МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ АВІАЦІЙНИЙ УНІВЕРСИТЕТ
КАФЕДРА КОМП'ЮТЕРИЗОВАНИХ СИСТЕМ ЗАХИСТУ ІНФОРМАЦІЇ

ДОПУСТИТИ ДО ЗАХИСТУ
Завідувач кафедри

_____ С.В. Казміруук

«_____»_____20__ р.

На правах рукопису
УДК 004.056.5:510.22(043.3)

МАГІСТЕРСЬКА АТЕСТАЦІЙНА РОБОТА

ВИПУСКНИКА ОСВІТНЬОГО СТУПЕНЯ
«МАГІСТР»

Тема: Програмний модуль моніторингу Інтернет-трафіку користувачів

Автор: А.Р.Клюцко

Науковий керівник: к.т.н., доц. С.С. Ільєнко

Нормоконтролер: к.т.н., доц. С.С. Ільєнко

Київ 2020

ВСТУП

Актуальність. В IT-інфраструктурі сучасної компанії щодня генеруються великі обсяги мережевого трафіку. Відстежувати вразливі місця в мережевих взаємодіях між пристроями та користувачами в міру розростання інфраструктури і впровадження нових технологій стає складніше. У свою чергу, кіберзлочинці володіють цілим арсеналом технік для приховування своєї присутності в скомпрометованої інфраструктурі і маскування генерується шкідливого трафіку під легітимний. Інформації про мережеві адреси, порти і протоколи, за якими встановлюються з'єднання, вже недостатньо для своєчасного виявлення загроз і реагування на них. Необхідний глибокий аналіз трафіку - з розбором протоколів до рівня додатків включно(L7). З цим завданням успішно справляються рішення класу network traffic analysis (NTA).

У NTA-системах застосовуються алгоритми машинного навчання, поведінковий аналіз, правила детектування, а також є можливість ретроспективного аналізу. Це дозволяє виявляти підозрілу мережеву активність, активність шкідливого ПО, спроби експлуатації вразливостей як на периметрі, так і всередині мережі, порушення регламентів ІБ і інші загрози. Аналітичне агентство Gartner відзначає, що ряд організацій використовують NTA-рішення в складі центрів моніторингу і реагування на загрози інформаційної безпеки (security operations centers, SOC) поряд з рішеннями для захисту кінцевих користувачів і SIEM-системами [1].

Забезпечення кібербезпеки не повинно обмежуватися периметром і традиційними засобами захисту. Як показали результати дослідження, 92% загроз виявляються тоді, коли ворог вже всередині [2].

Кібер-угрупування долають захист периметру організацій, і про це свідчить тенденція до зростання частки успішних цільових атак. Це привід змістити фокус уваги з запобігання атак на периметрі на своєчасне виявлення

компрометації і реагування всередині мережі. Однак виявити ретельно сплановану кібератаку складно. Зловмисників більше не зупиняють антивіруси: вони регулярно модифікують вихідні коди шкідливого ПО, застосовують безтілесні техніки, експлуатують вразливості нульового дня. Динамічний аналіз теж не панацея: зловмисники навчилися виявляти технології віртуалізації, які зазвичай застосовують в пісочницях. Зрештою, не можна виключати, що зловмисник зможе обійтися всередині мережі взагалі без шкідливого ПЗ, обмежуючись тими інструментами, використання яких дозволено політиками безпеки.

Проте дії зловмисників залишають сліди в мережевому трафіку, а значить, завдання фахівця з кібербезпеки - виявити ці сліди.

Окрім цього, за умовами масового переходу на віддалену роботу під час епідемії SARS Covid-19, дуже важливо посилити контроль за трафіком користувачів. Чимало людей працює на власному обладнанні з-за неможливості навіть великих компаній надати потрібну техніку абсолютно усім працівникам. Технології віддаленого робочого не завжди коректно контролюють трафік користувачів, тому з боку компанії є потреба в створенні спеціалізованого ПЗ або у використанні вже готових рішень.

Метою роботи є розробка програмного модулю моніторингу Інтернет-трафіку користувачів

Виходячи з мети, завданням даної дипломної роботи є:

1. дослідити відомі системи та програмні рішення моніторингу мережного трафіку;

2. розробити алгоритм та програмну реалізацію модулю моніторингу Інтернет-трафіку користувачів на базі нейромережевого методу;

3. провести тестування та оцінку доцільності використання розробленого програмного модулю моніторингу Інтернет-трафіку користувачів.

Галузь застосування. Розроблений програмний модуль відносяться до

галузі інформаційної безпеки і може бути використаним для підвищення рівня захищеності мережі за рахунок нейромережевого методу обробки пакетів та балансування навантаження.

Об'єктом дослідження є процеси обробки, зберігання та передачі даних по відкритій мережі інтернет

Предметом дослідження є методи виявлення атак та системи моніторингу мережного трафіку.

Методи досліджень. Проведені дослідження базуються на сучасних методах та засобах виявлення мережевих атак і моніторингу Інтернет-трафіку користувачів.

Наукова новизна. Удосконалено систему моніторингу Інтернет-трафіку користувачів, за рахунок використання нейромережевого методу та балансування навантаження, що дозволило підвищити швидкість оброки трафіку користувачів зі збереженням цілісності потоків пакетів.

Практична цінність одержаних результатів дослідження полягає в тому, що розроблено програмний модуль моніторингу Інтернет-трафіку користувачів на базі використання балансування навантаження за допомогою якого контрольований трафік розподіляється по вихідним портам 10G з використанням обєктно-орієнтованої мови програмування C#, python 2.7 та json, який дозволяє забезпечити захисту інформації при її передачі в інформаційних мережах.

РОЗДІЛ 1. СУЧАСНІ ПІДХОДИ ЩОДО ОРГАНІЗАЦІЇ ЗАХИСТУ ІНФОРМАЦІЇ

1.1. Організаційно-правові основи забезпечення захисту інформації

Простота, велика кількість способів доступу та модифікації інформації, велика кількість кваліфікованих фахівців, широке використання у громадському виробництві спеціальних технічних засобів – усі ці фактори дозволяють зловмиснику практично в будь-який момент та в будь-якому місці здійснювати дії, що представляють загрозу інформаційній безпеці як в локальному, так і в глобальному масштабах.

Закони й нормативні акти виконуються тільки тоді, коли вони підкріплюються організаторською діяльністю відповідних структур. Під час розгляду питання безпеки інформації така діяльність ставиться до організаційних методів захисту інформації.

Вирішальним фактором існування кіберзлочинців стала популярізація вузькоспеціалізованого ПЗ та доступність отримання знань з його використання. Законотворча система виявила себе з боку нездатності до адаптації до швидкого усунення існуючих проблем.

Організаційно-правове забезпечення інформаційної безпеки являє собою сукупність рішень, законів, нормативів, що регламентують як загальну організацію робіт із забезпечення інформаційної безпеки, так і створення та функціонування систем захисту інформації на конкретних об’єктах. Організаційно правова база має такі основні функції:

- Розробка основних принципів віднесення відомостей, що мають конфіденційний характер, до захищуваної інформації.
- Визначення системи органів та посадових осіб, що відповідають за забезпечення інформаційної безпеки в країні та порядку регулювання діяльності підприємств і організацій в цій області.

- Створення повного комплексу нормативно-правових матеріалів, що регламентують питання забезпечення інформаційної безпеки як у країні в цілому, так і на конкретному об’єкті.
- Визначення міри відповідальності за порушення правил захисту.
- Визначення порядку вирішення спірних і конфліктних ситуацій з питань захисту інформації.

Під юридичними аспектами організаційно-правового забезпечення захисту інформації розуміється сукупність законів та інших нормативно-правових актів, за допомогою яких мали б досягатися наступні цілі:

- усі правила захисту інформації є обов’язковими для дотримання усіма особами, що мають відношення до конфіденційної інформації;
- узаконюються усі міри відповідальності за порушення правил захисту інформації;
- узаконюються (набувають юридичної сили) техніко-математичні рішення питань організаційно-правового забезпечення захисту інформації;
- узаконюються процесуальні процедури розв’язування ситуацій, що виникають у процесі функціонування системи захисту.

Організаційні методи захисту інформації включають заходи та дії, які посадові особи повинні вживати в процесі створення та експлуатації системи для забезпечення заданого рівня інформаційної безпеки. Це включає моніторинг несанкціонованого використання інформації з таких джерел, як ті що містять державну або комерційну таємницю [3].

Розробка законодавчої бази інформаційної безпеки будь-якої держави є необхідною мірою, що задовольняє першочергову потребу в захисті інформації при розвитку соціально-економічних, політичних, військових напрямків розвитку цієї держави. Особлива увага з боку розвинутих країн до формування такої бази викликана все зростаючими затратами на боротьбу з „інформаційними“ злочинами. Все це заставляє серйозно займатися питаннями законодавства із захисту інформації.

Відповідно до законів і нормативних актів у міністерствах, відомствах, на підприємствах для захисту інформації створюються спеціальні служби безпеки. Ці служби підпорядковуються, як правило, керівництву установи. На організаційному рівні вирішуються наступні завдання для забезпечення безпеки інформації в системі:

- розробка систем захисту інформації;
- обмеження доступу до об'єкта та ресурсів системи;
- жорстке мандатне розмежування доступу;
- планування заходів;
- розробка документації;
- підготовка кадрів та підтримка користувачів;
- сертифікація засобів захисту інформації;
- ліцензування діяльності по захисту інформації;
- атестація об'єктів захисту;
- удосконалювання системи захисту інформації;
- оцінка ефективності функціонування системи захисту інформації;
- контроль за виконанням встановлених правил роботи системи.

Організаційні методи є основою комплексної системи захисту інформації в системи. Тільки за допомогою цих методів можливе об'єднання на правовій основі технічних, програмних і криптографічних засобів захисту інформації в єдину комплексну систему. Найбільша увага організаційним заходам приділяється під час вирішення проблем щодо побудови й організації функціонування комплексної системи захисту інформації.

Створенням законодавчої бази в області інформаційної безпеки кожна держава прагне захистити свої інформаційні ресурси. Інформаційні ресурси держави у першому наближенні можуть бути розділені на три великі групи:

- відкрита інформація – розповсюджується без обмежень;
- запатентована інформація – охороняється внутрішньодержавним

законодавством чи міжнародними угодами як об’єкт інтелектуальної власності;

- інформація, що захищається її власником – до цього виду зазвичай відносять інформацію, яка не відома іншим особам, яка або не може бути запатентована, або зумисно не патентується з метою уникнення чи зменшення ризику заволодіння нею суперниками чи конкурентами.

Законом про державну таємницю визначено, що до державної таємниці відносяться відомості, які належать до сфер оборони, економіки, науки, техніки, зовнішніх відносин а також у державної безпеки та охорони правопорядку. Не можна засекречувати інформацію і надавати їй статус державної таємниці у таких випадках:

- якщо її втрата (розголошення та ін.) не приводить до збитків національній безпеці країни, порушення діючих законів;
- якщо приховування інформації буде порушувати конституційні та законодавчі права громадян;
- для приховування діяльності, що наносить збитки навколишньому природному середовищу, що загрожує життю та здоров’ю громадян.

До методів і засобів організаційного захисту інформації відносяться організаційно-технічні й організаційно-правові заходи, проведені в процесі створення й експлуатації системи для забезпечення захисту інформації. Ці заходи повинні проводитися під час будівництва або ремонту приміщень, у яких буде розміщатися системи; проектування системи, монтажу й налагодження головних технічних і програмних засобів; тестуванні та перевірці працездатності готової системи.

Основні методи організаційного захисту:

- обмеження фізичного доступу до об'єктів захисту та реалізація режимних заходів;
- обмеження можливості перехоплення побічного електромагнітного випромінювання;

- розмежування доступу до інформаційних ресурсів і процесів (встановлення правил розмежування доступу , шифрування інформації при її зберіганні і передачі , виявлення та знищення апаратних і програмних закладок);
- резервне копіювання критичних даних;
- будь-які роботи в кімнаті , що проводяться поза часом проведення
- після проведення наради кімната повинна ретельно оглядатися , закриватися і опечатуватися;
- профілактика зараження комп'ютерними вірусами.

Основною метою проведення організаційних заходів є використання й підготовка законодавчих і нормативних документів в області інформаційної безпеки, які на правовому рівні повинні регулювати доступ до інформації з боку користувачів.

Нормативно-правове забезпечення захисту інформації складається з низки законів і підзаконних актів, пов’язаних з різними галузями права: кримінальним, цивільним, адміністративним тощо. Правові механізми захисту різних видів інформації в Україні закріплені, перш за все, в таких нормативних актах, як[4]:

1) Конституція України від 28.06.1996;

2) Цивільний кодекс України від 16.01.2003;

3) Кримінальний кодекс України від 05.04.2001;

4) Кримінальний процесуальний кодекс України від 13.04.2012;

5) Кодекс України про адміністративні правопорушення від 07.12.1984;

6) Кодекс адміністративного судочинства від 06.07.2005;

7) Закон України «Про захист персональних даних» від 01.06.2010;

8) Закон України «Про оперативно-розшукову діяльність» від 18.02.1992;

9) Закон України «Про Національну поліцію» від 02.07.2015;

10) Закон України «Про інформацію» від 02.10.1992;

11) Закон України «Про державну таємницю» від 21.01.1994;

12) Закон України «Про національну безпеку України» від 21.06.2018;

13)Закон України «Про захист інформації в інформаційно-

телекомунікаційних системах» від 05.07.1994;

14) Закон України «Про основні засади забезпечення кібербезпеки України» від 05.10.2017;

15)Доктрина інформаційної безпеки, затверджена Указом Президента України від 25.02.2017 № 47/2017;

16)Правила забезпечення захисту інформації в інформаційних, телекомунікаційних та інформаційно-телекомунікаційних системах, затверджені Постановою Кабінету Міністрів України від 29.03.2006 № 373;

17)Типова інструкція про порядок ведення обліку, зберігання,

використання і знищення документів та інших матеріальних носіїв інформації, що містять службову інформацію, затверджена Постановою Кабінету Міністрів України від 19.10.2016 № 736;

18) інші нормативно-правові акти, що містять окремі питання

захисту інформації.

Окрім законів і підзаконних актів правове забезпечення системи захисту інформації складають внутрішні нормативно-організаційні документи: накази та розпорядження; інструкції, в тому числі посадові; положення; договори; пам’ятки, протоколи тощо.

1.2. Класифікація сучасних мережевих атак та загроз інформації

Мережеві атаки настільки ж різноманітні, як і системи, проти яких вони спрямовані. Деякі атаки відрізняються великою складністю, інші під силу звичайному оператору, який навіть не припускає, до яких наслідків може привести його діяльність. Для оцінки типів атак необхідно знати деякі обмеження, спочатку властиві протоколу TPC/IP. Мережа Інтернет створювалася для зв'язку між державними установами та університетами з метою надання допомоги навчального процесу та наукових досліджень[3]. Творці цієї мережі не підозрювали, наскільки широке поширення вона отримає. В результаті в

специфікаціях ранніх версій Інтернет-протоколу (IP) були відсутні вимоги безпеки. Саме тому багато реалізацій IP є уразливими.

Через багато років, після безлічі рекламацій (Request for Comments, RFC), нарешті стали впроваджуватися засоби безпеки для IP. Однак з огляду на те, що спочатку засоби захисту для протоколу IP не розроблялися, всі його реалізації стали доповнюватися різноманітними мережевими процедурами, послугами і продуктами, що знижують ризики, властиві цим протоколом. Далі ми коротко розглянемо типи атак, які зазвичай застосовуються проти мереж IP.

Перш за все варто сказати, що всі мережеві атаки можна класифікувати за кількома категоріями. А саме: за характером впливу; за метою впливу; за наявністю зворотного зв'язку з об'єктом; за початковими умовами здійснення впливу; за розташуванням порушника щодо об'єкта під загрозою; за рівнем еталонної моделі ISO/OSI, на якому здійснюється вплив.

Розглянемо докладніше.

За характером атак

Дана категорія ділиться ще на два типи: пасивна та активна.

Під пасивною атакою мається на увазі вплив, який не надає безпосередній вплив на роботу системи, але при цьому здатний порушити її політику безпеки. Оскільки відсутній прямий вплив на систему, яка під загрозою, таку атаку складно виявити.

Активна атака, на відміну від пасивної, робить прямий вплив на роботу системи, порушуючи працездатність, змінюючи конфігурацію, або порушуючи політику безпеки. На відміну від пасивної атаки, активну легко виявити, оскільки в системі відбуваються конкретні зміни, тоді як при пасивній не залишається ніяких слідів.

За метою атаки

Такі атаки можна умовно розділити на наступні види: порушення функціонування або доступу до системи, порушення цілісності інформації і її конфіденційності. Те, до якого з цих видів відноситься конкретна атака, залежить від того, яка саме загроза в ній відтворюється: відмова в обслуговуванні, розкриття, або порушення цілісності. Як правило, при будь-якій атаці головною метою є отримання доступу до інформації. Тут є два основні шляхи - це спотворення, або перехоплення. Якщо ми перехоплюємо інформацію, то ми отримуємо до неї доступ, але без можливості її зміни. Але це вже веде до порушення її конфіденційності. У разі підміни інформації вже можливий контроль над обміном даними між об'єктами системи. Відповідно, підміна інформації веде до порушення цілісності, і це вже варіант активного впливу.

За наявністю зворотнього зв'язку з об'єктом

Є два варіанти розвитку подій: зі зворотним зв'язком та без зворотного зв'язку (односпрямована атака).

У першому випадку порушник відправляє цільові запити і очікує певну відповідь. Вони потрібні для того, щоб реагувати на різні зміни на об'єкті. Однак в такому випадку між хакером і об'єктом атаки встановлюється зворотний зв'язок. А ось атаки без зворотного зв'язку не дозволяють своєчасно реагувати на зміни, що виникають під час процесу. Але зловмиснику немає необхідності отримувати і аналізувати відповіді.

За початковими умовами здійснення впливу

- атака за отриманням цільового запиту
- атака по настанню очікуваної події на атакується об'єкті
- безумовна атака

Тут все залежить від того, що в даному конкретному випадку вважається початком атаки. Якщо мова йде про атаку на запит від об'єкта - значить, хакер почне свою справу після того, як система відправить запит певного типу. Як варіант, такими запитами можуть бути DNS- або ARP-запити. Якщо порушник

вже довгий час безперервно спостерігає за станом операційної системи об’єкта, то він може зреагувати на конкретну подію. А безумовна атака - це негайний початок дій без будь-якого очікування або події.

За розташуванням порушника щодо об'єкта під загрозою

Даний вид атаки може відрізнятися виходячи з розташування порушника і цілі. Тому даний вид атак ділять на міжсегментні і внутрішньосегментні. У першому випадку, хакер і мета знаходяться в одному мережевому сегменті. В іншому випадку - у різних.

За рівнем еталонної моделі ISO / OSI, на якому здійснюється вплив

Міжнародною організацією зі стандартизації (ISO) був прийнятий стандарт ISO 7498, який описує взаємодію відкритих систем (OSI), до яких належать також і РВС (розподільні обчислювальні системи). Кожен мережевий протокол обміну, так само як і кожну мережеву програму, можна умовно спроектувати на еталонну 7-рівневу модель OSI. Відповідно, така багаторівнева проекція дає можливість описати в термінах цієї моделі використовуються в мережевому протоколі або програмою функції.

Тому віддалені атаки можна так само описати по еталонної моделі ISO / OSI, в залежності від того, на якій з рівнів здійснюється вплив. Рівнів 7: Прикладний рівень (Application layer); Рівень представлення (Presentation layer); Сеансовий рівень (Session layer); Транспортний рівень (Transport layer); Мережевий рівень (Network layer); Канальний рівень (Data Link layer); Фізичний рівень (Physical layer)

Основні мережеві атаки

Всі види і варіанти атак не перелічити. Але можна перерахувати найпоширеніші.

Фрагментація даних - ініціювання відправки великого числа фрагментів пакета даних, що служить причиною для переповнення програмних буферів на

приймальній стороні і, якщо все пройде успішно, до аварійного завершення системи.

Нестандартні протоколи, інкапсульовані в IP - використання нестандартного значення даного поля для передачі даних, які не будуть фіксуватися стандартними засобами контролю інформаційних потоків.

Smurf - передача в мережу широкомовних ICMP запитів від імені атаки пристрою, після чого комп'ютери, які виконують ці широкомовні пакети, відповідають на запити, через що відбувається зниження пропускної здатності каналу зв'язку, що може привести до повної ізоляція атакується мережі.

