МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ДОНЕЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМЕНІ ВАСИЛЯ СТУСА

Д. В. Чернов, В. Г. Крижановський

СИСТЕМИ ВИЯВЛЕННЯ ВТОРГНЕНЬ

Методичні вказівки до самостійної роботи
з дисципліни для здобувачів освіти ОС «Бакалавр»
спеціальності 125 Кібербезпека та захист інформації

Частина перша

Вінниця
2024

УДК 004.056.53:681.5(075.4)
С 409

Рекомендовано до друку вченою радою
факультету інформаційних і прикладних технологій
Донецького національного університету імені Василя Стуса
(протокол № 8 від 20 березня 2024 р.)

Укладачі:
Чернов Д. В., старший викладач кафедри прикладної математики та кібербезпеки Донецького національно університету імені Василя Стуса;
Крижановський В. Г., проф. кафедри прикладної математики та кібербезпеки Донецького національно університету імені Василя Стуса.

Рецензенти:
Ніколюк П. К., д-р фіз.-мат. наук, проф., проф. кафедри інформаційних технологій Донецького національного університету імені Василя Стуса;
Васильківський М. В., канд. техн. наук, доцент, доц. кафедри інфокомунікаційних систем і технологій Вінницького національного технічного університету.

С 409 Системи виявлення вторгнень: методичні вказівки до самостійної роботи з дисципліни для здобувачів освіти ОС «Бакалавр» спеціальності 125 Кібербезпека та захист інформації. Частина перша / укл. Д. В. Чернов, В. Г. Крижановський. Вінниця: ДонНУ імені Василя Стуса, 2024. 28 с.

У методичних вказівках надано додаткові відомості для вивчення окремих тем курсу «Системи виявлення вторгнень», які переважно стосуються класифікації атак у мережі за видами та з використанням вад програмного забезпечення.

Посібник рекомендовано для студентів закладів вищої освіти спеціальності 125 Кібербезпека та захист інформації та може бути корисний студентам споріднених спеціальностей.

УДК 004.056.53:681.5(075.4)

ЗМІСТ

ВСТУП

Великі зміни відбуваються у сфері пропозиції та попиту на інформацію завдяки широкому розповсюдженню комп'ютерів і експоненційному зростанню комп'ютерних мереж, як-от Інтернет. Інтернет став популярним засобом комерційної діяльності, і це підвищило ставки як для зловмисників, так і для співробітників служби безпеки. У кожній великій фінансовій установі щоденно здійснюються операції на трильйони доларів. Наприклад, Visa обробляє 4 000 транзакцій на секунду, що означає, що якщо система Visa вийде з ладу на одну хвилину через розподілену атаку на відмову в обслуговуванні (DDoS), і припустимо, що транзакція становить лише 100 доларів США, за одну хвилину буде втрачено понад 24 мільйони доларів США.

Сьогодні світ бізнес-обчислень стикається зі все більшою ймовірністю незапланованих простоїв через різні атаки та порушення безпеки. У цьому середовищі невизначеності, повному хакерів і зловмисних загроз, ті компанії світу, які найкраще підтримують безперервність своїх послуг (тобто виживають у системі) і зберігають свою обчислювальну потужність, мають значну конкурентну перевагу.

Перерви в роботі мережі призводять до фінансових втрат і ще більшої шкоди довірі до комерційних підприємств, особливо провайдерів. Зведення до мінімуму або, можливо, усунення незапланованих простоїв системи забезпечує безперервність обчислювальних послуг. Звести до мінімуму неочікувану і незаплановану зупинку можна шляхом визначення пріоритетів і захисту від неправильного використання, атак і вразливостей. Завдання полягає в тому, щоб зменшити ймовірність катастрофічних інцидентів шляхом: а) використання відповідних машинних і статистичних методів навчання для оцінки відносної небезпеки окремих загроз і б) автономного надання ефективної та адекватної відповіді на відповідні загрози.

