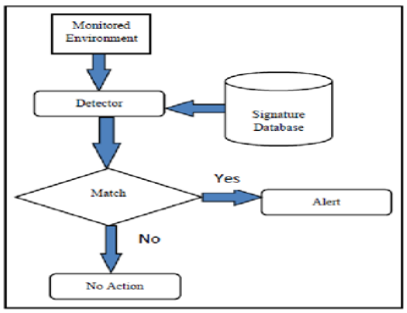
7

Відмінності між функціонуванням системи виявлення вторгнень (IDS) і системи запобігання вторгненням (IPS)

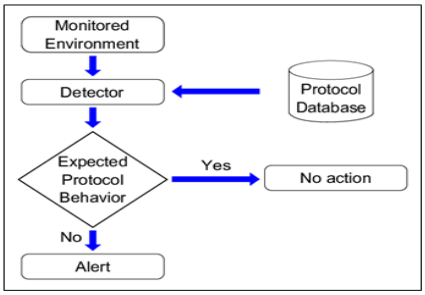


8

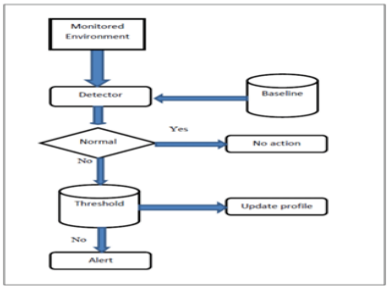
Методи виявлення вторгнень



Архітектура методології виявлення вторгнень на основі сигнатур



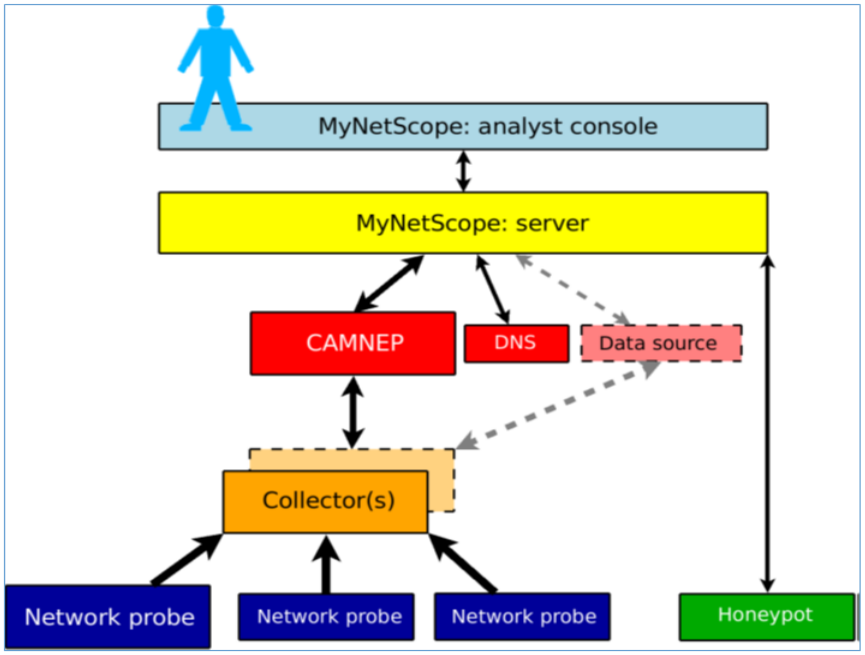
Архітектура методології виявлення вторгнень на основі аналізу протоколу з підтримкою стану



Архітектуру методології виявлення вторгнень на основі аномалій

9

Архітектура запропонованої системи виявлення вторгнень



10

Загальна структура програмного продукту

11

Висновки

Аналіз систем виявлення вторгнень дозволяє зазначити, що вони розподіляються на два основні класи в залежності від їхнього положення у мережі:

- хостові системи виявлення вторгнень (HIDS);
- мережеві системи виявлення вторгнень (NIDS).

Системи запобігання вторгненням можна розділити на чотири класи:

- мережеві системи запобігання вторгненням;
- системи запобігання вторгненням у безпроводову мережу (WIPS);
- системи запобігання вторгненням на основі мережевої поведінки;
- хостові системи запобігання вторгненням.

У порівнянні з системою виявлення вторгнень система запобігання вторгненням є реактивною системою, в якій система виявлення вторгнень тісно пов'язана із брандмауером (і має бути частиною каналу зв'язку).

Під час дослідження було проаналізовано методи виявлення вторгнень і визначено, що основними методами виявлення вторгнень є наступні:

- метод виявлення вторгнень на основі сигнатур;
- метод виявлення вторгнень на основі аналізу протоколу з підтримкою стану;
- метод виявлення на основі аномалій
- метод Холта-Уінтерса;
- метод виявлення вторгнень на основі міннесотської системи виявлення вторгнень (MINDS).

12

Висновки

При цьому було об'єднано деякі існуючі підсистеми та розроблено інтеграційну платформу. Під час розробки інтеграційної платформи було використанои зонди NetFlow з апаратним прискоренням, honeypots, колектори NetFlow, платформу MyNetScope та інші джерела даних, такі як DNS, whois та вихідні дані інших сценаріїв, які (попередньо) обробляють отримані дані.

Крім того, в магістерській роботі розроблено програмний модуль системи виявлення атак на мережеві ресурси з застосуванням нейромережевих технологій. Були реалізовані наступні моделі: багатошаровий перцептрон, «відтворювальна нейронна мережа» та нейронна мережа «автокодувальник». Був розглянутий початковий набір даних, а також результати роботи програмного продукту. Також був проведений аналіз оцінок якості роботи нейронних мереж.

В результаті аналізу було виявлено, що «відтворювальна нейронна мережа» має більшу точність, ніж інші нейронні мережі серед побудованих Набір даних для навчання був взятий NSL-KDD і на основі цих даних, найкращі результати з розпізнання атаки на комп'ютерну систему показала «відтворювальна нейронна мережа».

Дану систему розпізнання загроз можна застосовувати на реальних комп'ютерних системах, оскільки дані, що приймаються на вхід до нейронної мережі є легкодоступними для моніторингу. Для практичного застосування варто лише змінити вихідні дані, тобто на вихід отримувати актуальну інформацію про стан комп'ютерної системи та у разі атаки приймати відповідні дії для забезпечення безпеки системи.

13

ДЯКУЮ ЗА УВАГУ!

14