Sniffing - прослуховування каналу за рахунок перехоплення пакетів, що передаються по локальній мережі. Сніффер пакетів є прикладною програмою, яка використовує мережеву карту, що працює в режимі promiscuous mode (в цьому режимі всі пакети, отримані по фізичних каналах, мережевий адаптер відправляє додатком для обробки). При цьому сніффер перехоплює всі мережеві пакети, які передаються через певний домен. В даний час сніффери працюють в мережах на цілком законній підставі. Вони використовуються для діагностики несправностей і аналізу трафіку. Однак з огляду на те, що деякі мережеві додатки передають дані в текстовому форматі (Telnet, FTP, SMTP, POP3 і т.д.), за допомогою сніффінгу можна дізнатися корисну, а іноді і конфіденційну інформацію (наприклад, імена користувачів і паролі). Процес перехоплення імен і паролів створює велику небезпеку, так як користувачі часто застосовують один і той же логін і пароль для безлічі додатків і систем. Багато користувачів взагалі мають єдиний пароль для доступу до всіх ресурсів і додатків.

Якщо додаток працює в режимі «клієнт-сервер», а аутентифікаційні дані передаються по мережі у відкритому текстовому форматі, то цю інформацію з великою ймовірністю можна використовувати для доступу до інших корпоративних або зовнішніх ресурсів. Хакери занадто добре знають і використовують людські слабкості (методи атак часто базуються на методах

соціальної інженерії). У найгіршому випадку зловмисник отримує доступ до призначеного для користувача ресурсу на системному рівні і з його допомогою створює нового користувача, якого можна в будь-який момент використовувати для доступу в мережу і до її ресурсів [5].

IP spoofing - підміна вихідного IP-адреса.Ping flooding - посил ICMP-пакет типу ECHO REQUEST. В активному режимі потік ICMP echo request / reply-пакетів може призвести до перевантаження невеликої лінії , через що вона втратить можливість передавати інформацію. Цей тип атаки відбувається в тому випадку, коли хакер, що знаходиться всередині корпорації або поза нею, видає себе за санкціонованого користувача. Це можна зробити двома способами: хакер може скористатися або IP-адресою, що знаходиться в межах діапазону санкціонованих IP-адрес, або вповноваженим зовнішнім адресою, якому дозволяється доступ до певних мережевих ресурсів. Атаки IP-спуфінга часто є відправною точкою для інших атак. Класичний приклад - атака DoS, яка починається з чужого адреси, що приховує справжню особистість хакера.

Як правило, IP-спуфінг обмежується вставкою помилкової інформації або шкідливих команд у звичайний потік даних, переданих між клієнтським і серверним додатком або по каналу зв'язку між однорангових пристроями. Для двостороннього зв'язку хакер повинен змінити все таблиці маршрутизації, щоб направити трафік на помилковий IP-адреса. Деякі порушники, проте, навіть не намагаються отримати відповідь від додатків - якщо головне завдання полягає в отриманні від системи важливого файлу, то відповіді додатків не мають значення. Якщо ж зловмиснику вдається поміняти таблиці маршрутизації і направити трафік на помилковий IP-адреса, він отримує всі пакети і зможе відповідати на них так, як ніби є санкціонованим користувачем.

DNS spoofing - внесення нав'язуваного відповідності між IP-адресою і доменним ім'ям в кеш DNS сервера, в результаті чого всі користувачі DNS сервера можуть отримати невірну інформацію про доменні імена і IP-адреси .

Нав'язування пакетів - відправка в мережу пакетів даних з помилковим зворотною адресою, за допомогою чого можна перемикати на свій комп'ютер з'єднання, встановлені між іншими комп'ютерами, і при цьому права доступу атакуючого стають рівними прав того користувача, чиє з'єднання з сервером було переключено на комп'ютер зловмисника.

Перехоплення пакетів на маршрутизаторі - перехоплення пакетів, що проходять через певний маршрутизатор, шляхом отримання привілейованих доступу до даного маршрутизатора.

Підміна довіреного хоста - посилка пакетів обміну зі станції атакуючого від імені довіреної станції (хоста, легально підключився до Інтернету), що знаходиться під його контролем. Нав’язування хосту хибного маршруту з допомогою протоколу ICMP - посил з будь-якого хоста в сегменті мережі помилкового redirect-повідомлення від імені маршрутизатора на певний хост через спеціальний протокол ICMP (Internet Control Message Protocol), однією з функцій якого є інформування хостів про зміну поточного маршрутизатора, що в підсумку може призвести до активного нав'язування помилкового маршруту всередині одного сегмента мережі. Переадресація портів являє собою різновид зловживання довірою, коли зламаний хост використовується для передачі через міжмережевий екран трафіку, який в іншому випадку був би обов'язково відбракований. Хоча при цьому не порушується жодне правило, чинне на екрані, зовнішній хост в результаті переадресації отримує прямий доступ до захищеного хосту. Прикладом програми, з якою можна надати такий доступ, є netcat.

Парольні атаки. Хакери можуть проводити парольні атаки за допомогою цілого ряду методів, таких як простий перебір (brute force attack), троянський кінь, IP-спуфинг і сніффінг пакетів. Хоча логін і пароль часто можна отримати за допомогою IP-спуфинга і сніффінга пакетів, хакери нерідко намагаються підібрати пароль і логін, використовуючи для цього численні спроби доступу. Такий підхід носить назву простого перебору (brute force attack). Часто для такої

атаки використовується спеціальна програма, яка намагається отримати доступ до ресурсу загального користування (наприклад, до сервера). Якщо в результаті хакеру надається доступ до ресурсів, то він отримує його на правах звичайного користувача, пароль якого був підібраний. Якщо цей користувач має значні привілеї доступу, хакер може створити собі «прохід» для майбутнього доступу, який буде діяти, навіть якщо користувач змінить свої пароль і логін.

Man-in-the-Middle. Для атаки типу Man-in-the-Middle хакеру потрібен доступ до пакетів, що передаються по мережі. Такий доступ до всіх пакетів, що передаються від провайдера в будь-яку іншу мережу, може, наприклад, отримати співробітник цього провайдера. Для атак даного типу часто використовуються сніфери пакетів, транспортні протоколи і протоколи маршрутизації. Атаки проводяться з метою крадіжки інформації, перехоплення поточної сесії і отримання доступу до приватних мережевих ресурсів, для аналізу трафіку і отримання інформації про мережу та її користувачів, для проведення атак типу DoS, спотворення переданих даних і введення несанкціонованої інформації в мережеві сесії.

Атаки на рівні додатків. Атаки на рівні додатків можуть проводитися кількома способами. Найпоширеніший з них - використання добре відомих слабкостей серверного програмного забезпечення (sendmail, HTTP, FTP). Використовуючи ці слабкості, хакери можуть отримати доступ до комп'ютера від імені користувача, що працює з додатком (зазвичай це буває не простий користувач, а привілейований адміністратор з правами системного доступу). Відомості про атаки на рівні додатків широко публікуються, щоб дати адміністраторам можливість виправити проблему за допомогою корекційних модулів (патчів). На жаль, багато хакерів також мають доступ до цих відомостей, що дозволяє їм удосконалюватися.

1.3. Сучасні засоби виявлення мережевих атак.

В даний час зростання масштабів діджиталзації усіх сфер життя українського суспільства призводить до посилення загроз використання проти інтересів України кібернетичних засобів як зсередини держави, так і з-за кордону. На теперішній час можливості систем виявлення вторгнень є необхідним критерієм щодо інфраструктури захисту інформації в системах управління з'єднань та частин ЗС України, які використовують інформаційні системи з підключенням до глобальної мережі Інтернет. Згідно статистики сайту кіберполіції за 2018 рік, було зареєстровано 11131 кримінальних проваджень, з яких виявлено 6000 кримінальних правопорушень [6].

Традиційний підхід до забезпечення безпеки мережевих взаємодій полягає в використанні криптографічних засобів захисту трафіку і міжмережевих екранів, що дозволяють обмежити безліч можливих взаємодій до деякого мінімуму.

У той же час існує загроза використання зловмисником навіть тих мінімальних можливостей доступу, які надають міжмережеві екрани, а криптографічні засоби не забезпечать захисту від шкоди яку можуть заподіяти користувачі.

Засіб виявлення вторгнень (ЗВВ) - програмний або апаратний засіб, призначений для виявлення фактів несанкціонованого доступу в комп'ютерну систему або мережу або несанкціонованого управління ними в основному через Інтернет.

Системи виявлення вторгнень є додатковим механізмом захисту, що дозволяє побудувати ефективну систему захисту від мережевих атак як з боку зовнішніх, так і з боку внутрішніх зловмисників[7].

Засоби виявлення вторгнень використовуються для виявлення деяких типів шкідливої активності, які можуть негативно вплинути на безпеку

комп'ютерної системи. До такої активності відносяться мережеві атаки проти вразливих сервісів, атаки, спрямовані на підвищення привілеїв, неавторизований доступ до важливих файлів, а також дії шкідливого програмного забезпечення.

Зазвичай архітектура ЗВВ включає:

- сенсорну підсистему, призначену для збору подій, пов'язаних з безпекою, що захищається системи;
- підсистему аналізу, призначену для виявлення атак і підозрілих дій на основі даних сенсорів;
- сховище, що забезпечує накопичення первинних подій і результатів аналізу; консоль управління, що дозволяє конфігурувати ЗВВ, спостерігати за станом захищеності системи і ЗВВ, переглядати виявлені підсистемою аналізу інцидентів.

Для того щоб захищеність інформаційної мережі була високою, до ЗВВ висувають такі вимоги:

1) засоби виявлення вторгнень повинні виявляти атаку в реальному часі (проактивний моніторинг, 10050), аналізуючи реальний трафік в мережі, а не журнал аудиту. В цьому випадку бажаною властивістю такої системи буде тісна інтеграція з фаєрволом для негайного блокування трафіку зловмисника;

2) засоби виявлення вторгнень повинні враховувати тенденції розвитку технологій корпоративних мереж - наявність великої кількості комутованих сегментів, логічну структуризацію мережі на основі VLAN, захист внутрішнього трафіку за допомогою VPN і т. п. Тільки в цьому випадку аналіз трафіку буде повним, а захищеність мережі - високою; 3) масштабованість, яка потрібна для виконання ефективного контролю в умовах постійно збільшує кількості сегментів і підмереж, а також кількості захищених вузлів в інформаційній мережі.

На рис. 1.1 зображена архітектура типових систем виявлення вторгнень.

Види СВВ (систем виявлення вторгнення)

СВВ систем, що відрізняються різними алгоритмами моніторингу даних та способами їх аналізу. Кожна з систем має свій ряд переваг та недоліків.

Один із методів класифікації СВВ систем ґрунтується на з'ясуванні того, як вони проводять моніторинг інформаційної системи або мережі. Одні контролюють весь мережевий трафік і аналізують мережеві пакети, інші розгортаються на окремих комп'ютерах і контролюють операційну систему на предмет виявлення ознак небажаної активності.

За способами моніторингу СВВ системи підрозділяються на network-based (NIDS) і host-based (HIDS)[7].

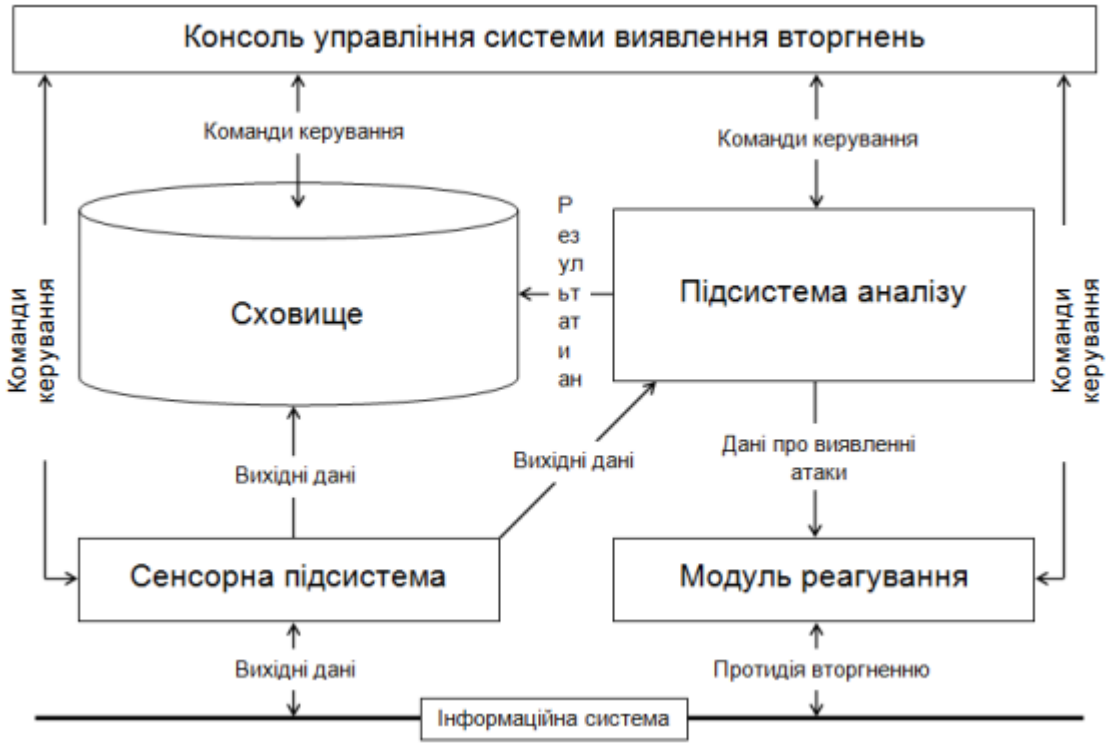

Рис. 1.1. Архітектура типових систем виявлення вторгнень

NIDS (network-based) визначають атаки за допомогою зчитування та аналіза мережевих пакетів. Прослуховуючи мережевий сегмент, NIDS може

обробляти трафік від кількох хостів, які приєднані до мережевого сегменту, і таким чином захищати ці хости.

Зважаючи на особливості розташування таких систем виявлення можна виділити такі переваги:

- не впливають на продуктивність існуючої інформаційної системи;
- велике покриття для моніторингу та у зв'язку з цим централізоване управління;
- такі СВВ, як правило, працюють у реактивному режимі (10051 порт).

Але на фоні важливих переваг такого виду систем виявлення вторгнень спостерігаються наступні недоліки:

- не здатні аналізувати зашифровану інформацію;
- повідомляють про ініційований напад, не аналізуючи ступінь проникнення;
- не в змозі розпізнавати напад в момент високого навантаження інформаційної системи;
- не можуть розпізнати результат атаки. NIDS не можуть сказати чи була атака успішної, вони можуть тільки визначити, що атака була почата. Це означає, що після того як NIDS визначить атаку, адміністратор повинен вручну досліджувати кожен атакований хост для визначення, чи відбувалося реальне проникнення;
- вимагають додаткового налаштування і функціональності мережевих пристроїв;

HIDS (host-based) мають справу з інформацією, що збирається всередині єдиного комп'ютера. Таке вигідне розташування дозволяє HIDS аналізувати діяльність з великою вірогідністю і точністю, визначаючи тільки ті процеси і

користувачів, які мають відношення до конкретного вторгнення в операційну систему інформаційної мережі. HIDS зазвичай використовують інформаційні джерела двох типів: результати аудиту операційної системи і системні логи.

За рахунок вигідного розташування безпосередньо на сервері, такий вид систем виявлення вторгнень має значні переваги:

- не вимагають додаткової функціональності мережевих пристроїв;
- мають можливість стежити за подіями локально щодо хоста, можуть визначати атаки, які не можуть бачити NIDS;
- працюють в мережі, що використовує шифровані дані, коли інформація знаходиться в відкритому вигляді на сервері до її відправки клієнту;

Але через особливості роботи операційної системи, місце розташування систем виявлення та методи вторгнень можна спостерігати наступні недоліки:

- механізми збору інформації(опитування агентів) повинні встановлюватися і підтримуватися на кожному сервері та пристої, який буде контролюватися;
- можуть бути атаковані і заблоковані підготовленим зловмисником;
- не здатні контролювати ситуацію у всій мережі, так як «бачать» тільки мережеві пакети, отримувані сервером, на якому вони встановлені;
- використовують обчислювальні ресурси сервера, який контролюють, знижуючи тим самим ефективність його роботи.

1.4.Висновки до розділу 1

Таким чином, підсумовуючи викладене в розділі, можна зробити наступні висновки:

1) Основою правового регулювання у сфері інформаційної безпеки є

Закон України «Про інформацію», який встановлює загальні правові засади основних видів ІД, таких як одержання, використання, поширення, зберігання інформації тощо. Даний закон визначає основні принципи інформаційних відносин, гарантує право громадян на інформацію, встановлює режим доступу до інформації та його контроль, дає визначення інформації з обмеженим доступом.

2) Пряме відношення до теми даної дипломної роботи має Закон України «Про захист інформації в інформаційно-телекомунікаційних системах», оскільки він регулює відносини у сфері ЗІ в інформаційних (автоматизованих) системах. Згідно з цим законом об’єктами захисту в системі є інформація, що обробляється в ній, та програмне забезпечення, яке призначено для обробки цієї інформації. Закон встановлює порядок доступу до інформації в системі, умови її обробки, забезпечення захищеності даних засобами технічного та/або криптографічного ЗІ. Закон також встановлює відповідальність за порушення законодавства про ЗІ в системах.

3) Галузь права на даний момент досить слабка по відношенню до інформаційної безпеки, кібершпигунства та відстеження перехоплення цінної інформації.

4) Чи можна виявити вторгнення в інформаційну систему? При всіх перерахованих недоліках дати ствердну відповідь можна. Найефективнішою СВВ можна вважати ту, яка працює в реальному часі і здійснює моніторинг трафіку мережі, а також відстежує аномалії, як в трафіку так і в діях користувачів і системи. Для ефективного виявлення атаки потрібно застосування як сигнатурного, так і поведінкового методів. Найперспективнішим методом можна вважати комбінований метод, що використовує спільно алгоритми, визначені в сигнатурних методах і методах виявлення аномалій. Так як, тільки комплексний підхід може значно знизити ризик вторгнення в ІС і виключити втрату цінних даних. Для підвищення ймовірності виявлення атак на ІС потрібний збір великих обсягів інформації про функціонування ІС, а також їх аналіз.

5) Найкращим захистом для інформаційної системи з’єднання буде гібридне використання одночасно обох видів систем IDS у одній інформаційній мережі. Але варто пам’ятати, що системи IDS це універсальний інструмент захисного арсеналу. Захист інформації потребує комплексного підходу і найбільш ефективний, коли в мережі підтримується багаторівневий захист.

РОЗДІЛ 2. СУЧАСНІ РІШЕННЯ В СФЕРІ МОНІТОРИНГУ ІНТЕРНЕТ-ТРАФІКУ КОРИСТУВАЧІВ

2.1. Характеристика сучасних методів моніторингу та виявлення атак

В даний час відсутній загальний підхід до вирішення проблеми виявлення аномальних ситуацій під час обробки інформації комп'ютерними системами та інформаційними мережами. Однак в умовах активного розвитку інформаційних технологій і постійної модернізації програмного і апаратного забезпечення комп'ютерних систем, рішення задач виявлення аномалій не може забезпечувати безпеку системи[8].

Методи виявлення аномалій часто застосовуються для вирішення завдань виявлення атак на обчислювальні системи і інформаційні мережі. Вони вибираються стосовно до певного набору параметрів системи, і їх ефективність е залежить тільки для цього набору параметрів. Класифікація за схемою прийняття рішення представляється найбільш адекватною з позицій теорії розпізнавання образів, до якої в загальному випадку стосується ця задача.

Сигнатурні методи - методи виявлення вторгнень на основі сигнатур зазвичай використовуються в системах виявлення вторгнень, в яких містяться сигнатури (шаблони) типових атак, створені на основі заголовків або вмісту мережевих пакетів. Велика кількість сигнатур робить цей метод більш витратним з точки зору вартості обчислень. Для вирішення цього обмеження був запропонований метод, що поєднує новий метод аналізу даних з традиційним зіставленням з сигнатурами.

Головною перевагою цього підходу є збільшення продуктивності сигнатурного методу і зменшення помилкових спрацьовувань, оскільки пошук йде тільки в певних частинах пакетів. Переваги сигнатурного методу: ефективне визначення атак на ІС;

- відсутність великої кількості помилкових спрацьовувань;

- достовірно оцінити використання конкретного інструментального засобу або методу атаки;
- можливість найбільш точно задати параметри сигнатури.