1. ОГЛЯД СИСТЕМ ВИЯВЛЕННЯ ВТОРГНЕНЬ

Система виявлення вторгнень (IDS – Intrusion Detection System) – це галузь, яка швидко розвивається і займається виявленням зловмисного мережевого трафіка, неправомірного використання комп'ютера та реагуванням на нього. Виявлення вторгнень – це процес ідентифікації та (можливо) реагування на зловмисну діяльність, спрямовану на обчислювальні та мережеві ресурси. Будь-яке апаратне або програмне забезпечення автоматизації, яке відстежує, виявляє або реагує на події, що відбуваються в мережі або на головному комп'ютері, вважається відповідним підходу до виявлення вторгнень. Різні IDS надають різні функції та переваги.

Спроба зламати або зловживати системою називається «вторгненням». Зазвичай вторгнення використовує певну вразливість і має бути виявлене якомога швидше. Системи виявлення вторгнень є важливими компонентами інфраструктури безпеки мережі. Вони перевіряють системну або мережеву активність, щоб виявити можливі вторгнення або атаки, та ініціюють сповіщення безпеки про зловмисну діяльність. Зазвичай їх класифікують як системи виявлення на основі сигнатур і аномалій. До інших категорій належать мережеві та хост-системи виявлення вторгнень.

Мережеві IDS розміщуються у стратегічній точці або точках у мережі, щоб перевірити мережевий трафік, що проходить, на ознаки вторгнення, тоді як IDS на основі хоста запускаються на окремих хостах або пристроях у мережі та перевіряють активність користувачів і процесів на ознаки зловмисної поведінки.

У більшості мереж розгортаються ***IDS на основі сигнатур,*** які дуже ефективні проти відомих атак. Системи на основі сигнатур (також відомі як системи виявлення зловживання) виявляють атаки на основі відомих випадків зловживання. Перевагами виявлення зловживань є висока впевненість у виявленні, низький рівень помилкових позитивних результатів і однозначна детальна ідентифікація атаки. Вони також більш зрозумілі та широко застосовуються. Недоліками є нездатність виявляти невідомі атаки та необхідність експертних знань для створення підписів.

Для захисту від невідомих атак необхідно використовувати схему виявлення аномалій, яка створює модель нормальної поведінки системи та виявляє відхилення від цієї моделі. Методи, які використовуються для виявлення аномалій, включають аналіз даних, кластеризацію та статистичну обробку сигналів. Основною перевагою систем, заснованих на аномаліях, є здатність виявляти невідомі атаки. Недоліками є високий відсоток помилкових позитивних результатів і складність визначення типу атаки. Навіть більше, оскільки те, що вважається нормальним, може відрізнятися в різних середовищах, окрему модель нормальності потрібно вивчити індивідуально.

Останнім класом детекторів вторгнень є детектори на основі специфікацій, які намагаються досягти спільного між системами, заснованими на неправильному використанні, і системами, заснованими на аномаліях. Вони в основному базуються на специфікаціях, отриманих з протоколів, і виявляють відхилення від цих специфікацій. Незважаючи на те, що вони поєднують переваги виявлення аномалій і виявлення неправильного використання, вони страждають від недоліку, пов'я-

заного з тим, що повні специфікації важко створити, особливо враховуючи, що більшість протоколів постійно розширюються.

Через експоненціальне зростання розміру, розподілу та складності комунікаційних мереж поточні технології IDS не дуже ефективні проти нових атак і мають серйозні обмеження щодо продуктивності, масштабованості та гнучкості. Навіть більше, вдосконалення IDS часто надто повільні та замалі, щоб встигати за нововведеннями зловмисників.

Основними недоліками сучасних IDS є: 1) велика кількість хибних спрацьовувань; 2) неможливість виявлення невідомих атак; 3) нездатність належним способом оцінити відносну небезпеку неправильного використання та забезпечити відповідну відповідь. Існує загальний консенсус щодо того, що першочерговою метою технологій виявлення вторгнень має бути: а) зниження рівня помилкових спрацьовувань; б) розроблення несигнатурних методів виявлення вторгнень; в) робота над запобіганням замість виявлення.

Існує критична потреба мати можливість створювати системи, які можуть автоматично виявляти шаблони вторгнень і вузькі місця продуктивності, а також динамічно захищатися. Основна мета системи виявлення вторгнень – вижити в системі та зберегти її основні служби. Живучість часто визначається опором, визнанням і відновленням. Опір займається посиленням системи, щоб запобігти злому чи іншим зловмисним діям. Метою розпізнавання є виявлення нав'язливої поведінки від нормальної поведінки. Відновлення стосується способів виживання після зловмисних дій.