Недоліки сигнатурного методу:

- необхідність оновлювати бази сигнатур для виявлення нових атак; неможливість виявлення атак, що не описаних в експертній системі;
- неможливість виявити атаки, що відрізняються від сигнатурного опису, або без опису.

Поведінкові методи - засновані не на моделях інформаційних атак, а на моделях «нормального» функціонування ІС. Принцип роботи будь-якого з цих методів полягає в виявленні розбіжностей між поточним режимом роботи інформаційної системи і режиму роботи, який є еталонним для цієї ІС. Будь-яка невідповідність розглядається як вторгнення або аномалія. Складність даного принципу є створення точної еталонної моделі «нормального» режиму інформаційної системи. Переваги поведінкового методу:

- визначення атаки без знання конкретних деталей (сигнатури);
- детектори аномалій можуть створювати інформацію, яка в подальшому буде використовуватися для визначення сигнатур атак;
- висока чутливість до змін станів ІС.

Недоліки поведінкового методу:

- помилкові сигнали при непередбаченому поводженні користувачів;
- помилкові спрацьовування при непередбачуваною мережевої активності;
- часові витрати на етапі навчання системи.

У кожного принципу є свої методи, побудовані на основі цих принципів. Сигнатурні методи: Метод контекстного пошуку полягає в виявленні в вихідної інформації певного набору символів. Наприклад, для виявлення атак на веб-сервера під Unix-подібні ОС, спрямованої на отримання несанкціонованого

доступу до файлу паролів проводиться пошук послідовності символів "Get * / etc / password" в заголовку запиту HTTP.

Методи аналізу стану засновані на формуванні сигнатур атак у вигляді послідовності переходів інформаційної системи з одного стану в інший, набори таких подій задається параметрами сигнатур атак.

Поведінкові методи (методи виявлення аномалій): Методи, засновані на статистичних моделях, визначають статистичні показники, що характеризують параметрами регулярного поведінки системи. Якщо з плином часу є деяке відхилення цих параметрів від заданих значень, то фіксується виявлення атаки. Комбіновані методи:

1. Метод продукційних правил дозволяють описати моделі для атак на природній мові з високим рівнем абстракції. Експертні системи, що використовують ці методи, складаються з двох баз даних: факти і правила. Факти є вхідними даними з інформаційної системи, а правила - алгоритми для логічних рішень про факт нападу на основі вхідного набору фактів. Отримана система правил повинна описувати характеристики атак, які повинна виявити система виявлення вторгнення.

2. База правил. За допомогою експертної системи може точно визначити взаємодію між вузлами ІС, яке завжди здійснюється відповідно до визначених протоколами. Якщо в процесі обміну інформацією між вузлами з'являється невідома команда або нестандартне значення одного з параметрів, можна вважати це ознакою атаки.

3. Метод імітації поведінки біологічних систем використовує алгоритми, моделей, засновані на біологічних об'єктах, таких як генетичні алгоритми і штучні нейронні мережі (ШНМ). Методи, засновані на біологічних моделях, вважаються найбільш перспективними, в першу чергу, через їх адаптації та саморозвитку.

На етапі вторгнення можна виявляти атаку як сигнатурним, так і поведінковим методом. Будь-яке вторгнення характеризується певними особливостями, які з одного боку, можуть бути представлені у вигляді сигнатур, а з іншого - як свого роду відхилення від еталонної поведінки інформаційної системи. Найбільш ефективне поєднання обох методів одночасно, при цьому, для отримання необхідних вхідних дані застосувати будь-які (мережеві або вузлові) датчики.

Чимало спроб побудови ADS вже зроблено. Більшість з них це концептуальні моделі, мета яких – перевірити можливість застосування математичної моделі чи підходу. Комерційних продуктів IDS дуже мало, а існуючі майже ніколи не виходять за межі MDS. Практично всі описані методи для виявлення аномалій можна розділити на:

а) ті, що базуються на зберіганні прикладів поведінки;

б) частотні;

в) нейромережеві;

г) ті, що будують скінченні автомати;

д) інші спеціальні.

Основним завданням є пошук найоптимальніших методів побудови ADS у системах IDS[9]. Методи, що основані на зберіганні прикладів поведінки - найпростішим підходом є пряме запам’ятовування прикладів дій, послідовностей команд користувачів чи взагалі будь-яких параметрів, доступних реєстрації (instance-based learning). Попри неможливість застосування цього підходу в інших випадках моделювання людської поведінки, у задачах виявлення вторгнень він є ефективним, що зумовлено обмеженою кількістю можливих дій суб’єктів комп’ютерної системи, значною детермінованістю задач, що можна виконати, і самою структурою операційної системи. Реакцією комп’ютерної системи на аномальну поведінку процесу є його примусове уповільнення. Отже,

процеси, що демонструють жваву аномальну поведінку, будуть майже зупинені й автоматично знищені системою як такі, що не реагують на запити.

Зафіксовані раніше підпослідовності системних викликів, що входили в тренувальну множину, запам’ятовуються і під час роботи перевіряється їх наявність в поточній сесії. Оскільки спостереження ведеться за системними викликами програми, то через їх регулярність (послідовність викликів і їх типи значно детерміновані вихідним кодом програми) розміри бази підпослідовностей не будуть значними. Підпослідовність з поточної сесії, якої не було серед тренувальних, вважається аномальною. Підхід потребує доопрацювання ядра операційної системи, що не завжди можливо. Крім того, постійна наявність такого моніторингового компонента призводить до загального уповільнення роботи всієї системи, яке становить 4-50 %.

Метод на базі частотної моделі

Розвиненням ідей instance-based систем є врахування частотного розподілу параметрів системи. Вже було запропоновано зберігання інформації про суб’єктів у патернах активності – виражених у статистичних термінах наборах характеристик поведінки суб’єкта відносно певного об’єкта, таких як: авторизація, запуски програм, доступи до файлів і пристроїв з метою реєстрації аномалій. Потім йде перевірка, чи попадає відносна кількість певних подій у заданий експертом інтервал. Модифікацією частотного підходу є робота де пропонується метод, оснований на так званих структурних нулях. Він полягає у використанні інформації про команди, які використовують дуже рідко або зовсім не використовують, – відповідні їм комірки у таблиці ймовірностей дорівнюють нулю, тобто є структурними нулями. Вводиться індекс унікальності, що обчислюється для кожної сесії та кожного користувача і. Цей індекс отримує додатний приріст для частих команд у межах поточної сесії, який тим менший, чим частіше ця команда використовується взагалі. Так, широке використання рідких команд спричинить великі значення індексу унікальності. В разі

виникнення команд, не властивих користувачеві, індекс зменшується. Припустивши, що значення індексу є стабільним для певного суб'єкта, автори намагаються розрізняти їх за значеннями індексу. Поширеними недоліками частотних методів є неадаптивність, оскільки часто еталонні значення частот визначаються одноразово, за тренувальною множиною або за експертними даними, і невраховування послідовності виконання команд.

Метод на базі нейромережевої моделі

Застосування нейроних мереж зумовлене самою неформальною постановкою задачі – виявити аномальні події. Ідея полягає в тому, щоб, отримавши деяку "тренувальну" множину параметрів вхід-вихід, що характеризують поведінку системи, дати мережі адаптуватися до них і вважати нормою. Виходом може бути деякий коефіцієнт "нормальності" поведінки або один із параметрів системи. Якщо вхідні дані мають закономірності, то роблять припущення, що мережа здатна до навчання на запропонованих паттернах.

Якщо в процесі роботи запропонований нейронною мережею вихід є деяким коефіцієнтом, попадає в небезпечну область або відрізняється від наявного в реальній системі за умови, що це один із параметрів системи, то робиться висновок, що в системі наявна аномалія.

Для побудови шаблону поведінки користувача використовуються такі параметри: години роботи, набір вузлів, з яких він починає робочу сесію, характеристики використання ресурсів системи. Ці параметри оброблюються і є вхідними для нейронної мережі зворотного поширення помилки (backpropagation neural network, BPNN), а вихідним є коефіцієнт, що дорівнює нулю для користувача з нормальною поведінкою й одиниці – з аномальною. Тобто мережа тренується на парах типу ("нормальні" параметри, 0) та ("аномальні" параметри, 1).

Оскільки для отримання "ненормальної" поведінки треба було б змусити користувача поводитися не так, як він звик, то аномальні дані генеруються

випадково, що ускладнює інтерпретацію результатів відносно роботи з реальними даними. Вихідні дані – ідентифікатор користувача (параметр, який, хоч і має в UNIX числове значення, але створює штучну близькість користувачів з близькими ідентифікаторами). Якщо користувача ідентифіковано неправильно, до адміністратора від системи надходить лист. Якщо отримано результати, але лише з 10 користувачами, то такі умови є “тепличними”, оскільки в реальних системах кількість користувачів досягає кількох тисяч, причому більшість виконує однотипні дії, що ускладнює їх розрізнення таким способом.

Результати свідчать про можливість застосування такого підходу і про ефективність вибраного методу кодування команд, проте нерегулярність поведінки користувачів значно підвищує рівень помилок типу false positives. Очікується, що кращі результати можна отримати на даних від регулярніших джерел, таких як системні процеси. Недослідженою залишається можливість застосування нейронних мереж неперсептронного типу, наприклад, ансамблевих.

Недоліком багатьох нейронних мереж є їхня погана пристосованість для роботи з невпорядкованими величинами. Введення штучного порядку на множині значень елементів тільки спотворить картину, оскільки нейронна мережа враховуватиме близькість числових величин.

Метод, що базується на моделях, які будують скінченні автомати

В цьому методі досягається більша моделювальна здатність, ніж у разі використання тривіальних частотних та instance-based методів. Вхідні дані розглядаються як потік дискретних подій, наприклад, системних викликів або ідентифікаторів процесів.

Мета складається у отриманні алгоритма, який моделює вказану послідовність подій. Для багатьох послідовностей характерно, що ймовірність наступного символу, елемента або сигналу залежить від попередніх. Часто вони залежать лише від невеликої кількості попередніх. Це наштовхує на думку

моделювати їх за допомогою марковських ланцюгів. Проте у разі зростання порядку ланцюгів, що може суттєво збільшити точність моделі, кількість станів відповідного автомата поводить себе як O(ΣL), де Σ – розмір алфавіту символів; L – порядок ланцюга. Це ставить великі вимоги до ресурсів і збільшує час обробки. За вхідними даними будується матриця переходів ланцюга першого порядку і ймовірність сесії визначається як добуток імовірностей переходу між станами, що відповідають елементарним подіям у файлі аудиту.

За другим способом будується імовірнісний автомат з множиною входів – подіями з аудит-потоку і множиною виходів – послідовністю чисел, що вказують на ступінь аномальності вхідних подій. Ці числа показують, як імовірність наявного переходу з поточного стану в наступний співвідноситься з вектором перехідних імовірностей з цього стану. Попри деяку необґрунтованість вибору саме такої міри аномальності, підхід на експерименті виявився ефективним.

Інші методи

Крім описаних вище методів, для виявлення аномалій використовуються байєсівські мережі – це графічні моделі представлення у власній структурі залежностей між об’єктами та розподілів імовірностей. За множиною тренувальних даних оцінюються кореляції між станами і будується мережа зі з’єднаними, залежними між собою, вузлами, ймовірності переходів між станами, які заповнюються, і зв’язані з вузлами таблиці розподілів імовірностей за даними станами вузлів предків.

Через жорстку залежність побудованої структури від наданих тренувальних даних отримана модель не є адаптивною.

За вибірку береться множина бінарних рядків фіксованої довжини. Кількість рядків у множині також фіксована. У процесі еволюції, за адаптацією, вибираються рядки, з яких генерується наступне покоління рядків. Генерація наступного покоління відбувається з використанням трьох основних операторів над рядками: селекція (вибір найпристосованішого представника), репродукція

(обмін частинами рядків між собою) і мутація (випадкові зміни бітів у рядках для запобігання незворотній втраті інформації).

Генетичні алгоритми використовуються для знаходження песимістичного сценарію в гібридній задачі виявлення зловживань і аномалій. З одного боку, виявляються тільки відомі атаки, з іншого – застосування генетичних алгоритмів і несуворих обмежень на гіпотези може дозволити реєструвати також і варіації цих атак. Підхід перевірено на штучних даних зі змодельованими простими атаками.

Отже, в реальних IDS, спираючись на аналіз вищезгаданих методів, можна рекомендувати реалізувати комбінацію різних методів і згідно з нею робити остаточний висновок про наявність чи відсутність вторгнень та їх характер[10].

Моделі в цих методах не повинні залежати від конкретного типу аудит-даних, тоді їх можна застосувати до будь-яких аудит-послідовностей. Так, такими даними, окрім команд, що безпосередньо виконує користувач, можуть стати послідовності системних викликів, характерні для програми, значення інтервалів між натисканнями клавіш на клавіатурі, інтервали між робочими сесіями, послідовні значення координат положення курсора миші на екрані, послідовності значень полів заголовка мережевих пакетів усіх рівнів або величини, що в них не містяться, але від них залежать.

Тобто будь-яка послідовність сигналів, команд, елементів або хронологічних чи інших значень, характерна для користувача, процесу, системи або мережевого сегмента, може бути використана в ADS. Зрозуміло, що користувачам властиво змінювати свою поведінку (через зміну задач, набуття нових звичок), тому моделі методів обов’язково повинні бути адаптивними. Велика частина атак – це атаки проти системних програм, наприклад, набуття прав суперкористувача шляхом використання вразливостей у SUID програм. Після цього програма, як правило, демонструє кардинально відмінний характер системних викликів від тих, що спостерігалися до атаки. Тому слід звернути

увагу на виявлення аномалій на рівні системних викликів, що стає дедалі популярнішим, оскільки дає змогу абстрагуватися від нерегулярної людської поведінки.

Одночасно не можна відмовлятися від аналізу аудит-даних, що надходять від користувачів, оскільки тільки аналіз на цьому рівні дає можливість виявити вторгнення, які на рівні системних викликів не проявляються (наприклад, використання вкраденого пароля). Повноцінна IDS повинна містити також компоненту з виявлення зловживань, оскільки сьогодні атаки з використанням експлойтів залишаються одними з основних видів атак. Але є ряд причин, за яких нейромережевий спосіб найкращий, не дивлячись на те що його впровадження потребує чималих зусиль та адаптації під потреби бізнесу.

2.2. Класифікація засобів моніторингу та аналізу

Постійний контроль працездатності локальної мережі, що становить основу безпеки будь-якої корпоративної мережі, необхідний для підтримки її в працездатному стані. Контроль - це необхідний комплекс заходів, який повинен виконуватися при управлінні мережею. Зважаючи на важливість цієї функції її часто відокремлюють від інших систем управління і реалізують спеціальними засобами. Такий поділ функцій контролю і власне управління корисно для невеликих і середніх мереж, для яких встановлення інтегрованої системи управління економічно недоцільна. Використання автономних засобів контролю допомагає адміністратору мережі незалежно виявити проблемні ділянки і досліджувати топографію мережі, у той час як відключення або реконфігурацію він може виконувати в цьому випадку вручну.

Процес контролю роботи мережі поділяють на два етапи – аналіз та моніторинг[11].

На етапі моніторингу збираються первинниі даниі про роботу мережі:

статистика про кількість обміну в мережі кадрів і пакетів різних протоколів, стан портів концентраторів, комутаторів і маршрутизаторів і т. п.

Далі виконується аналіз, що собою являє процес осмислення зібраної на етапі моніторингу інформації, зіставлення її з даними, отриманими раніше, і вироблення припущень щодо можливих причини сповільненої або ненадійної роботи мережі.

Завдання моніторингу вирішуються програмними і апаратними каунтерами, тестерами, мережевими аналізаторами, вбудованими засобами моніторингу комунікаційних пристроїв, а також агентами систем управління. Завдання аналізу вимагає більш активної участі людини і використання таких складних засобів, як експертні системи, що акумулюють практичний досвід багатьох мережевих фахівців.

Всі засоби моніторингу та аналізу мереж, можна розділити на кілька великих класів:

Системи управління мережею (Network Management Systems) – це централізовані програмні системи, що збирають дані про стан вузлів і комунікаційних пристроїв мережі, дані про трафік в мережі. Ці системи не тільки здійснюють моніторинг і аналіз, а й виконують в автоматичному чи напівавтоматичному режимі адміністрування мережі - включення і відключення портів пристроїв, зміни параметрів адресних таблиць мостів, комутаторів і маршрутизаторів і т.п. Прикладами систем управління можуть служити популярні системи HPOpenView, SunNetManager, IBMNetView.

Засоби управління системою (System Management). Об'єктом управління є програмне і апаратне забезпечення комп'ютерів мережі, а також - комунікаційне устаткування. Разом з тим, деякі функції цих двох видів систем управління можуть дублюватися, наприклад, засоби управління системою можуть виконувати найпростіший аналіз мережевого трафіку.До найбільш відомих систем управління системами відносяться LANDesk, IBM Tivoli, Microsoft

Systems Management Server, HP OpenView, Novell ZENworks і CA Unicenter.

Вбудовані системи діагностики і управління (Embedded Systems). Ці системи впроваджуться у вигляді програмно-апаратних модулів, які встановлюються в комунікаційне обладнання, а також у вигляді програмних модулів, вбудованих в операційні системи. Вони виконують функції діагностики і управління тільки одним пристроєм, і в цьому їх основна відмінність від централізованих систем управління. Прикладом засобів цього класу може служити модуль управління концентратором Distrebuted 5000, яке реалізує функції автосигментації портів при виявленні несправностей, приписування портів внутрішнім сегментам концентратора і деякі інші. Як правило, вбудовані модулі управління також виконують роль SNMP-агентів, які поставляють дані про стан пристрою системам управління.

Аналізатори протоколів (Protocolanalyzers). Представляють собою програмні або апаратно-програмні системи, які виконують на відміну від систем управління лише функціями моніторингу і аналізу трафіку в мережах. Оптимальний аналізатор протоколів може захоплювати і декодувати пакети великої кількості протоколів, що застосовуються в мережах - зазвичай кілька десятків. Ці засоби дозволяють встановити деякі логічні умови для захоплення окремих пакетів і виконують повне декодування захоплених пакетів, тобто показувати в зручній для користувача формі вкладеність пакетів протоколів різних рівнів один в одного з розшифруванням змісту окремих полів кожного пакета.

Обладнання для діагностики і сертифікації кабельних систем. Є чотири групи: мережеві монітори, прилади для сертифікації кабельних систем, кабельні сканери і тестери (мультиметри). Мережеві монітори (називають також мережевими аналізаторами) призначені для тестування кабелів різних категорій. Слід розрізняти мережеві монітори і аналізатори протоколів. Мережеві монітори збирають дані тільки про статистичні показники трафіку - середньої

інтенсивності загального трафіку мережі, потоку пакетів з певним типом помилки і т.п. Сертифікація виконується відповідно до вимог одного з міжнародних стандартів. Кабельні сканери використовуються для діагностики мідних кабельних систем. Тестери призначені для перевірки кабелів на відсутність фізичного розриву.

Експертні системи. Типаж цих систем являє собою компдекс методів щодо виявлення причин аномальної роботи мереж і можливі способи приведення мережі у працездатний стан. Експертні системи часто реалізуються у вигляді окремих підсистем різноманітних засобів моніторингу та аналізу мереж: систем управління мережами, аналізаторів протоколів, мережевих аналізаторів. Найпростішим варіантом експертної системи є контекстно-залежна help-система. Більш складні системи представляють собою бази знань, які включають в себе елементи штучного інтелекту. Прикладом такої системи є експертна система, вбудована в систему управління Spectrum компанії Cabletron.

Багатофункціональні пристрої аналізу та діагностики. У зв'язку з розповсюдженням локальних мереж виникла необхідність розробки бюджетних портативних приладів, які суміщають функції декількох пристроїв: аналізаторів протоколів, кабельних сканерів і, навіть, деяких можливостей ПЗ мережного управління. Як приклад такого роду пристроїв можна привести Compas компанії MicrotestInc. або 675 LANMeterкомпанii FlukeCorp.

2.2.1. Програмні засоби

Аналізатори протоколів

У ході проектування мережі часто виникає необхідність в кількісному вимірі деяких характеристик мережі, як інтенсивності потоків даних по лініях зв'язку, затримки, що виникають на різних етапах обробки пакетів, часи реакції на запити того чи іншого виду, частота виникнення певних подій та інших характеристик.

Для цих цілей можуть бути використані різні засоби і насамперед - засоби

моніторингу в системах управління мережею, які вже обговорювалися в попередніх розділах. Деякі вимірювання мережі можуть бути виконані і вбудованими в операційну систему програмами.

Але найбільш досконалим засобом дослідження мережі є аналізатор протоколів[12]. Аналізатор мережевих протоколів може використовуватися для:

- локалізації складних проблем;
- виявлення та ідентифікації несанкціонованого програмного забезпечення;
- отримання такої інформації, як базові моделі трафіку (baseline traffic patterns) і метрики утилізації мережі;
- ідентифікації невикористовуваних протоколів для видалення їх з мережі;
- генерації трафіку для випробування на вторгнення (penetration test) з метою перевірки системи захисту;
- роботи з системами виявлення вторгнень Intrusion Detection System (IDS);
- прослуховування трафіку, тобто локалізації несанкціонованого трафіку з використанням Instant Messaging (IM) або бездротових точок доступу Access Points - (AP);
- вивчення роботи мережі.

Аналізатор протоколів є самостійним спеціалізованим пристроєм, або персональним комп'ютером, зазвичай переносним, класу Notebook, оснащений спеціальною мережевою картою і відповідним програмним забезпеченням. Мережева карта і програмне забезпечення, що використовуються повинні відповідати топології мережі (кільце, шина, зірка). Аналізатор підключається до мережі точно так, як і звичайний вузол. Відмінність полягає в тому, що аналізатор може приймати всі пакети даних, що передаються по мережі, в той час як звичайна станція - лише адресовані їй. Програмне забезпечення

аналізатора складається з ядра, що підтримує роботу мережевого адаптера і декодує одержувані дані, та додаткового програмного коду, що залежить від типу топології досліджуваної мережі. Крім того, поставляється ряд процедур декодування, орієнтованих на певний протокол, наприклад, IPX. До складу деяких аналізаторів може входити також експертна система, яка може видавати користувачеві рекомендації про те, які експерименти слід проводити в даній ситуації, що можуть означати ті чи інші результати вимірювань, як усунути деякі види несправності мережі.

Незважаючи на відносне різноманіття аналізаторів протоколів, представлених на ринку, можна назвати деякі риси, в тій чи іншій мірі притаманні всім їм:

Інтерфейс користувача. Більшість аналізаторів мають розвинений дружній інтерфейс, який базується, як правило, на Windows чи Motif. Цей інтерфейс дозволяє користувачеві: виводити результати аналізу інтенсивності трафіку; отримувати миттєву і середню статистичну оцінку продуктивності мережі; задавати певні події і критичні ситуації для відстежування їх виникнення; робити декодування протоколів різного рівня і представляти в зрозумілій формі вміст пакетів.

Буфер захоплення. Буфери різних аналізаторів відрізняються за обсягом. Буфер може розташовуватися на мережевій карті, або для нього може бути відведено місце в оперативній пам'яті одного з комп'ютерів мережі. Якщо буфер розташований на мережевій карті, то управління ним здійснюється апаратно, і за рахунок цього швидкість введення підвищується. Однак це призводить до подорожчання аналізатора. У разі недостатньої продуктивності процедури захоплення, частина інформації буде губитися, і аналіз буде неможливий. Розмір буфера визначає можливості аналізу по більш або менш представницьким вибіркам даних, що захоплюються. Але яким би великим не був буфер захоплення, рано чи пізно він заповниться. У цьому випадку або припиняється

захоплення, або заповнення починається з початку буфера.

Можливість вимірювання середньостатистичних показників трафіку в сегменті локальної мережі, в якому встановлений мережевий адаптер аналізатора.

Вимірюється коефіцієнт використання сегменту, матриці перехресного трафіку вузлів, кількість хороших і поганих кадрів, що пройшли через сегмент.

Можливість роботи з декількома агентами, котрі поставляють захоплені пакети з різних сегментів локальної мережі. Ці агенти найчастіше взаємодіють з аналізатором протоколів за власним протоколом прикладного рівня.

Фільтри. Фільтри дозволяють керувати процесом захоплення даних, і, тим самим, дозволяють економити простір буфера. Залежно від значення певних полів пакета, заданих у вигляді умови фільтрації, пакет або ігнорується, або записується в буфер захоплення. Використання фільтрів значно прискорює і спрощує аналіз, оскільки виключає перегляд непотрібних в даний момент пакетів.

Перемикачі - це деякі умови початку і припинення процесу захоплення даних з мережі, що задаються користувачем. Такими умовами можуть бути виконання ручних команд запуску і зупинки процесу захоплення, тривалість процесу захоплення, поява певних значень в кадрах даних. Перемикачі можуть використовуватися спільно з фільтрами, дозволяючи більш детально й тонко проводити аналіз, а також продуктивніше використовувати обмежений обсяг буфера захоплення.

Пошук. Деякі аналізатори протоколів дозволяють автоматизувати перегляд інформації, що знаходиться в буфері, і знаходити в ній дані за заданими критеріями. У той час, як фільтри перевіряють вхідний потік на предмет відповідності умовам фільтрації, функції пошуку застосовуються до вже накопичених в буфері даних.

Багатоканальність. Деякі аналізатори протоколів дозволяють проводити одночасний запис пакетів від декількох мережевих адаптерів, що зручно для зіставлення процесів, що відбуваються в різних сегментах мережі. Можливості аналізу проблем мережі на фізичному рівні у аналізаторів протоколів мінімальні, оскільки всю інформацію вони отримують від стандартних мережевих адаптерів. Тому вони передають і узагальнюють інформацію фізичного рівня, яку повідомляє їм мережевий адаптер, а вона багато в чому залежить від типу мережного адаптера. Деякі мережні адаптери повідомляють більш детальні дані про помилки кадрів та інтенсивності колізій в сегменті, а деякі взагалі не передають таку інформацію верхнім рівням протоколів, на яких працює аналізатор протоколів.

Методологія проведення аналізу може бути представлена у вигляді наступних етапів[13]:

- Аналізатор протоколів виконує моніторинг мережевого трафіку
- Аналізатор працює на станції хоста.

Коли аналізатор запускається в хаотичному режимі (promiscuous mode), драйвер мережевого адаптера, NIC, перехоплює весь трафік, що проходить через нього. Аналізатор протоколів передає перехоплений трафік декодеру пакетів аналізатора (packet - decoder engine), який ідентифікує і "розщеплює" пакети по відповідним рівнями ієрархії. Програмне забезпечення протокольного аналізатора вивчає пакети і відображає інформацію про них на екрані хоста у вікні аналізатора. Залежно від можливостей конкретного продукту, представлена інформація може згодом додатково аналізуватися і фільтруватися.

Зазвичай вікно протокольного аналізатора складається з трьох областей. Верхня область відображає підсумкові дані перехоплених пакетів. Зазвичай в цій області відображається мінімум полів, а саме:

- дата та час (у мілісекундах), коли пакети були перехоплені;
- вихідні і цільові IP- адреси;

- вихідні і цільові адреси портів;
- тип протоколу (мережевий , транспортний або прикладного рівня);
- деяка сумарна інформація про перехоплених даних.

У середній області показано детальну інформацію про пакет згідно мережевої моделі OSI. І нарешті, в нижній області пакет представлений в шістнадцятковому вигляді або в символьній формі - ASCII.

2.2.2. Апаратні засоби

Обладнання для діагностики та сертифікації кабельних систем

Кабельна мережа (дротова мережа, лінія зв'язку) – це мережа, елементами якої є кабельні лінії й компоненти. До кабельних компонентів належить все пасивне комутаційне устаткування, що слугує для з'єднання або фізичного закінчення (термінування) кабелю.

Умовно, обладнання для діагностики кабельних систем можна поділити на три основні групи: мережеві аналізатори, кабельні сканери та тестери[14]. Для вибору відповідного обладнання потрібно визначитись з ТЗ. Правила вибору засобів (див. табл. 2.1):

Табл. 2.1.

Апаратні засоби та їх призначення

Вид операцій	Обладнання, яке використовується
Перевірка кабелю на відсутність фізичного обриву	Тестери
Діагностика кабельних систем	Мережеві екрани
Еталонне тестування кабелів різних категорій, сертифікація кабельних систем на відповідність певному стандарту	Мережеві аналізатори

Перш, ніж перейти до більш докладного огляду цих пристроїв, наведемо деякі необхідні відомості про основні електромагнітні характеристики кабельних систем [15].

Основні електромагнітні характеристики кабельних систем

Основними електричними характеристиками, що впливають на роботу кабелю, є: затухання, імпеданс (хвильовий опір), перехресні наводки двох кручених пар і рівень зовнішнього електромагнітного випромінювання.

Перехресні наводки між витими парами або NearEndCrosstalk (NEXT) - являють собою результат інтерференції електромагнітних сигналів, що виникають у двох кручених парах. Один з кабелів крученої пари передає сигнал, а другий - приймає. При проходженні сигналу по одному з кабелів, наприклад, по тому, що передає, у кабелі, що приймає сигнал виникають перехресні наводки. Величина NEXT залежить від частоти переданого сигналу - чим вище величина NEXT, тим краще (для категорії 5 NEXT повинен бути не менше 27 Дб при частоті 100 Мгц, для кабелю категорії 3 на частоті 10 МГц NEXT повинен бути не менше 26 Дб).

Затухання (Attenuation) - являє собою втрату амплітуди електричного сигналу при його поширенні по кабелю. Затухання має два основних джерела: електричні характеристики кабелю і поверхневий ефект. Останній пояснює залежність затухання від частоти. Затухання вимірюється в децибелах на метр. Для кабелю категорії 5 при частоті 100 Мгц загасання не повинно перевищувати 23.6 Дб на 100 м, а для кабелю категорії 3, за стандартом IEEE 802.3 10BASE-T, допустима величина затухання на сегменті довжиною 100 м не повинна перевищувати 11,5 Дб при частоті змінного струму 10 МГц.

Імпеданс (хвильовий опір) - це повний (активне і реактивне) опір в електричному ланцюзі. Імпеданс вимірюється в омах і є відносно постійною

величиною для кабельних систем. Для неекранованої крученої пари найбільш часті значення імпедансу - 100 і 120 Ом. Характерні значення імпедансу для мереж стандарту Ethernet на коаксіальному кабелі становлять 50 Ом, а для мереж стандарту Arcnet - 93 Ом. Різкі зміни імпедансу по довжині кабелю можуть викликати процеси внутрішнього відображення, що призводять до виникнення стоячих хвиль. Стояча хвиля — тип коливань у неперервному середовищі, при яких кожна точка середовища здійснює періодичний рух зі сталою амплітудою, залежною від її положення. Стоячі хвилі не переносять енергію. Робоча станція, підключена до кабелю у районі вузла стоячої хвилі, не зможе отримувати адресовані їй повідомлення.

Активний опір - це опір постійному струму в електричному ланцюзі. На відміну від імпедансу активний опір не залежить від частоти і зростає зі збільшенням довжини кабелю. Для неекранованої крученої пари категорії 5 активний опір не повинен перевищувати 9.4 Ом на 100 м.

Ємність - це властивість металевих провідників накопичувати енергію. Два електричних провідника в кабелі, розділені діелектриком, являють собою конденсатор, здатний накопичувати заряд. Ємність є небажаною величиною, тому її слід робити якомога менше. Високе значення ємності в кабелі приводить до спотворення сигналу і обмежує смугу пропускання лінії. Для кабельних систем категорії 5 значення ємності не повинен перевищувати 5.6нФ на 100 м.

Рівень зовнішнього електромагнітного випромінювання, або електричний шум - це небажана зміна напруги в провіднику. Електричний шум буває двох типів: фоновий і імпульсний. Електричний шум можна також розділити на низько-, середньо-і високочастотний. Джерелами фонового електричного шуму є в діапазоні до 150 КГц лінії електропередачі, телефони і лампи денного світла; в діапазоні від 150 КГц до 20 Мгц комп'ютери, принтери, ксерокси; в діапазоні від 20 МГц до 1 ГГц – теле- і радіопередавачі, мікрохвильові печі. Основними джерелами імпульсного електричного шуму є мотори, перемикачі і зварювальні

агрегати. Електричний шум вимірюється в мВ. Кабельні системи на крученій парі не сильно схильні до впливу електричного шуму (на відміну від впливу NEXT).

Мережеві аналізатори

Мережеві аналізатори (не слід плутати їх з аналізаторами протоколів) являють собою еталонні вимірювальні інструменти для діагностики та сертифікації кабелів і кабельних систем. Як приклад можна привести мережеві аналізатори компанії HewlettPackard - HP 4195A і HP 8510C. Мережеві аналізатори містять високоточний частотний генератор і вузькосмуговий приймач. Передаючи сигнали різних частот в передавальну пару і вимірюючи сигнал у приймальній парі, можна виміряти затухання і NEXT. Мережеві аналізатори - це великогабаритні і дорогі прилади, призначені для використання в лабораторних умовах спеціально навченим технічним персоналом.

Кабельні сканери

Дані прилади дозволяють визначити довжину кабелю, NEXT, затухання, імпеданс, схему розводки, рівень електричних шумів і провести оцінку отриманих результатів. Існує досить багато пристроїв даного класу, наприклад, сканери компаній MicrotestInc., FlukeCorp., DatacomTechnologiesInc., ScopeCommunicationInc. На відміну від мережевих аналізаторів сканери можуть бути використані не тільки спеціально навченим технічним персоналом, але навіть адміністраторами-новачками.

Для визначення місця розташування несправності кабельної системи (обриву, короткого замикання, неправильно встановленого роз'єму і т.д.) використовується метод "кабельного радара", або TimeDomainReflectometry (TDR). Суть цього методу полягає в тому, що сканер випромінює в кабель короткий електричний імпульс і вимірює час затримки до приходу відбитого

сигналу. За полярності відображеного імпульсу визначається характер пошкодження кабелю (коротке замикання або обрив). У правильно встановленому і підключеному кабелі відбитий імпульс зовсім відсутній.

Точність вимірювання відстані залежить від того, наскільки точно відома швидкість розповсюдження електромагнітних хвиль у кабелі. У різних кабелях вона буде різною. Швидкість розповсюдження електромагнітних хвиль у кабелі (NVP) зазвичай задається у відсотках до швидкості світла у вакуумі. Сучасні сканери містять в собі електронну таблицю даних про NVP для всіх основних типів кабелів і дозволяють користувачеві встановлювати ці параметри самостійно після попереднього калібрування.

Найбільш відомими виробниками компактних кабельних сканерів є компанії MicrotestInc., WaveTekCorp., ScopeCommunicationInc.

Тестери

Тестери кабельних систем - найбільш прості і дешеві прилади для діагностики кабелю, які дозволяють визначити пошкодження кабеля, проте, на відміну від кабельних сканерів, не можуть визначити в якому місці стався збій.

2.2.3. Програмно-апаратні рішення

Головними ініціаторами переходу до відкритих мереж стали такі компанії, як Google, Amazon, Facebook і Microsoft. Вони вже досить давно оцінили всі ті переваги, в першу чергу зниження витрат, які дає розробка власних серверів і їх виробництво тайванськими компаніями (такі продукти отримали назву «white box»). Реалізація аналогічної моделі стосовно до мережного обладнання стала лише питанням часу[16].

Традиційно основні вузли мережевих інфраструктур, комутатори і маршрутизатори, постають для замовників у наступному вигляді: пропрієтарне обладнання, фірмова мережева операційна система, вбудований виробником набір функцій, та спеціалізоване ПЗ.

Рис.2.1. Запропоновані рішення типу “white box”

Розробка інтернет-гігантами власного мережевого обладнання (на фото зліва - модульний комутатор Facebook, рис.2.1) підштовхнула виробників традиційних рішень до того, щоб зробити свої платформи відкритими (на фото праворуч - комутатор Dell Networking серії ON - від Open Networks, рис.2.1).

На відміну від вертикально інтегрованих пропрієтарних мейнфреймів, сучасні обчислювальні рішення вже кілька десятиліть є дезагрегірованнними. Замовники можуть набувати апаратні платформи і сервери як у відомих брендів (Dell, HP, Lenovo), так і у постачальників «white-box» (Quanta Computers, Super Micro). При цьому операційні системи і додатки можуть закуповуватися окремо у інших постачальників і інсталюватися на апаратній платформі. Подібна дезагрегація стимулювала інновації на всіх рівнях такого IT-стека: в процесорах і інших апаратних компонентах, операційних системах і додатках.

Разом з тим в частині архітектури пристроїв мережева галузь зупинилася в своєму розвитку на довгі роки десь з початку 90-х років минулого століття[17]. Програмне забезпечення для мережевих пристроїв, власне апаратну платформу, включаючи спеціалізовані набори мікросхем ASIC, традиційно поставляли вертикально інтегровані вендори, що не сприяло зниженню цін і гальмувало інновації. Але ситуація почала швидко змінюватися з проникненням в мережевий світ ідей програмованих мережевих інфраструктур (SDN), а додатковими каталізаторами виступили такі продукти, як bare metal, white box і brite box:

2.3. Порівняльна характеристика рішень в сфері моніторингу Інтернет-трафіку користувачів

Аналіз трафіку є процесом, важливість якого відома будь-якому IT-професіоналу, не залежно від того, чи працює він в невеликій компанії або у великій корпорації. Адже виявлення і виправлення проблем з мережею - це складний процес, яке безпосередньо залежить як від навичок самого фахівця, так і від глибини і якості оперованих їм даних. І аналізатор трафіку є саме тим інструментом, який ці дані надає[18].

Зараз на ринку представлена велика кількість варіацій програмного забезпечення для аналізу мережевого трафіку. Деякі використовують термінальний шрифт і інтерфейс командного рядка, і на перший погляд здаються складними у використанні. Інші рішення, навпаки, - виділяються простотою встановлення і орієнтовані на аудиторію з візуальним сприйняттям. Ціновий діапазон цих рішень також має велике значення відрізняється - від безкоштовних до рішень з дорогою корпоративною ліцензією[19].

2.3.1. Системи моніторингу трафіку

Спільнота Nagios (https://www.nagios.org/), що веде свою історію з 1999 року, є одним з лідерів галузі в області рішень для моніторингу IT-інфраструктури будь-якого масштабу - від малого до корпоративного рівня.

Програмне рішення для моніторингу комп'ютерних систем і мереж Nagios здатне здійснювати моніторинг практично будь-яких компонентів, включаючи мережеві протоколи, операційні системи, системні показники, додатки, служби, веб-сервера, веб-сайти, сполучна програмне забезпечення (Middleware) .

Базова функціональність системи для моніторингу (див.рис.2.2) Nagios реалізована на ядрі Core 4, який забезпечує високий рівень продуктивності за рахунок меншого споживання ресурсів сервера.

Можливо інтегрувати плагін практично з будь-яким типом стороннього

програмного забезпечення.

При використанні сполучного ПЗ (Middleware), ви можете використовувати Nagios для моніторингу WebLogic, WebSphere, JBoss, Tomcat, Apache, URL, Nginx і т. Д ..

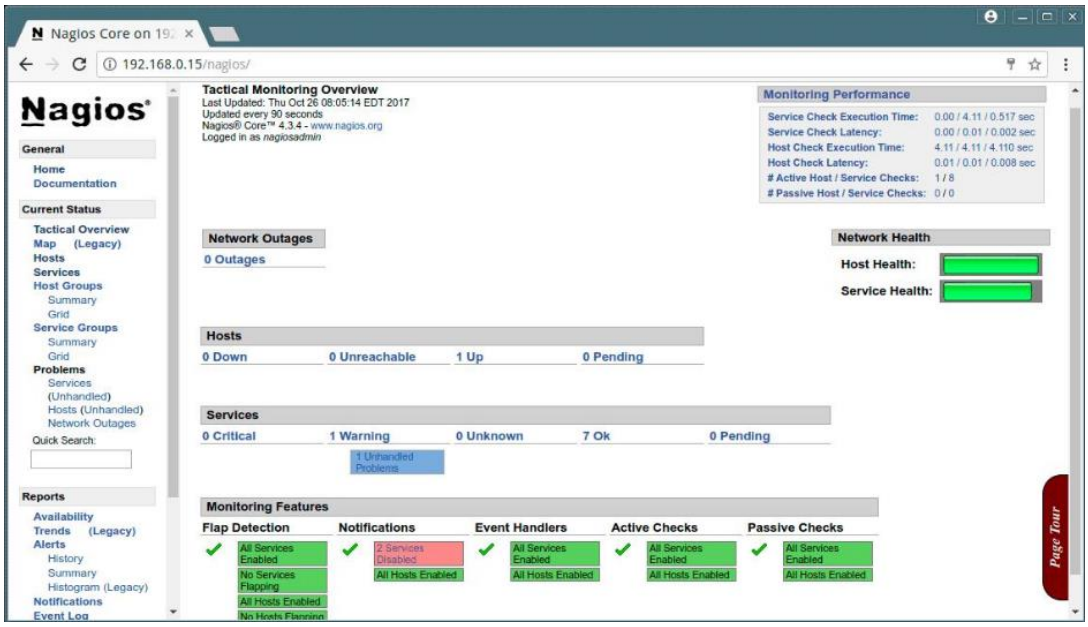

Рис. 2.2. Приклад інтерфейсу СМ Nagios

До переваг входить:

- Централізоване бачення всій контрольованій IT-інфраструктури.
- Автоматичний перезапуск додатків, здійснюваний оброблювачем подій, якщо в роботі цих додатків виявлений збій.
- Розрахований на багато користувачів доступ.
- Обмежений доступ дозволяє управляти видимістю для користувачів тільки тими компонентами IT-інфраструктури, які безпосередньо пов'язані з їх зоною відповідальності.
- Спільнота Nagios налічує понад 1 млн. Активних користувачів.
- Розширювана архітектура.