Будь-яке рішення проблеми живучості має відповідати трьом основним критеріям, включно з динамічною зміною шаблонів мережевого трафіка, появою непередбачуваних подій (порушень безпеки) і нескінченною базою середовища мережевого трафіка. Один зі способів розробки інструментів живучості, здатних легко інтегруватися в поточні динамічні мережеві середовища, – це спроєктувати їх як розподілену систему на основі агентів. Ключовим елементом такої системи є інтелектуальний агент, здатний аналізувати ситуацію, приймати рішення та спілкуватися з іншими агентами та користувачами. Багатоагентні системи разом із нечіткими системами можна використовувати для створення спільноти інтелектуальних агентів із функціями автономії та самостійності.

Протягом останніх років дослідники виявлення вторгнень приділяли значну увагу методам машинного навчання та інтелектуального аналізу даних, щоб усунути недоліки методів виявлення на базі знань. Це призвело до застосування різних контрольованих і неконтрольованих методів з метою виявлення вторгнень. Книга [1] містить вичерпний огляд сучасних тенденцій у системах виявлення вторгнень і відповідних технологіях. Ми представляємо набір експериментів, які проводяться для аналізу ефективності неконтрольованих і керованих методів машинного навчання з урахуванням їх основних варіантів дизайну.

Протягом останнього десятиліття виявлення аномалій привернуло увагу багатьох дослідників, щоб подолати слабкість IDS на основі сигнатур у виявленні нових атак. Однак, маючи відносно високий рівень помилкових тривог, виявлення аномалій не було широко використано в реальних мережах. У [1] представлено керовані даними підходи до автоматизації моделювання поведінки мережі. Одним із

підходів є техніка, яку розроблено для створення моделі мережевих сигналів ARX і її використання для виявлення мережевих аномалій, викликаних вторгненнями. Сигнали мережі нестаціонарні, високонестабільні, і їх важко моделювати традиційними методами. Техніка моделювання, що використовує поєднання теорії ідентифікації системи та вейвлет-апроксимації, дуже ефективна для вирішення цієї проблеми.

Кореляція сповіщень є важливою технікою для керування великою кількістю сповіщень про вторгнення, які надсилаються різнорідними IDS. Остання тенденція досліджень у цій галузі спрямована на вилучення стратегій атак із необроблених сповіщень про вторгнення. Знання реальної ситуації з безпекою мережі та стратегій, які використовують зловмисники, дає змогу мережевим адміністраторам задіяти відповідну відповідь, щоб зупинити атаки та запобігти їх ескалації. Техніка керування сповіщеннями та кореляцію, яка може допомогти автоматично витягти стратегії атаки з великої кількості сповіщень про вторгнення без певних попередніх знань про ці сповіщення описана в [1].

Книги з виявлення вторгнень на ринку є відносно неорієнтованими. Вони схильні пропускати деталі різноманітних ключових технік і моделей. До того ж у багатьох книгах бракує детальної інформації щодо різних типів атак, теоретичних основ підходів до виявлення атак, реалізації, збору даних, оцінки та реагування на вторгнення. Заявлена мета книги [1] – надати просту, але докладну та стислу інформацію з усіх цих тем. Також надається детальний огляд деяких комерційно / загальнодоступних систем виявлення вторгнень і реагування на них. Щодо системи виявлення вторгнень неможливо охопити все, що можна сказати з усіх тем. Проте зроблена спроба охопити найважливіші та найпоширеніші з них.