Cacti

Додаток для моніторингу мережі Cacti (https://www.cacti.net/) - це ще один програмний інструмент з відкритим вихідним кодом для моніторингу мережі, який може бути встановлений на Linux або Windows(див.рис.2.3). Він дозволяє відобразити збирані статистичні дані у графічному вигляді за допомогою набору утиліт RRDTool.

Cacti працює з SNMP і є мережевою статистику у вигляді простих для розуміння графіків.

Cacti потрібно MySQL, Apache або IIS з підтримкою PHP.

Короткий перелік доступних можливостей:

- Необмежена кількість елементів відображення графіків може бути задано, як через опцію CDEF (дозволяє застосовувати різні математичні функції до графіку для зміни вихідних даних), так і використовуючи шаблони графіків з Cacti.
- Підтримка автоматичного заповнення для графіків.
- Підтримка файлів RRD (Round-Robin Database, Циклічна база даних) з більш ніж одним джерелом даних, а також використання RRD-файлів, що зберігаються в будь-якому місці локальної файлової системи.
- Орієнтоване на користувача управління і безпеку.
- Скрипти для вибіркового збору даних користувача.

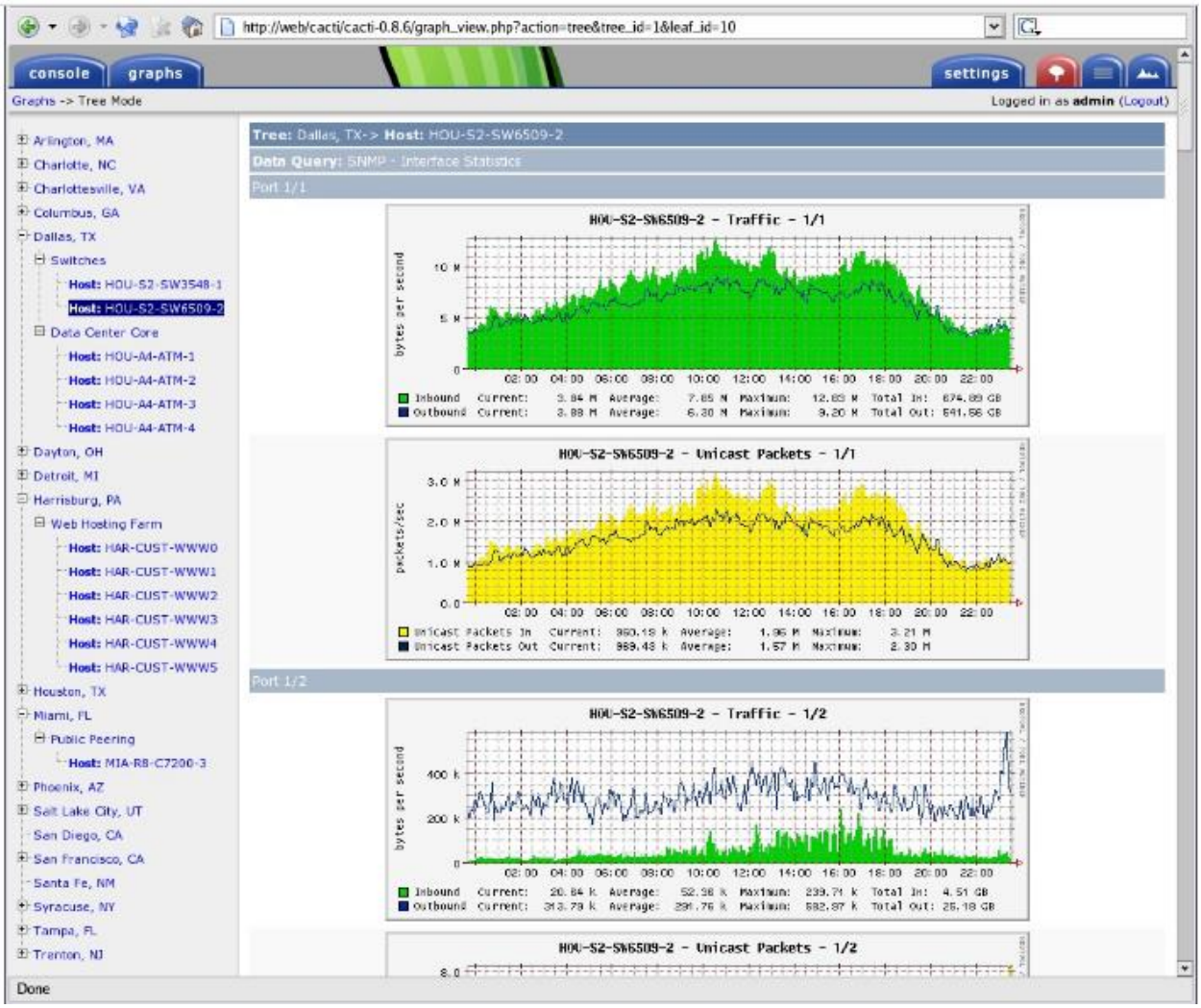

Рис. 2.3. Приклад інтерфейсу CM Cacti

OpenNMS

Високорівнева програмна платформа для моніторингу мереж і мережевих пристроїв OpenNMS (https://www.opennms.org/en) дозволить створити рішення мережевого моніторингу для будь-якої IT-інфраструктури промислового масштабу. Ви можете збирати системні показники за допомогою JMX, WMI, SNMP, NRPE, XML HTTP, JDBC, XML, JSON та ін.

За допомогою OpenNMS (див.рис.2.4) ви можете у вашій мережі, як виявляти зв'язку мережевих топологій на другому рівні моделі OSI, так і відстежувати неполадки в маршрутизації на рівні 3. Ця система моніторингу не використовує агентів, а побудована на подієво-орієнтованій архітектурі, а також підтримує роботу в зв'язці з системою агрегації даних і відображення графіків в

реальному часі Grafana.

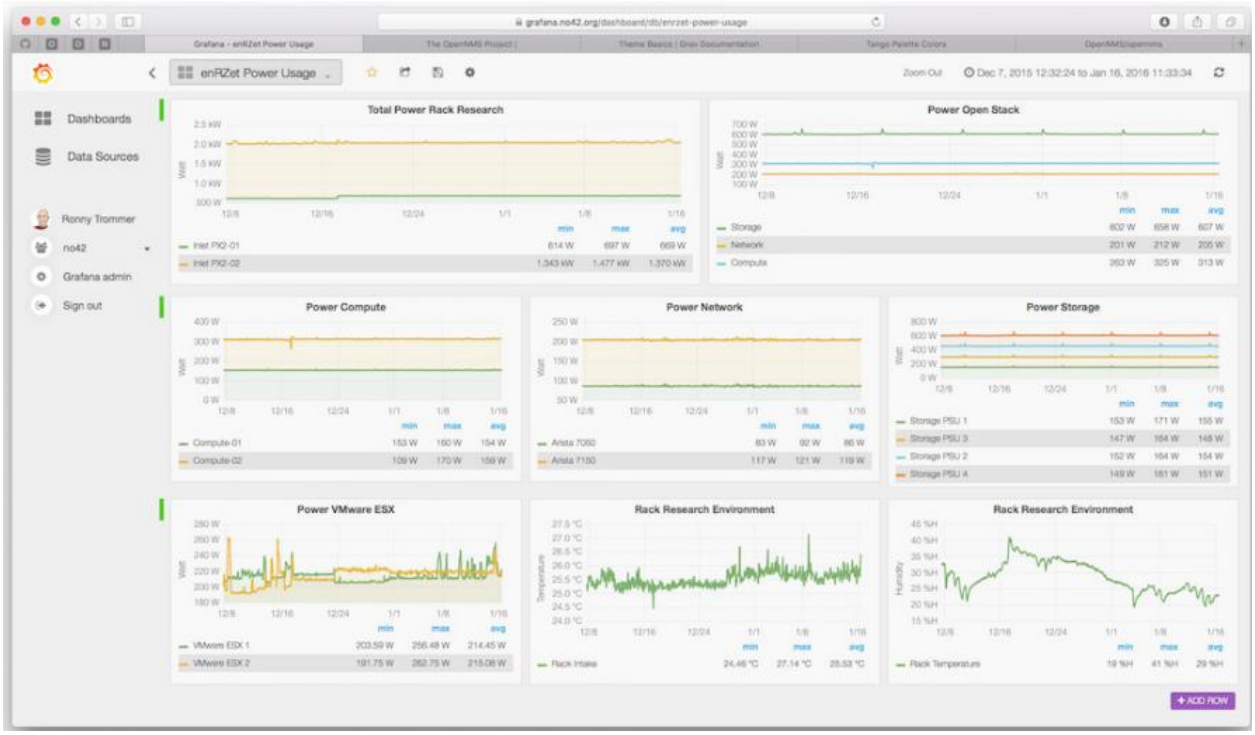

Рис. 2.4. Приклад інтерфейсу СМ OpenNMS

OpenNMS має вбудовані модулі формування звітності, а це означає, що ви можете переглядати звіти у вигляді красивих дашборда (dashboard, аналітичних інформаційних панелей) і діаграм. В цілому, OpenNMS отримав прекрасний користувальницький інтерфейс.

Ви також можете встановити OpenNMS в Docker - програмний інструментарій для управління ізольованими Linux-контейнерами.

Короткий перелік доступних можливостей:

- OpenNMS спеціально розроблявся для Linux, але також є реалізована підтримка Windows, Solaris і OSX.
- Моніторинг температури пристроїв.
- Налаштовуючи інформаційна панель адміністратора.
- Моніторинг електропостачання.
- Підтримка IPv4 і IPv6.

- Налаштування формування повідомлень про події та їх відправка по електронній пошті, СМС, XMPP (розширюваний протокол обміну повідомленнями та інформацією про присутність, раніше відомий як Jabber) і іншими способами.
- Географічна карта мережевих вузлів для відображення місця розташування «проблемних» вузлів і перебоїв у наданні послуг з використанням карт таких картографічних порталів, як Open Street Map, Google Maps або Mapquest.

Icinga

Безкоштовна програмна система для моніторингу комп'ютерних систем(див.рис. 2.5) і мереж Icinga (https://icinga.com/) дозволить вам здійснювати моніторинг усіх доступних систем у вашій мережі. Вона підтримує різні способи попереджень, а також надасть вам базу даних для ваших звітів про рівень обслуговування.

Icinga, історія якої розпочалася в 2009 році, як відгалуження від системи моніторингу Nagios, з виходом Icinga версії 2 змогла повністю звільнитися від «кайданів» ядра Nagios, ставши швидше, простіше в налаштуванні і значно краще масштабується.

Короткий перелік доступних можливостей:

- Моніторинг стану мережевих сервісів, серверних компонентів, а також принтерів, маршрутизаторів і т. Д.
- Здійснення моніторингу за допомогою плагінів Icinga 2.
- Підтримка обробників подій і створення повідомлень.
- Відправлення повідомлень електронною поштою, СМС, а також через різні служби миттєвих повідомлень.
- Кроссплатформенная підтримка різних операційних систем.
- Паралельні перевірки сервісів.
- Можливість вибору між класичним призначеним для користувача

інтерфейсом і веб.

- Формування звітів на основі шаблонів.

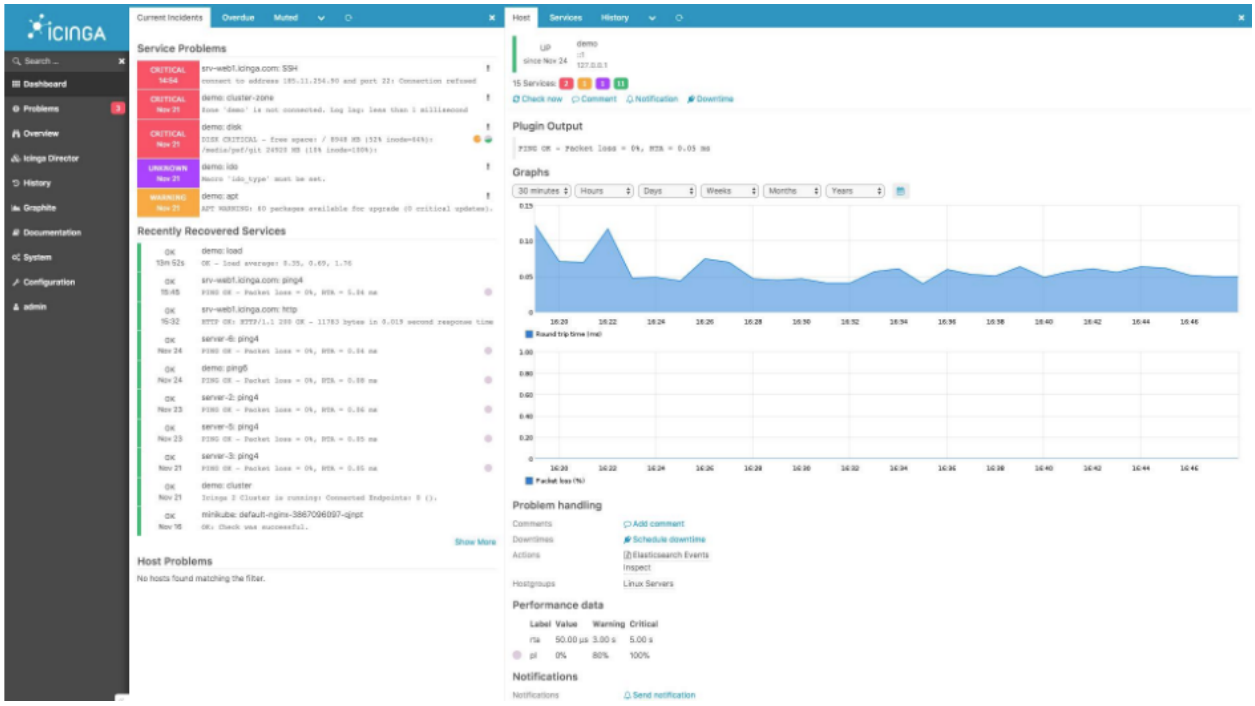

Рис. 2.5. Приклад інтерфейсу СМ Icinga

Prometeus

Prometheus (див. рис.2.6) - популярний CNCF-проект з відкритим вихідних кодом, велика частина компонентів якого написана на Golang, а частина - на Ruby. Це означає, що у вас буде всього один бінарний файл, який потрібно завантажити і запустити разом з компонентами Prometheus.

Як відомо, це безкоштовне програмне забезпечення, яке використовується для моніторингу подій та сповіщення. Він записує показники в реальному часі в базу даних часових рядів, побудовану за допомогою моделі HTTP pull, з гнучкими запитами та сповіщеннями в режимі реального часу.

Prometheus - це база даних часових рядів. До нього можна приєднати цілу екосистему інструментів, щоб розширити функціонал. Prometheus моніторить найрізноманітніші системи: сервери, бази даних, окремі віртуальні машини,

діяльність користувачів та ін.

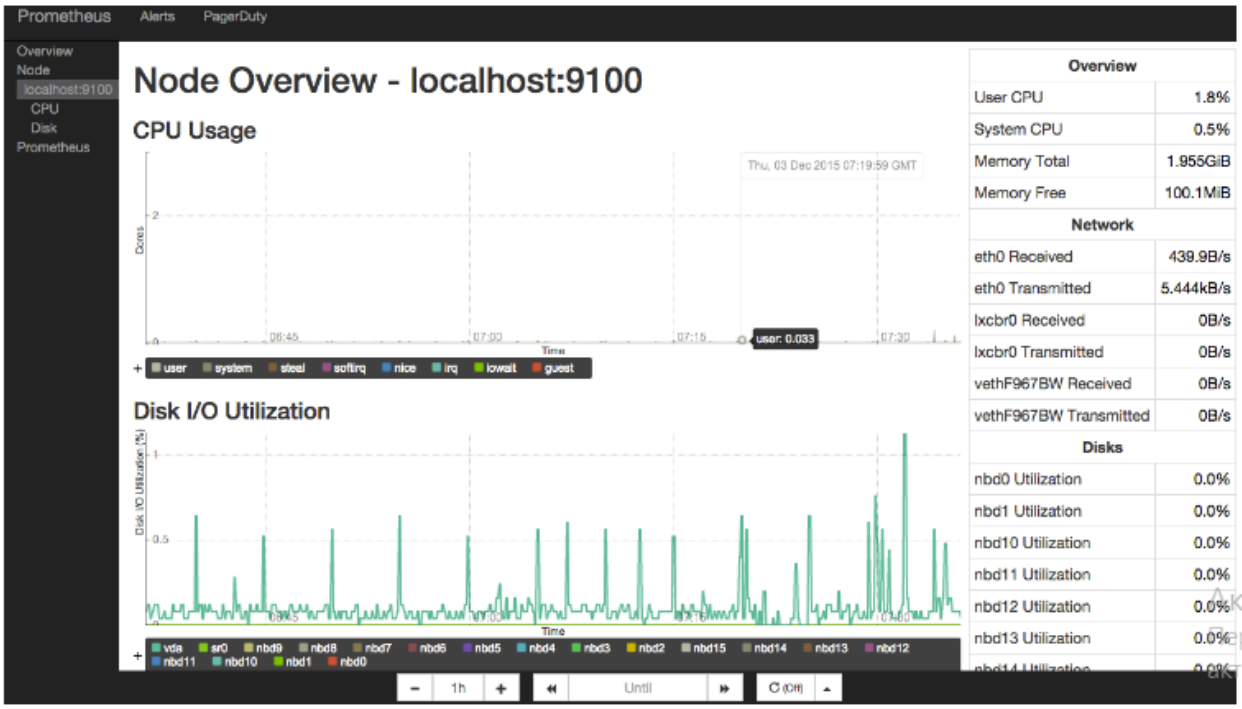

Рис. 2.6. Приклад інтерфейсу CM Zabbix

SolarWinds Network Bandwidth Analyzer

Дане рішення позиціонується виробником як програмний пакет з двох продуктів - Network Performance Monitor (базове рішення) і NetFlow Traffic Analyzer (модульне розширення). Вони мають схожі риси, але все ж відрізняються функціональні можливості для аналізу мережевого трафіку, що доповнюють один одного при спільному використанні відразу двох продуктів. Network Performance Monitor, як випливає з назви, здійснює моніторинг продуктивності. Купуючи це рішення, адміністратор отримує можливість контролювати загальну працездатність вашої мережі: спираючись на величезну кількість статистичних даних, таких як швидкість і надійність передачі даних і пакетів, в більшості випадків ви зможете швидко ідентифікувати несправність в роботі вашої мережі. А добре оптимізований штучний інтелект виявлює потенційні проблем і широкі можливості по візуальному представленню

результатів у вигляді таблиць і графіків з чіткими попередженнями про можливі проблеми.

Модульне розширення NetFlow Traffic Analyzer більше сконцентровано на аналізі самого трафіку. У той час, як функціональність базового програмного рішення Network Performance Monitor більше призначена для отримання загального уявлення про продуктивність мережі, в NetFlow Traffic Analyzer фокус уваги спрямований на більш детальний аналіз процесів, що відбуваються в мережі. Зокрема, ця частина програмного пакета дозволить проаналізувати перевантаження або аномальні скачки смуги пропускання і надасть статистику, відсортовану по користувачах, протоколів або додатків. Дана програма доступна тільки для середовища Windows.

Wireshark

Безкоштовний open-source аналізатор трафіку WireShark надає своїм користувачам неймовірно просунутий функціонал і по праву визнаний зразковим рішенням в області мережевої діагностики. Він ідеально інтегрується з системами на базі * NIX / Windows / macOS.

Інтерфейс не інтуїтивний, проте дане рішення використовує GUI (у разі потреби, модернізувати набір стандартних можливостей за допомогою Lua).

Розгорнувши і налаштувавши його одного разу на своєму сервері, бізнес отримає централізований елемент для моніторингу за найдрібнішими змінами в роботі мережі і мережевих протоколах. Таким чином, це ПЗ вміє на ранніх етапах виявляти і ідентифікувати проблеми, що виникають в мережі.

Tcpdump

Аналізатор трафіку tcpdump має застарілий інтерфейс і погано адаптується під потреби великого бізнесу. Це сніфер (утиліта UNIX), яка захоплює та аналізує мережевий трафік.

Основні призначення tcpdump:

- Налагодження мережевих програм
- Налагодження мережі і мережної конфігурації загалом

Програма складається з двох частин: захоплення пакетів (звернення до бібліотеки, libcap (Linux) або pcap (Windows)) і відображення захоплених пакетів (ця частина на рівні вихідного коду є модульною і для підтримки нового протоколу досить додати новий модуль).

Частина захоплення пакетів передає "вираз вибору пакетів" (що йде після всіх параметрів командного рядка) безпосередньо бібліотеці захоплення пакетів, яка перевіряє правильність синтаксису виразу, компілює його, а потім копіює у внутрішній буфер програми мережні пакети, які проходять через вибраний мережевий інтерфейс і задовольняють умовам у виразі. Частина відображення пакетів по черзі вибирає захоплені пакети з внутрішнього буфера, і виводить їх на стандартний вивід, згідно із заданим рівнем деталізації. Якщо задано докладний вивід пакетів, програма перевіряє для кожного мережевого пакету, чи є модуль розшифровки даних, і, у разі наявності, відповідною підпрограмою витягує (і відображає) тип та інші параметри пакету в протоколі.

Kismet

Це вільний мережевий аналізатор для бездротових мереж стандарту 802.11b. Дозволяє прослуховувати трафік за допомогою практично будь-яких підтримуваних бездротових мережевих адаптерів, що використовують драйвери Airo, HostAP, Wlan-NG або Orinoco (з латкою для ядра).

Аналізатор складається з сервера та клієнта і може використовуватись для аналітики роботи мереж (зібрані дані мають оброблятись сторонніми додатками) та їх захисту (як детектор зовнішніх атак на мережі). Адаптований до програм SoX і Festival для програвання оповіщень про мережеві події і промовляння короткого опису при їх виявленні. Додатково може використовувати службу gpsd системи GPS для прив'язки до місцевості.

EtherApe

За своїми функціональними можливостями EtherApe багато в чому наближається до WireShark, і він також є програмним забезпеченням з відкритим вихідним кодом і розповсюджується безкоштовно. Однак те, чим він дійсно виділяється на тлі інших рішень - це орієнтація на графіку. І якщо результати аналізу трафіку WireShark можливо переглянути в класичному цифровому вигляді, то мережевий трафік EtherApe відображається за допомогою графічного інтерфейсу, де кожна вершина графа являє собою окремий хост, розміри вершин і ребер вказують на розмір мережевого трафіку, а кольором відзначаються різні протоколи. Для бізнесу, який віддає перевагу візуальному сприйняттю статистичної інформації, аналізатор EtherApe може стати найкращим вибором. Доступний для середовищ * NIX і macOS.

Cain and Abel

У даного програмного забезпечення можливість аналізу трафіку є скоріше допоміжною функцією, ніж основною. З його допомогою можливо відновлювати паролі для ОС Windows, виробляти атаки для отримання втрачених облікових даних, вивчати дані VoIP в мережі, аналізувати маршрутизацію пакетів і багато іншого. Це дійсно потужний інструментарій для системного адміністратора з широкими повноваженнями. Працює тільки в середовищі Windows.

NetworkMiner

Рішення NetworkMiner - ще одне програмне рішення, чия функціональність виходить за рамки звичайного аналізу трафіку. У той час як інші аналізатори трафіку зосереджують свою увагу на відправку та отримання пакетів, NetworkMiner стежить за тими, хто безпосередньо здійснює цю відправку та отримання. Цей інструмент більше підходить для виявлення проблемних комп'ютерів або користувачів, ніж для проведення загальної діагностики або моніторингу мережі як такої. NetworkMiner розроблений для ОС Windows.

Проактивний моніторинг - моніторинг, під час якого ведеться пошук закономірностей у подіях для передбачення можливих майбутніх збоїв[20].

Реактивний моніторинг - моніторинг, який виконується у відповідь на подію. Наприклад, запуск пакетного завдання після виконання попередньої задачі або реєстрація інциденту виникненні помилки.

Таблиця 2.2

Порівняння існуючих рішень моніторингу

ПЗ	Вид моніторингу	User-friendly	Додаткові засоби графічного представлення	Журнал подій	Повний аналіз пакетів	Віддалені команди	Ліцензія	Тип моніторингу	Вимогливий до ресурів	Фільтрація результатів
Zabbix	Проактивний	+	+	+	+	+	-	Нейромережевий	+	+
Wireshark	Реактивний	-	+	-	+	-	-	Скінченний автоматний	-	-
tcpdump	Реактивний	-	+	-	+	-	-	Скінченний автоматний	-	-
Cacti	Реактивний	+	-	+	-	+	-	Частотний	-	-
EtherApe	реактивний	-	-	+	+	-	-	Скінченний автоматний	-	-

Cisco Appdynamics	Реактивний	+	-	+	-	+	+	Нейромережевий, паатерновий (шаблони поведінки)	+	+
Icinga	Проактивний	+	+	+		+	-	Нейромережевий, частотний	-	+

З таблиці 2.2 можемо зробити висновки що найкраще рішення це Zabbix. Цей продукт безкоштовний, легко адаптується, проте вимагає чимало ресурсів.

2.4. Висновки до розділу 2

Таким чином, підсумовуючи викладене в розділі, можна зробити наступні висновки:

1) Розглянуто проблему виявлення атак у комп’ютерних системах та існуючі методи для систем виявлення аномалій. По кожному методу та його моделі проаналізовано переваги та недоліки. Запропоновано альтернативні шляхи створення ефективної системи виявлення аномалій в комп’ютерних системах

2) Управління продуктивністю мережі допомагає моніторингу мережевого трафіку, для перегляду, аналізу та управління мережевим трафіком на предмет будь-яких відхилень. Аналізатор мережевого трафіку це процес, який може вплинути на продуктивність, доступність і / або безпеку мережі. Монітор мережевого трафіку використовує різні інструменти і методи для вивчення мережевого трафіку вашого комп'ютера .Для моніторингу трафіку існує безліч рішень, реалізація яких залежить від ТЗ.

3) Програми для аналізу мережевого трафіку можуть стати життєво важливим інструментарієм, коли бізнес періодично стикаєтеся з мережевими проблемами різних видів - будь то продуктивність, скинуті сполуки або проблеми з мережевими резервними копіями. Практично все, що пов'язано з передачею і отриманням даних в мережі, може бути швидко ідентифіковано і виправлено завдяки даним, отриманим за допомогою програмного забезпечення з вищенаведеного списку. З даних зведеної таблиці можемо зробити висновки що найкраще рішення це Zabbix. Цей продукт безкоштовний, легко адаптується , проте вимагає чимало ресурсів.

РОЗДІЛ 3. РОЗРОБКА АЛГОРИТМУ ТА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ МОНІТОРИНГУ ТА КОНТРОЛЮ ІНТЕРНЕТ-ТРАФІКУ КОРИСТУВАЧІВ

3.1. Формулювання вимог до програмного забезпечення моніторингу та контролю Інтернет-трафіку

Проблеми в роботі мережі можуть значно погіршувати якість обслуговування користувачів, знижуючи ступінь їх задоволення мережевими сервісами і породжуючи невдоволення тими, хто надає ці послуги. Тому вкрай важливо максимально швидко виявляти, діагностувати і усувати проблеми. Різноманітні системи мережевого моніторингу та діагностичні засоби прискорюють процес виявлення та аналізу проблем і тим самим сприяють скороченню періоду часу між появою проблеми і її усуненням. Більш того, збираючи та аналізуючи інформацію про роботу мережі, засоби моніторингу дозволяють виявляти можливі проблеми і не допускати їх виникнення[21].

Більшість експертів схиляються до того, що фаєрволи, системи безпеки і аналіз вхідного і вихідного трафіку повинні бути фізично рознесені. Не можна покладатися на логи з брандмауера, який зберігає всі записи про трафік, тому що під час атаки (хакерів або вірусу) він буде недоступний і не надасть ніякої інформації[22].

Виділена система запису і аналізу трафіку завжди надасть додаткову і цінну інформацію і найголовніше, що ви зможете аналізувати трафік як в реальному часі, так і за будь-який період (обмежена обсягом сховища і шириною каналів зв'язку). Це дозволить бачити, як відбувається атака і що відбувається з вихідним трафіком з корпоративної мережі, запобігаючи витік даних[23].

Основні вимоги, яким повинна відповідати система аналізу трафіку:

- Гарантія запису повного пакету (НЕ метадані).
- Гарантія записи на повній швидкості каналу (наприклад, системи

мають 8-10 гігабітний інтерфейс, а реально пишуть дані на швидкості не вище 5-7 Гбіт / сек).

- Можливість зберігання пакетів в часі, без модифікацій даних.
- Можливість побудови звітів, трендів, алярмів чи повідомлень SNMP, Syslog, e-mail і т.д.
- Можливість розширення сховища для збільшення часу запису трафіку.
- Вибір і керування системою має відбуватися заздалегідь, а не в момент витоку.

Також, вимоги розрізняються за джерелами (див.3.1):

Рис. 3.1. Висунуті вимоги до програмного забезпечення

Окрім цього, вимоги розподіляють на : функціональні; інтерфейсні; до продуктивності;специфічні; характеристики якості; інші.

3.2. Розробка алгоритму роботи програмного забезпечення

Програма виконує аналіз трафіку користувачів нейромережевим методом.

Програмний модуль реалізує алгоритм балансування навантаження за допомогою якого контрольований трафік розподіляється по вихідним портам

10G (призначеним для підключення зовнішніх ресурсів моніторингу, у описі тестування результати обробляє Zabbix) зі збереженням цілісності потоків пакетів. Важливим переваг цього алгоритму є можливість обмеження швидкості передачі трафіку засобу моніторингу, щоб не допустити його перевантаження.

Алгоритм обробки пакетів базується на формулі

$$x(t) = (x_1(t), x_2(t), \dots x_n(t)),$$

Де t — це час, а n — кількість вимірюваних параметрів.

Є декілька методів :

- Точковий – використовується тоді, коли окремий екземпляр даних може розглядатися як аномальний по відношенню до основних даних;
- Коллективний - коли сукупність вимірювань аномальна щодо всього набору даних, навіть коли кожне вимір не вибивається із загальної картини;
- Контекстуальний - в цьому випадку аномалія визначається з урахуванням контексту.

Для моніторингу користувацького трафіку найбільш оптимальний точковий метод виявлень аномалій. Головною ідеєю пошуку аномалій служить метод зворотнього поширення помилки [24].

Архітектура. Див. 3.2:

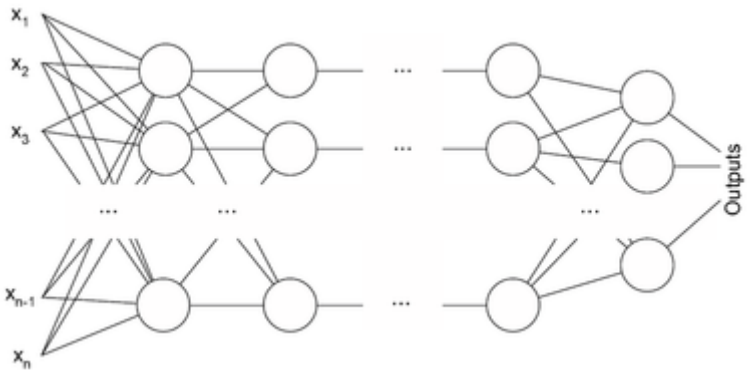

Рис. 3.2 Представлення архітектури алгоритму

Алгоритм:

$$(\eta, \alpha, \{x_i^d, t^d\}_{i=1,d=1}^{n,m}, NUMBER\_OF\_STEPS)$$

1. Ініціалізувати $\{\omega_{ij}\}_{i,j} = 0$ маленькими випадковими значеннями, $\{\Delta\omega_{ij}\}_{i,j} = 0$

2. Повторити п.1:

Для всіх d від 1 до m:

1. Подати $\{x_i^d\}$ на вхід сітки і підрахувати виходи кожного вузла.
2. Для всіх $k \in Outputs$ $\delta_k = o_k(1 - o_k)(t_k - o_k)$.
3. Для кожного рівня l, починаючи з останнього:

Для кожного вузла j рівня l порахувати

$\delta_j = o_j(1 - o_j) \sum_{k \in Children(j)} \delta_k \omega_{j,k}$

4. Для кожного ребра сітки {i, j}

$$\Delta\omega_{i,j} = \alpha\Delta\omega_{i,j} + (1 - \alpha)\eta\delta_j o_i$$

$$\omega_{i,j} = \omega_{i,j} + \Delta\omega_{i,j}$$

Видати значення $\omega_{i,j}$.

Компонент збору даних збирає дані про дії користувачів для навчання системи безпеки. Після збору даних навчання компонент підготовки даних готує дані для навчання моделі шляхом застосування різних фільтрів. Потім обраний алгоритм застосовується до підготовленим даними тренінгу для навчання моделі виявлення атаки. Час, що витрачається алгоритмом на навчання моделі реакції та ту чи іншу загрозу, варіюється від алгоритму до алгоритму. Як тільки модель буде навчена, вона буде перевірена, щоб дослідити, чи може модель виявляти загрози рівня додатку. Дані тестування фільтруються через компонент підготовки даних і подаються в модель виявлення атаки, яка аналізує дані для виявлення атак на основі правил, отриманих на етапі навчання. Результат аналізу даних відображається користувачеві через за допомогою Zabbix.

Алгоритм реалізації:

Існує чимало інструментальних засобів, які використовуються для реалізації проекту ІС від етапу аналізу до створення програмного коду. Є CASE-засоби верхнього рівня (upper CASE tools) і нижнього рівня (lower CASE tools).

Серед основних проблем використання CASE-засобів верхнього рівня виділяють проблеми їх адаптації під конкретні проекти, тому що вони жорстко регламентують процес розробки і не дають можливості організувати роботу на рівні окремих елементів проекту. Альтернативою їм може стати використання CASE-засоби нижнього рівня, але їх використання спричиняє інші проблеми – труднощі в організації взаємодії між командами, що працюють над різними елементами проекту[25].

Засобом, що дозволяє об’єднати ці підходи, є уніфікована мова об’єктно-орієнтованого моделювання (Unified Modeling Language – UML). До переваг UML відносяться різноманітні інструментальні засоби, які підтримують життєвий цикл ІС та дозволяють налаштувати і відображати специфіку діяльності розробників різних елементів проекту[26].

Клас – це опис набору об'єктів з однаковими атрибутами, операціями, зв'язками і семантикою. Кожен клас повинен володіти ім'ям, який вирізняє його від інших класів. Ім'я – це текстовий рядок. Ім'я класу може складатися з будь-якого числа букв, цифр і розділових знаків (за винятком двокрапки і крапки) і може записуватися в кілька рядків.

Атрибут (властивість) – це іменована властивість класу, яка описує діапазон значень, які може приймати примірник атрибута. Клас може мати будьяке число атрибутів або не мати жодного. В останньому випадку блок атрибутів залишають порожнім. Атрибут представляє деяку властивість сутності, що моделюється, якою володіють всі об'єкти даного класу. Ім'я атрибута, як і ім'я класу, може являти собою текст. На практиці для іменування атрибута використовуються одне або кілька коротких іменників, що виражають

якесь властивість класу, до якого належить атрибут. Атрибути можуть бути такими, типи значень яких вважаються заздалегідь визначеними в UML, як: розмір, площа, кут, видимість. Спроектована діаграма класів для програмної системи наведена на рис. 3.3

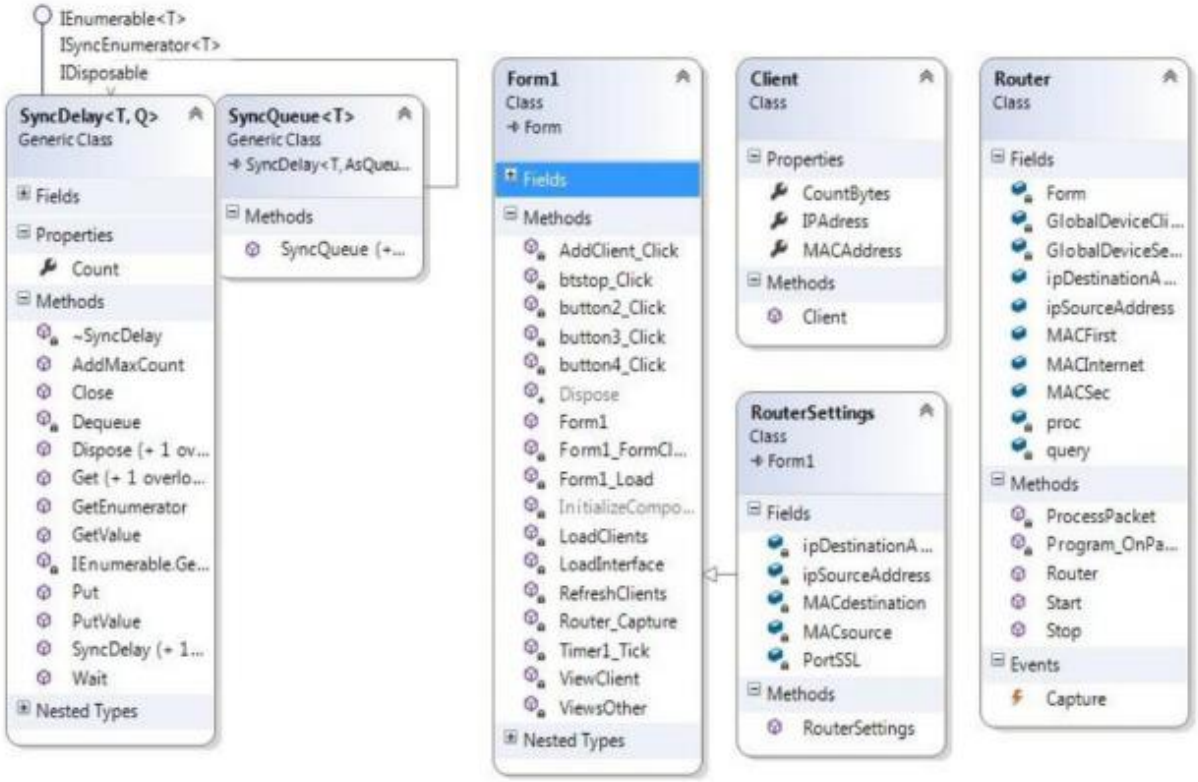

Рис. 3.3 Діаграма класів

В ході роботи виділено такі класи:

Form1 – клас екранної форми;

Client – клас відомостей про клієнтів;

Router – клас відповідає за захоплення і обробку пакетів;

CyncDelay, SyncQueue – класи роботи з чергами.

Алгоритм варіантів використання

Прецедент - еліпс з написом, який відображає виконувані системою дії, що

призводять до спостережуваних акторами результатами.

В даному дипломному проекті для засобу прогнозування на основі нейронної мережі було вирішено використовувати двух акторів:

– User (користувач програмного забезпечення).

– System (розроблений додок).

На рис. 3.4 представленні актори та їх прецеденти. Для більш детального розуміння розробленого програмного засобу розглянемо кожен прецедент окремо. Почнемо з можливостей користувача (User).

1) Внесення даних (користувач може завантажити набір даних та розпочати системну обробку на основі нейромережі).

2) Параметри мержі та навчання (користувач може додати або замінити параметри для нейронної мержі).