2. МЕРЕЖЕВІ АТАКИ

Мережеві атаки визначаються як набір зловмисних дій, спрямованих на порушення, заборону, погіршення або знищення інформації та послуг, які знаходяться в комп'ютерних мережах. Мережева атака здійснюється через потік даних у мережах і має на меті порушити цілісність, конфіденційність або доступність комп'ютерних мережевих систем. Мережеві атаки можуть варіюватися від набридливої електронної пошти, спрямованої на особу, до атак вторгнення на конфіденційні дані, комп'ютерні інформаційні системи та критичну мережеву інфраструктуру. Приклади комп'ютерних атак включають: віруси, вкладені в електронні листи; зондування системи для збору інформації; інтернет-хробаки; несанкціоноване використання системи; відмова в обслуговуванні через зловживання функцією системи; використання помилки в програмному забезпеченні для зміни системних даних. Деякі загальні підходи, які зловмисники можуть використовувати для отримання доступу до системи або обмеження доступності цієї системи, включають соціальну інженерію, маскування, вразливість реалізації та зловживання функціональністю. Зокрема, соціальна інженерія – це метод атаки для введення жертви в оману шляхом агресивних переконань або використання інших навичок міжособистісного спілкування для отримання інформації для автентифікації або доступу до системи: наприклад, фішинг електронної пошти та троянські програми електронної пошти;

маскування – тип атаки, коли зловмисник видає себе за авторизованого користувача системи, щоб отримати доступ до неї або отримати більші привілеї, ніж вони авторизовані (обхід механізму автентифікації за допомогою вкрадених ідентифікаторів входу та паролів); уразливість реалізації – помилка програмного забезпечення в довірених програмах, якою зловмисник може скористатися для отримання несанкціонованого доступу до системи (переповнення буфера, умови змагання та неправильна обробка тимчасових файлів); зловживання функціональними можливостями означає зловмисну діяльність, яку виконує зловмисник, щоб підштовхнути систему до збою, перестаравшись із законною дією (заповнюючи таблицю системних процесів шляхом відкриття сотень підключень Telnet до інших комп'ютерів). У цьому розділі ми детально розглянемо всі ці мережеві атаки.

2.1. Таксономії атак

Хоча класифікація – це просто поділ або впорядкування об'єктів на класи, таксономія – це теоретичне дослідження класифікації, включно з її основами, принципами, процедурами та правилами. Метою таксономії атак є надання теоретичних і послідовних засобів класифікації комп'ютерних і мережевих атак, підвищуючи в такий спосіб ефективність обміну інформацією під час опису атак. У цьому розділі ми розглянемо три типові класифікації комп'ютерних і мережевих атак із погляду загального використання, спеціального використання та зловмисника відповідно.

Таксономія під назвою VERDICT (Validation Exposure Randomness Deallocation Improper Conditions Taxonomy), показує, що всі комп'ютерні атаки можна класифікувати за чотирма неправильними умовами, а саме перевіркою, експозицією, випадковістю та звільненням.

Під час неналежної перевірки недостатня або неправильна перевірка призведе до несанкціонованого доступу до критичної інформації або захищеної системи. Помилки, викликані неправильною перевіркою, становлять широку категорію. У типовій операційній системі помилка захисту станеться для всієї системи, якщо її критичні компоненти та оператори отримають необмежені або недійсні дані. Щоб уникнути неналежної перевірки, параметри, які передаються між двома системними компонентами або між системним компонентом і зовнішнім об'єктом, повинні перевірятися відповідно до набору умов, а саме наявності чи відсутності, типів і форматів даних, кількості та порядку, діапазонів значень, прав доступу до пов'язаних місць зберігання, узгодженості між параметрами.

Неналежні ризики завжди трапляються, якщо задовольняються певні умови ризиків. Наприклад, підпорядкований процес може отримати доступ до привілейованої інформації, яка знаходиться у сховищі, або привілейована інформація передається підпорядкованому процесу опосередковано через підтвердження або час. У цій ситуації важлива інформаційна система буде неналежним способом піддана атаці.

Неправильна випадковість може призвести до атаки. Вирішальним аспектом у криптографії є генерування випадкових чисел. Однак через відсутність справжніх випадкових джерел замість них у сучасних комп'ютерних системах використо-

вуються псевдовипадкові числа, що надзвичайно ускладнює розробку незламних блоків шифрування.

Неправильне звільнення означає, що інформація, яка зберігається в системі, не видаляється належним способом після використання, а отже, це призведе до вразливості системи до атаки. Типовим прикладом неправильного звільнення є видалення файлу з диска. На практиці більшість операційних систем фактично не стирають дані файла з диска, натомість вони просто вилучають зайняті сектори з таблиць розподілу. Видалення не буде завершено, доки розташування цього файла на диску не буде повністю перезаписано певними шаблонами.