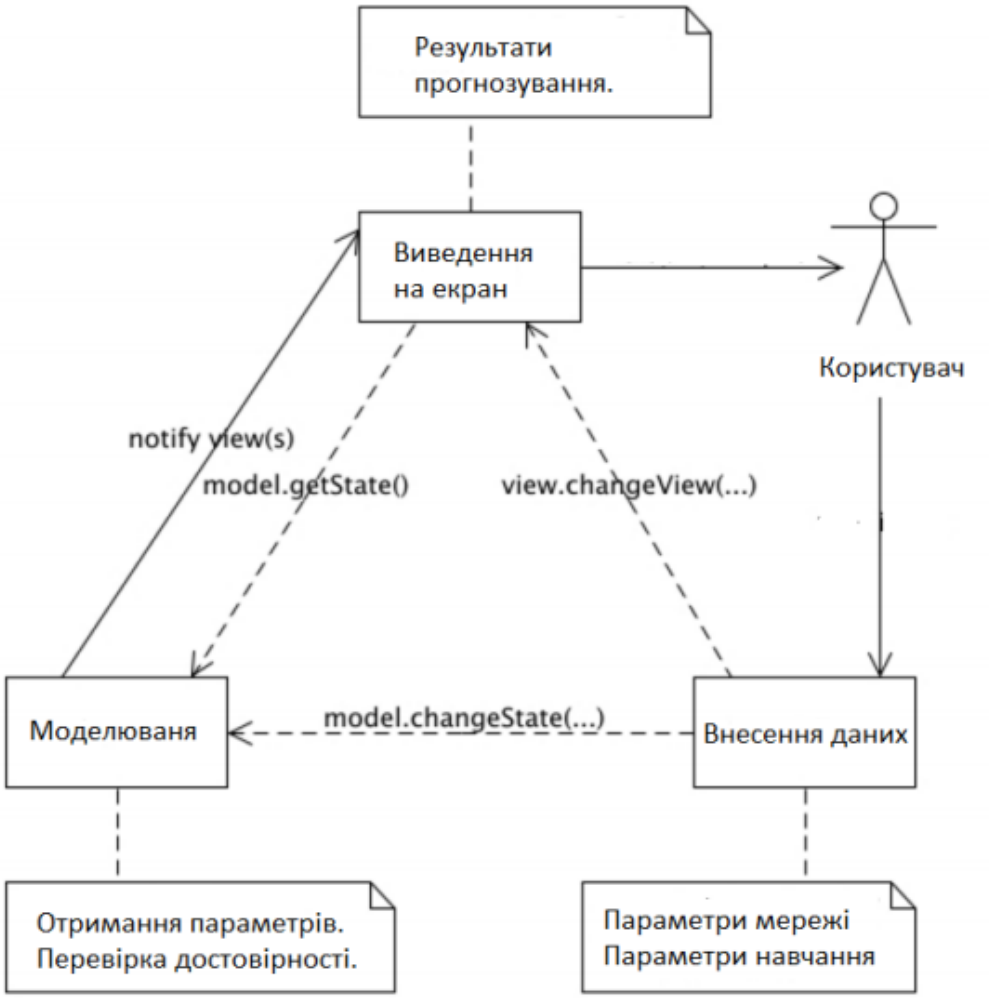

Рис. 3.4. Алгоритм варіантів використання

Так як, другим актором запропонованої системи є сам розроблений модуль проаналізуємо його можливості.

1) Моделювання (формується набір даних із запропонованих для обробки та набір даних для нейронної мережі).

2) Отримання параметрів та перевірка достовірності (відбувається обробка сформованих даних та їх конвертація в вид набору даних трафіку за допомогою прецидента Convert, який належить до виконанню Pre-Process. Нейромережа починає робити обробку даних на основі запропонованої моделі в прецендентi Binary format. Normalization дозволяє отримати дані після обробки у зрозумілому вигляді).

3) Вивдення на екран (використовується для показання користувачу

прогнозованих даних, для розуміння запропонованих дат та їх щогодиного використання трафіку).

4) Прогнозування результатів (прогнозування результатів є та сама обробка, що вибиває інформаціє на дисплей, то відтасовує прогнозовані дані від вже існуючих). Передачі класів описані за допомогою стрлочок та параметрів (див. рис. 3.4).

Алгоритм перевірок

Функціонування об'єкта в даній реалізації моделі побудовано навколо функції DoLive (), в якій по черзі відбуваються такі життєві етапи об'єкта:

RefreshSense - оновлення даних

Move – перехід до наступного пакету

SaveToMemory – оновивлення пам’яті

Train – аналіз пакетаь;

Compute - прийняття рішенняна основі Train;

Під час аналізу пакетів, нейромережа відправляє запит повторно для виключення ложнонегативних спрацювань(див. рис.3.5.):

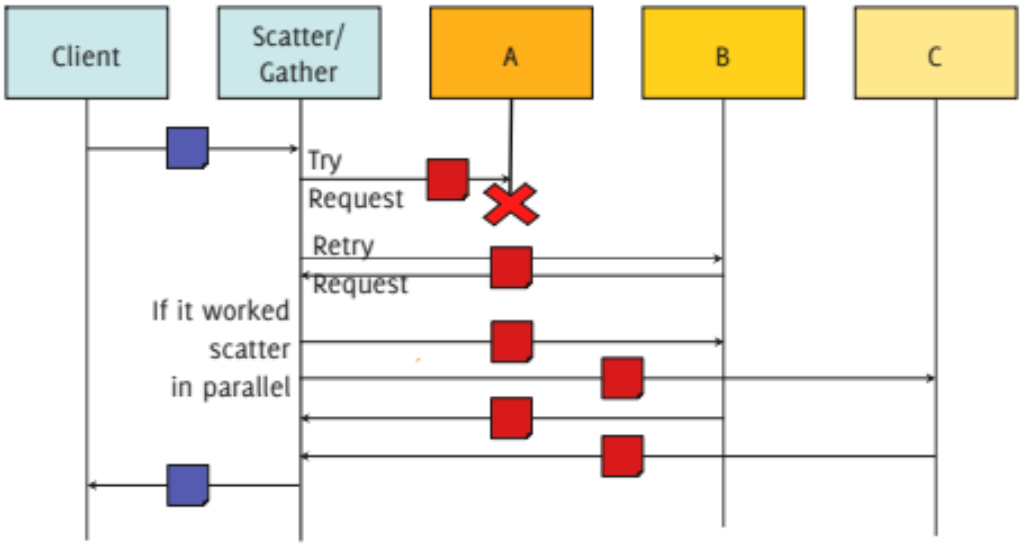

Рис. 3.5. Алгоритм перевірок

3.3. Розробка інтерфейсу програми

Для створення інтерфейсу користувача у середовищі Visual Studio 2019 було обрано проект WinForms. Консольний варіант програми не підходить для виконання поставленого завдання, через відсутність інструментів для управління ПЗ (див. рис. 3.6 і 3.7). Структурно, програма буде складатися з трьох головних складових:

- Інтерфейс програми розроблений у вигляді графічного вікна, на якому розміщені елементи управління а також поля для вводу та виводу даних.
- Логічна частина програми – це програмний код, який відповідає за логіку роботи програми. Іншими словами – це всі складові функції програми, які відповідають за її функціонал і роботу.

Допоміжні бібліотеки Pcap.dll, SharpPcap.dll, Pcap.NET.dll, необхідні для доступу до мережних інтерфейсів та перехоплення трафіку (пакетів).

Компоненти на формі:

- cbFirstDev – компонент comboBox, для вибору мережевого підключення з боку користувачів;
- cbLastDev – компонент comboBox, для вибору мережевого підключення з боку другої підмережі (роутера з Інтернет);
- tbDest – компонент textBox, для введення MAC адреси роутера підключеного до Інтернет або інший підмережі;
- dgOther – компонент dataGridView, для виведення списку всіх мережевих пристроїв, від яких йде мовлення в мережі (через другий інтерфейс основної машини);
- dgClients – компонент dataGridView, для виведення списку всіх дозволених мережевих пристроїв (користувачів). Терміну складається з: MAC адреси, IP адреси, обсягу трафіку (в байтах);
- btRefresh – компонент button, для оновлення списків dgOther і

- dgClients;
- AddClient – компонент button, для додавання обраного користувача зі списку dgOther;
- button2 – компонент button, для видалення вибраного користувача зі списку dgClients;
- btstart – компонент button, для запуску процесу сканування мережі,
- перехоплення пакетів та інших процесів;
- btstop – компонент button, для зупинки всіх процесів.

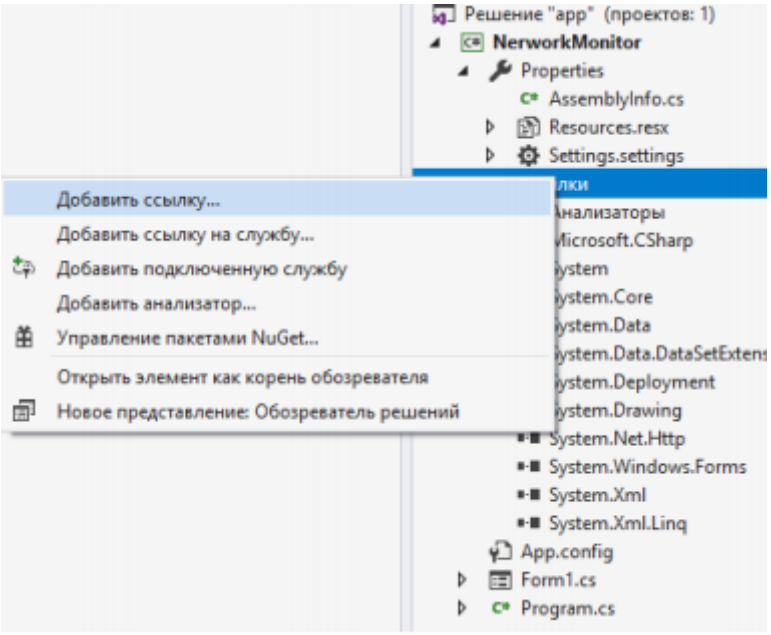

Рис. 3.6. Створення інтерфейсу

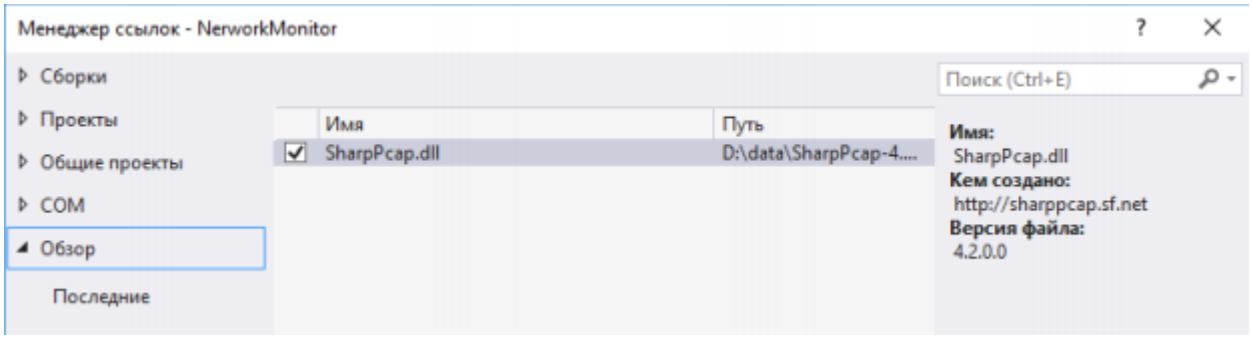

Рис. 3.7. Вибір і додавання посилань

Для наочності та опису всіх компонент були додані компоненти label. Для

періодичного оновлення списків dgOther і dgClients доданий компонент timer.

Фінальний вигляд інтерфейсу програми (див. рис. 3.8):

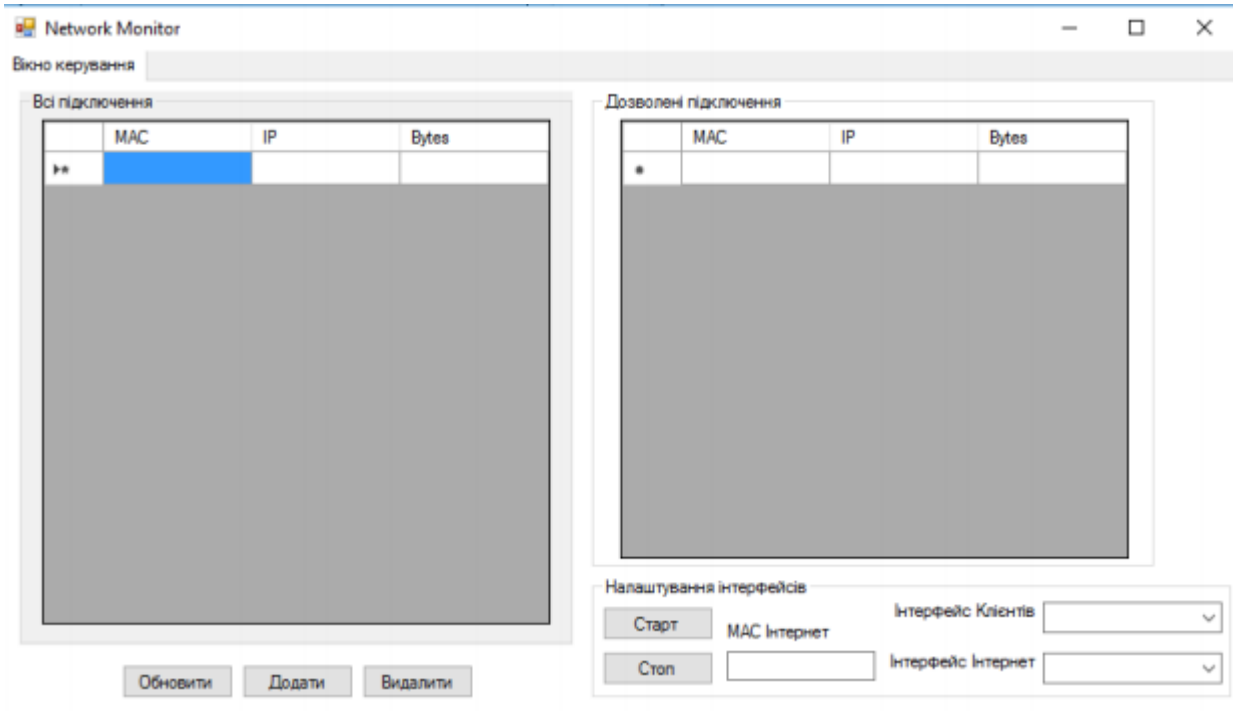

Рис 3.8. Вигляд інтерфейсу

3.4. Опис та тестування розробленого програмного забезпечення

З початку підключаємо бібліотеки (див. рис. 3.9) SharpPcap.dll і Pcap.NET.dll, які надають доступ до поточних мережевих підключень, пристроїв, а також дають можливість перехоплювати мережевий трафік і обробляти і змінювати пакети. Також під’єднаємо System.Net.NetworkInformation

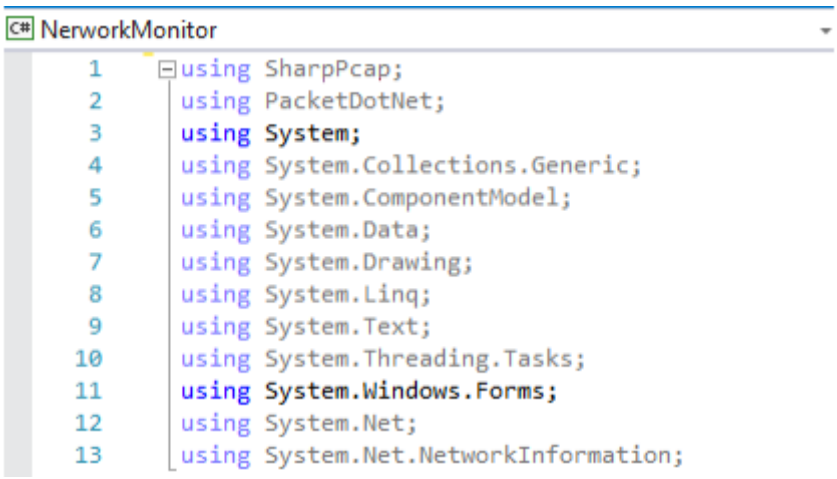

Рис.3.9. Підключення бібліотек

Для зберігання даних про дозволені мережеві інтерфейси для переміщення пакетів, використовується спеціальний текстовий файл "Users.txt". Формат записів файлу: [MAC адреса]: [IP адреса]: [обсяг трафіку].
Більш детально опис функцій наведено у додатку А. Нижче розглянемо призначення окремих функцій:

Функція Form1_Load() є обробником події завантаження форми. В тілі функції передбачена перевірка на наявність файлу "Users.txt". Якщо такий відсутній, то він створюється в директорії, де знаходиться бінарний(exe) файл. Якщо такий файл існує, виконується функція LoadInterface(), потім функція LoadClients() і в підсумку, функція ViewClient().

За допомогою методу CaptureDeviceList.Instance() можна отримати список мережевих підключень і пристроїв, записати цей перелік у компоненти cbFirstDev і cbLastDev. Далі виконується функція LoadClients().

Перш за все, з файлу "Users.txt" зчитуються дані про користувачів і записуються в об'єкт класу User.

Після того як всі записи з файлу "Users.txt" були прочитані і записані в список Users, викликається функція ViewClient() для виведення списку коричтувачів в форму в компонент dgClients.

Далі, при натисканні на кнопку "Старт" викликається функція обробник

події button3_Click().

Після запуску, функція, підключившись, до обраних мережних інтерфейсів, запускає процес перехоплення і обробки пакетів. За таймером здійснюється оновлення списку користувачів.

Функція ViewsOther() здійснює запис в компонент dgOther списку всіх

користувачів, від яких надходить трафік за час, який працює процес перехоплення. Працює при вибраному циклічно за таймером. Таймер встановлений на 20 секунд.

Функція AddClient_Click() – обробник події натискання на кнопку "Додати", здійснює копіювання вибраного рядка з компонента dgOther в файл

"Clients.txt", а звідти в компонент dgClients.

Функція button4_Click () при натисканні кнопки button4, здійснює оновлення списків dgClients, dgOther і списку Users, шляхом запуску функції RefreshClients().Оновлює список користувачів в файлі "Users.txt".

При закритті форми, виконується функція this.Dispose (). Для того, щоб

завершити процес перехоплення, необхідно натиснути на кнопку "Стоп" - btstop_Click().

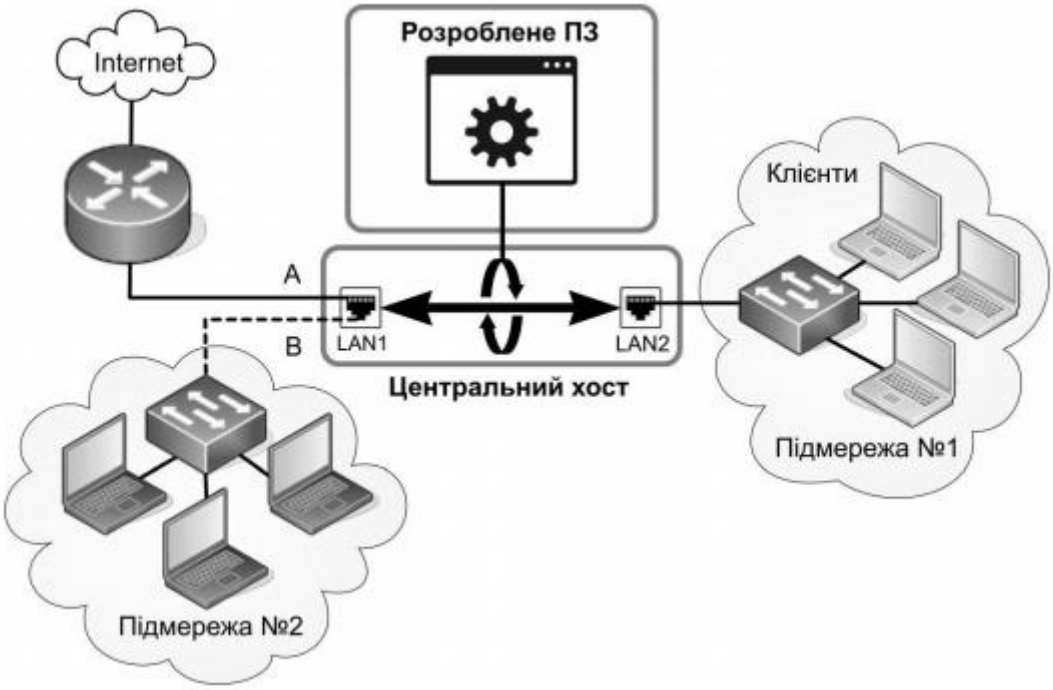

Рис. 3.10. Структура мережі з впровадженим програмним забезпеченням

У програмі реалізовані стеки і черги, для простоти реалізації процесу перехоплення і обробки пакетів. В окремому файлі "Router.cs", реалізований клас class Router. Для якого написаний користувальницький конструктор і наступні методи:

- public void Start (ICaptureDevice first, ICaptureDevice second) – приймає на вхід два мережевих пристроя, з яких буде зчитуватися і відправлятися трафік. Цей метод здійснює підміну MAC адрес і перенаправляє
- пакети (здійснює функцію роутера), саме завдяки йому, є можливість відправляти пакети з однієї підмережі, в іншу.
- public void Stop (Router router) – приймає на вхід об'єкт типу Router. Здійснює зупинку захоплення пакетів.
- void Program_OnPacketArrival10 (object sender, CaptureEventArgs) – метод, що обробляє події отримання нового пакету інтерфейсом.
- private void ProcessPacket () – здійснює обробку пакетів рівня IP.

Обробка результатів

До мінімальної конфігурації центрального ПК є певні вимоги:

- процесор: Intel Core i5-2450M, AMD Ryzen 5 3500U чи вище;
- обсяг ОП: не менше 256-512 Мб вільної пам'яті;
- обсяг жорсткого диска: не менше 128 Мб вільного місця;
- мінімум два мережевих інтерфейсу, з підтримкою режиму прослуховування мережі;
- ОС не нижче Windows 10;
- Встановлений Zabbix-агента

Обов'язковими вимогами є:

- версія NET.FrameWork не нижче 4.5.2;
- версія WinPcap не нижче 4.1.3;
- наявність бібліотек SharpPcap.dll, Pcap.NET.dll в директорії з виконуваним модулем програми;
- мережеві екрани на на центральній машині, повинні бути відключені в обов'язковому порядку.

До мінімальної конфігурації центрального ПК є певні вимоги Zabbix сервера:

- ОС RedHat Enterprise Linux
- CPU пам’ять 4 ядра / 8ГБ
- RAID10 MySQL InnoDB або PostgreSQL для БД

Перед тим як запускати програму, необхідно переконатися, що усі мережеві пристрої підключені і працюють правильно. Необхідно перевірити підключення до цих пристроїв клієнтів, інормація з яких буде фільтруватися і для яких буде здійснена політика розмежування доступу в зовнішню мережу. Зовнішньою мережею може бути як локальна мережа, адреса підмережі якої, відрізняється від першої, так і роутер, який має доступ до мережі інтернет.

Наступним кроком, є перевірка зв'язку підмережі №1 з відповідним інтерфейсом центральної машини, а підмережі №2 з мережевим пристроєм, в який вона подключена на центральній машині. Це здійснюється командою "ping". Після того як всі вузли пов'язані, можна приступати до запуску програми.

Після запуску, треба записати MAC адреси свого пристрою у текстовому полі на який буде здійснюватися перенаправлення трафіку (підмережа №1). Потім, в списках "Інтерфейс Клієнтів" і "Інтерфейс Інтернет", необхідно вибрати відповідні мережеві пристрої. Їх назви в списках будуть відповідати назвам, які вони мають в ОС. Після запуску, очікуємо появи перших адрес мережевих пристроїв, пакети від яких проходять через мережевий інтерфейс з боку підмережі №2. Інформація про ці мережеві пристрої з'являтиметься в списку "Всі підключення". А сам запис про пристрій матиме вигляд переліку MAC адреси, IP адреси і кількості байтів.

Ось такий вигляд має программа під час роботи:

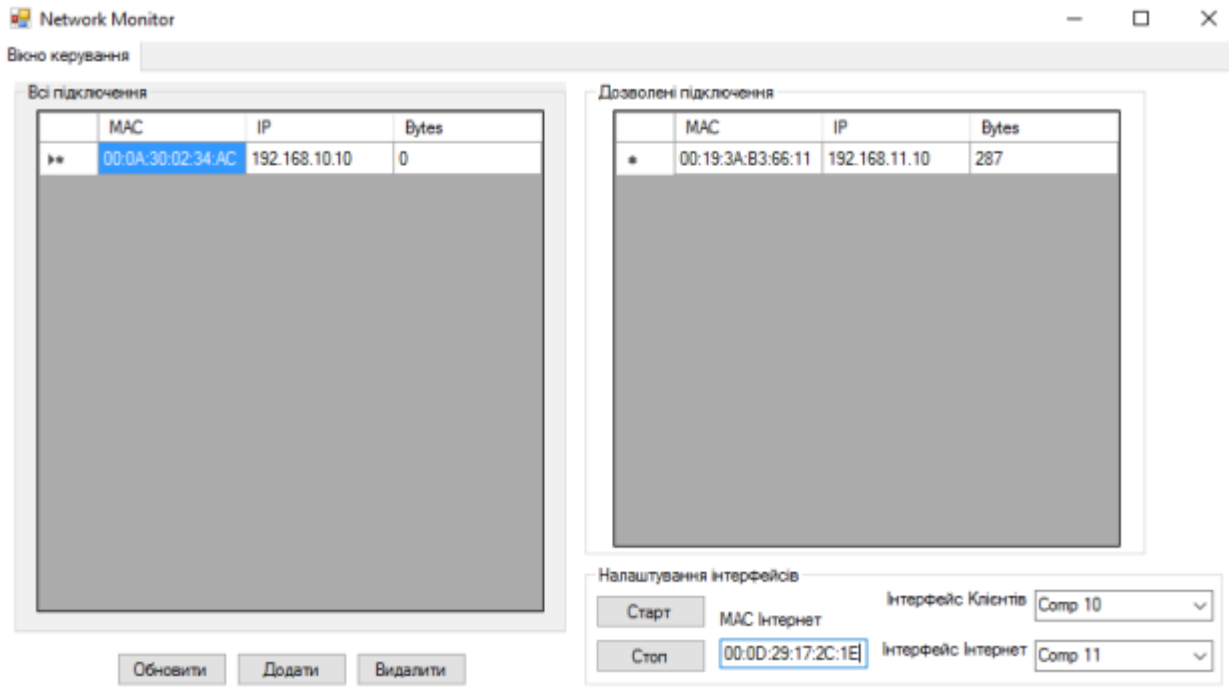

Рис. 3.11. Інтерфейс програмного модуля під час тестування

Для додавання мережевого інтерфейсу із загального списку в список "Дозволені підключення", необхідно його вибрати з списку "Всі підключення" і натиснути кнопку "Додати". Після чого, мережевий пристрій потрапляє в список "Дозволені підключення". Так само, після того як пристрій потрапляє в список "Дозволені підключення", для нього здійснюється підрахунок трафіку. Для того, щоб не чекати оновлення інформації в списку за таймером, можна скористатися

кнопкою "Обновити", і списки миттєво оновляться. Для видалення пристрою зі списку "Дозволені підключення", необхідно натиснути кнопку "Видалити". І обраний мережевий пристрій знову потрапить в список "Всі підключення". Для зупинки процесу моніторингу та фільтрації, необхідно натиснути на кнопку "Стоп" (рис.3.11). Для повторного запуску, досить буде знову натиснути на кнопку "Старт". Програма, при закритті, запам'ятовує списки дозволених клієнтів

Для тестування користувальницького інтерфейсу і реалізованих функцій, був розроблений набір можливих дій користувача:

- спроба підключення зовнішнього мережевого вузла з неіснуючою MAC адресою;
- спроба запуску програми з невибраними мережевими інтерфейсами на локальній машині;
- запуск програми без файлу клієнтів;
- вказівка неправильного формату даних у файлі з клієнтами;
- перевірка роботи всіх кнопок;
- перевірка відображення списків;
- перевірка роботи таймера;
- перевірка відображення перехоплення пакетів;
- перевірка коректності підрахунку і відображення обсягу трафіку;
- перевірка виходу з ПЗ;

На сервері Zabbix відкриваємо дію, яка опитує агентів і збирає статистику програми. Деталі можна побачити на скріншотах у вигляді графіків які збудовані внутрішніми інструментами графічної репрезентації. На перших двох зображено швидкість обробки трафіку вбудованими інструментами Zabbix (див. рис.3.10, 3.11).

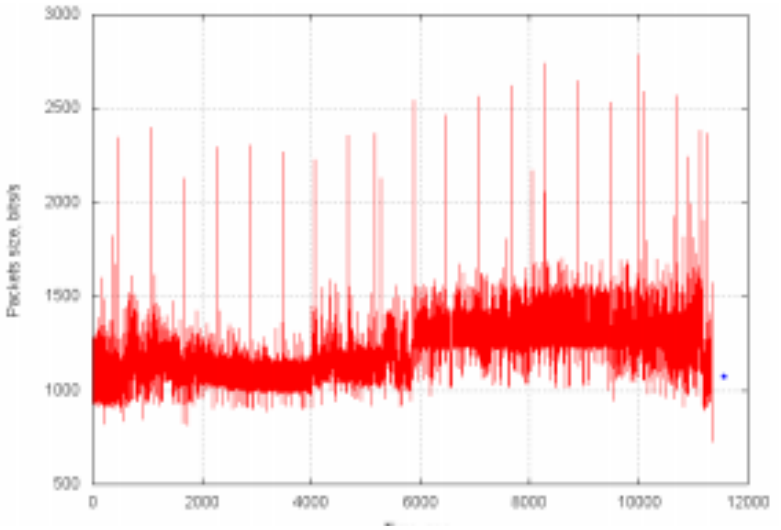

Рис. 3.12. Обробка пакетів вбудованими засобами Zabbix

Якщо порівняти ці два графіки, можна побачити що запропонований точковий метод нейромережевої обробки виявився більш ефективним у тому плані що може обробляти більший об’єм пакетів за однаковий проміжок часу.

В результаті тестування, були усунені знайдені помилки. В результаті повторного тестування додаткових помилок виявлено не було. Програмний модуль демонструє підвищену швидкість оброки трафіку користувачів у порівнянні з вбудованими інструментами. В цілому, програмне забезпечення в результаті тестування показало стабільну роботу, і готово до експлуатації.

З наведеного експертименту можна зробити ряд висновків.Можна констатувати як кількісний (у зв'язку з підйомами обсягів трафіку і ширини каналів зв'язку), такі якісне зростання (в зв'язку з новими прикладними завданнями) потреб в коштах аналізу трафіку. У питаннях реалізації систем аналізу можна відзначити наступний ряд тенденцій: агент збирає дані обробки трафіку з логів програмного модуля. У порівнянні з даними рис. 3.10 і 3.11, можна побачити що запропонований програмний модуль ефективіше оброблює той самий об’єм інформації за однаковий проміжок часу, дозволяючи аналізувати в середньому не 20000 пакетів за секунду, а більше 90000. Результат експертименту корисний як с точки зору бізнесу, так і з точки зору швидкого знаходження проблем на рівні додатків(табл.3.1).

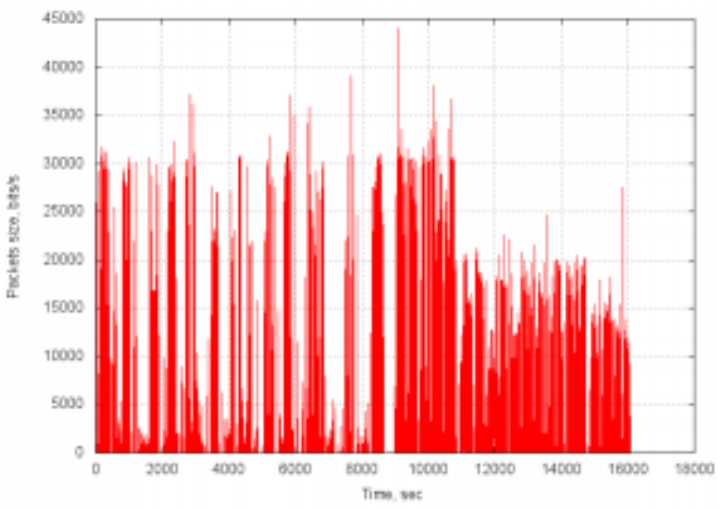

Рис. 3.13. Обробка пакетів вбудованими засобами Zabbix

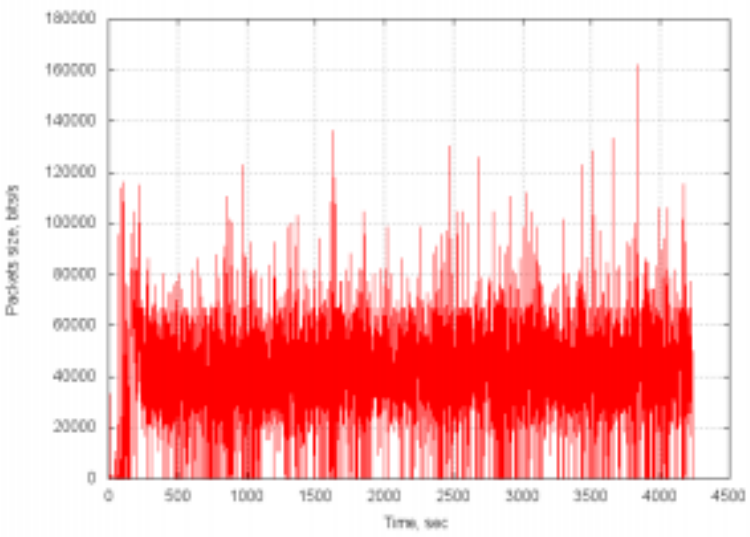

Рис. 3.14. Оброблені пакети , які агент перенаправив до Zabbix

Таблиця 3.1

Зведена таблиця порівняння існуючого з запропонованим рішення моніторинг Інтернет-трафіку

ПЗ	Швидкість	Затримка,	% втрат

	обробки, Мб/сек	мс	
Zabbix	20000	10 мс	83%
Розроблений модуль	90000	2 мс	92%

3.5. Висновки до 3 розділу

В результаті проведеної роботи по розробці програмного засобу для захисту корпоративної мережі від атак на рівні додатку, було:

- проаналізовано вимоги до моніторингу трафіку користувачів;
- обрано найбільш ефективний метод нейромережевого моніторингу для даної задачі – точкове виявлення аномалій;
- розроблено графічний користувацький інтерфейс;
- створено програмний модуль для виявлення загрози та аналізу пакетів, які намагався змінити порушник;
- за допомогою СМ Zabbix доведено ефективність обробки пакетів у реальному часі.

Всі цілі і задачі даної роботи були досягнуті. За час роботи був отриманий досвід розробки мережевих додатків мовою C#, python 2.7, json і набуті навички адміністрування Zabbix.

ВИСНОВКИ

Результатом виконаної роботи є вирішення задачі створення програмного модуля моніторингу інтернет-трафіку користувачів.

У процесі виконання роботи отримані наступні результати:

1. Наведено огляд організаційно-правової основи забезпечення захисту інформації, проведено класифікацію сучасних мережевих атак та загроз інформації і розглянуто сучасні напрями та методи захисту інформаційних ресурсів, що дало основу для подальших досліджень в даній області.

2. На основі проведеного аналізу сучасних напрямів та методів захисту інформаційних ресурсів, розглянуто методи виявлення атак та проаналізовано найпопулярніші рішення моніторингу трафіку користувачів: програмні, апаратні та програмно-апаратні.

3. Розроблено програмний модуль моніторингу Інтернет-трафіку користувачів на базі використання балансування навантаження за допомогою якого контрольований трафік розподіляється по вихідним портам 10G обєктно-орієнтованою мовою програмування C#, який дозволяє забезпечити захисту інформації при її передачі в інформаційних мережах.

4. Дослідження системи було проведено за допомогою СМ Zabbix, де було порівняно швидкість обробки однакової кількості трафіку за однаковий проміжок часу внутрішніми інструментами СМ Zabbix і розробленим програмним модулем. За результатами досліження, запропонований програмний модуль виявився більш ефективним у швидкості обробки трафіку.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Applying Network-Centric Approaches for Threat Detection and Response URL: https://www.gartner.com/en/documents/3904768/applying-network-centric-approaches-for-threat-detection

2. Распространенные угрозы ИБ в корпоративных сетях [URL: https://www.ptsecurity.com/ru-ru/research/analytics/network-traffic-analysis-2020/

3. Про захист інформації в інформаційно-телекомунікаційних системах [Текст] : Закон України № 80/94-ВР від 4 липня 2020 р. / Верховна Рада України // Відомості Верховної Ради України. – 1994. – №31. – Ст. 286.

4. Введение в информационную безопасность. Компьютеры: преступления, признаки уязвимости и меры защиты URL:http://www.bezpeka.com/ru/lib/sec/gen/art344.html

5. Лукацкий А. Обнаружение атак. — СПб.: БХВПетербург, 2001. – 624 с.

6. Офіційний сайт кіберполіції України URL:Режим доступу https:// https://cyberpolice.gov.ua/

7. М. А. Карпенко, студентка;и О. В. Коломієць, студент, АНАЛІЗ СУЧАСНИХ СИСТЕМ ВИЯВЛЕНЯ ВТОРГНЕНЬ В IoT ТА ЗАПОБІГАНЯ ЇМ URL:http://www.itrew.ru/windows/statistika-operacionnykh-sistem-za-ap.html.

8. Технологии обнаружения

URL:https://www.bytemag.ru/articles/detail.php?ID=6850&sphrase_id=3833171.

9. J.P. Anderson, Computer Security Threat Monitoring and Surveillance // James P. Anderson Co., Fort Washington, PA, April. 1980, P. 75-127.

10. Городецкий В.И., Котенко И.В., Карсаев О.В., Хабаров А.В. Многоагентные технологии комплексной защиты информации в

телекоммуникационных системах. ISINAS. 2000. – №3. – P. 89-92.

11. Бараматова И. С. Зайцева Е. В. Состояние и перспективы развития систем обнаружения компьютерных вторжений // Горный информационно-аналитический бюллетень (научно-технический журнал). Вып. S6. 2011

12. Кількість мобільних пристроїв з інтернетом скоро перевищить населення Землі URL: http://fmf.udpu.org.ua/novyny-suchasnoinauky/593-kilkist-mobilnykh-prystroiv-na-zemliperevyshchyla-kilkist-liudei.

13. Aberrant Behavior Detection in Time Series for Monitoring Business-Critical Metrics URL:http://www.imvu.com/technology/anomalous-behavior.pdf

14. Anomaly detection for monitoring: A statistical approach to time series anomaly detection. O'Reilly. URL: http://www.oreilly.com/webops-perf/free/files/anomaly-detection-monitoring.pdf

15. Context-Aware Time Series Anomaly Detection for Complex Systems. Proceedings of the SDM Workshop. URL: https://www.microsoft.com/en-us/research/publication/context-aware-time-series-anomaly-detection-for-complex-systems/

16. Long Short Term Memory Networks for Anomaly Detection in Time Series. ESANN 2015 Proceedings. URL: https://www.elen.ucl.ac.be/Proceedings/esann/esannpdf/es2015-56.pdf Pankaj Malhotra, Lovekesh Vig, Gautam Shroff, Puneet Agarwal.

17. S. Agrawal, J. Agrawal, "Survey on Anomaly Detection using Data Mining Techniques", Procedia Computer Science, vol. 60, 2015, pp. 708 – 713.

18. C. Chio, D. Freeman. "Machine Learning and Security", O'Reilly Media, Inc., – URL: https://github.com/oreilly-mlsec/bookresources/tree/master/chapter3/datasets/cpu-utilization. – 01.12.2017.

19. Network Monitoring Best Practices [Електроний ресурс] – URL:

https://www.dnsstuff.com/how-to-monitor-network-traffic

20. Dover D., Dafforn E. Search Engine Optimization Secrets. Indianapolis: Wiley Publishing, Inc., 2011. 456 p.

21. Cardoso, J., Sheth, Amit (Eds.) ,. Semantic Neural Services, Processes and Applications. - Springer, 2006. - ISBN 0-387-30239-5.

22. Томас Коннолли. Базы данных. Проектирование, реализация и сопровождение. Теория и практика.

23. Роберт К. Элсенпитер, Тоби Дж. Велт. Нейронные сети строим сами. 2006. - 256 с.

24. WiMAX Forum URL:wimaxforum.org/

25. WLAN: практическое руководство для администраторов и профессиональных пользователей» / Томас Мауфер. – М.: КУДИЦ-Образ, 2005

26. Марк Лутц. Программирование на Python / Пер. с англ. - Четвѐртый изд. - СПб .: Символ-Плюс, 2011. - Т. I. - 992 с. - ISBN 978-5-93286-210-0. 14) Neural Network in school and house — Scout Blog, 7 дек 2016- 482 с. - ISBN 978-5-97060-315-4.

27. Берлин А.Н. Цифровые сотовые системы связи. М.: Эко-Трендз, 2007 20) В. Вишневский, С. Портной, И. Шахнович. Энциклопедия WiMAX. Путь к 4G. Техносфера, 2009