Незважаючи на те, що VERDICT успішно застосовується для класифікації наявних атак і пошуку численних нових вразливостей у бездротовій мережі, у цій таксономії все ще є кілька недоліків. З погляду організації безпеки, як-от CERT (Computer Emergency Response Team), VERDICT не можна використовувати для ідентифікації та класифікації щоденних нових атак. Він є загальним і абстрактним і не дає опису атак з погляду вірусів, хробаків, троянів і шкідливих програм, як зазвичай описують атаки в реальності.

Далі ми розглянемо іншу типову таксономічну роботу, запропоновану Гансманом з щодо конкретних класів атак. Порівняно з VERDICT, таксономія Гансмана є більш повною та практичною, і включає чотири виміри:

– Перший вимір охоплює основну поведінку атаки.

– Другий вимір дає змогу класифікувати цілі атаки.

– Третій вимір класифікує вразливості та експлойти, які використовують зловмисники.

– Четвертий вимір враховує корисне навантаження, щоб атака мала ефект, що перевищує саму себе.

У першому вимірі атаки були класифіковані за 10 категоріями, які перераховані нижче:

– Вірус – самовідтворювана програма, яка приєднується до наявної програми та заражає систему без дозволу або відома користувача.

– Worm – самовідтворювана програма, яка поширюється мережевими службами на комп’ютерах без будь-якого втручання користувачів.

– Троян – частина програми, створена для виконання певної безпечної дії, але насправді виконує інший код зі зловмисною метою.

– Переповнення буфера – процес, який отримує контроль або аварійно завершує роботу іншого процесу шляхом перезапису межі буфера фіксованої довжини.

– Відмова в обслуговуванні – атака, яка перешкоджає законним користувачам отримати доступ або використовувати комп’ютер чи мережевий ресурс.

– Мережева атака – атака, яка призводить до збою користувачів у мережі або самої мережі через маніпулювання мережевими протоколами, починаючи від рівня каналу даних і закінчуючи прикладним рівнем.

– Фізична атака – атака, яка намагається пошкодити фізичні компоненти мережі або комп’ютера.

– Атака на пароль – атака, яка спрямована на отримання пароля і зазвичай проявляється серією невдалих входів протягом короткого періоду часу.

– Атака зі збором інформації – атака, яка збирає інформацію або знаходить відомі вразливості шляхом сканування або дослідження наявних комп'ютерних мереж.

У другому вимірі цілями зловмисника можуть бути дуже конкретні цілі (наприклад, IIS 4.0), або охоплювати клас цілей (наприклад, системи MacOS, системи Unix тощо). Цілями можуть бути апаратне або програмне забезпечення. Апаратні цілі зазвичай складаються з комп'ютера, мережевого обладнання та периферійних пристроїв. Цільовими комп'ютерами є всі пов'язані з комп'ютером компоненти, як-от процесори та жорсткі диски. Мережеве обладнання може бути пристроями, як-от маршрутизатори, комутатори або концентратори. Периферійні пристрої – це пристрої, які не є важливими для роботи комп'ютера, наприклад, миша, клавіатура тощо. Цільове програмне забезпечення включає три основні класи: операційна система, програми та мережа. Цілі операційної системи позначають різні сімейства ОС, як-от Linux, Unix, Windows тощо. Цільові програми – це програми, які працюють поверх операційної системи, що складається з серверної програми (наприклад, сервера бази даних, вебсервера) та програми користувача (наприклад, клієнт електронної пошти, редактор Word). Цільова мережа фокусується на самій мережі або її протоколі (наприклад, протоколі транспортного рівня).

У третьому вимірі розглядаються вразливості та експлойти, які використовуються під час атаки, які зазвичай класифікуються як записи про загальні вразливості та експозиції (CVE). Проєкт CVE призначений для створення загальних визначень вразливостей [1], які спочатку були запропоновані Манном і Крісті. Оскільки вразливості численні та різноманітні, вони зазвичай стосуються певних версій програмного забезпечення або операційних систем. Коли відомі вразливості, які використовує атака, можна знайти відповідні записи CVE.

Атаки з корисним навантаженням розглядаються в четвертому вимірі, оскільки різні корисні навантаження можуть мати різні наслідки, крім самої атаки. Наприклад, атака хробака може мати троянську програму. Внаслідок цього таксономія дає змогу атакам, класифікованим у першому вимірі, запускати інші атаки, визначені в четвертому вимірі. У четвертому вимірі корисні навантаження поділяються на п'ять категорій, а саме:

– корисне навантаження атаки першого виміру;
– пошкодження інформації;
– розкриття інформації;
– крадіжка послуги;
– підривна діяльність.

Корисне навантаження атаки першого виміру визначається відповідно до класу атаки в першому вимірі. Пошкодження інформаційного корисного навантаження змінює або знищує деяку інформацію. Розкриття корисної інформації розголошує інформацію без дозволу жертви. Крадіжка служб доступу до корисного навантаження систем без будь-якого дозволу та без жодного впливу на послуги законних користувачів. Підривна дія відбувається, коли корисне навантаження може отримати контроль над частиною цілі, а потім використовувати її для власних цілей.

2.2. Зонди

Мережеві зонди – це зазвичай атаки, які сканують комп'ютерні мережі для збору інформації або пошуку відомих вразливостей, які використовуються для подальших або майбутніх атак. Мета цього збору інформації полягає в тому, щоб дізнатися про комп'ютер і служби, які наявні в мережі, а також виявити можливість атаки на основі відомих вразливостей. У цьому розділі ми представляємо кілька загальнодоступних інструментів сканування, які використовуються для зондування мережі.

2.2.1. IPSweep і PortSweep

Атака IPSweep визначає, які хости прослуховують мережі за допомогою розгортки спостереження. Portsweep використовується для перевірки того, які порти зазначеного комп'ютера відкриті в мережі (або підмережі). Атаки IPSweep і PortSweep визначають запущений хост і типи його служб, а потім зібрану інформацію можуть використовувати зловмисники для постановки атак і пошуку вразливих комп'ютерів. Виявляти поточну атаку IPSweep або PortSweep доволі легко, особливо якщо вона виконується лінійним способом з одного джерела. Важче виявити, чи використовуються під час очищення кілька хостів, підроблені IP-адреси джерела або нелінійні в часі.

2.2.2. NMap

NMap – це безкоштовна утиліта з відкритим кодом, яка в основному виконує сканування IP-адрес, сканування портів, сканування брандмауера та сканування відбитків ОС за допомогою необроблених IP-пакетів, спрямованих на комп'ютери-жертви. Час між пакетами регулюється, і порти можуть скануватися послідовно або випадково. Виявлення сканування, проведеного NMap, є складним через розподіл кількох джерел і уповільнення графіка атаки, що робить зондування прихованим протягом тривалого періоду часу.

2.2.3. MScan

MScan використовує передачу зони DNS і сканування грубою силою для всіх доменів і повних діапазонів IP-адрес, щоб виявити активні комп'ютери та перевірити їх на відомі вразливості в різних мережевих службах, як-от statd, NFS, програми cgibin, відкриті сервери X, pop3, imap і палець. Існують різні сигнатури для виявлення атак MScan, залежно від того, які дефекти та скільки цільових комп'ютерів досліджується.

2.2.4. SAINT

SAINT – це акронім інтегрованого мережевого інструменту адміністратора безпеки. Він збирає велику кількість інформації з мережевих служб, як-от finger, ftp, telnet, tftp, NIS, NFS, rexd, statd та деяких інших служб. SAINT не є інструментом для атаки у своїй основній формі. Однак він збирає інформацію, яка може бути використана зловмисниками для подальших вторгнень. Інформація в основному містить неправильну конфігурацію мережевих служб, добре відомі вразливості в операційних системах або мережевих утилітах і слабкі політичні рішення.

SAINT пропонує три додаткові режими роботи, а саме легкий режим, звичайну модель і важку модель. У легкому режимі SAINT сканує цільовий комп'ютер на виявлення вразливостей RPC і DNS, а також незахищених точок монтування NFS. У звичайному режимі SAINT виявляє вразливості fingerd, bootd і rusersd і сканує кілька добре відомих TCP-портів, як-от HTTP, FTP, Telnet, SMTP, UUCP і NNTP, і UDP-портів, як-от DNS. Напружений режим схожий на звичайний режим, за винятком того, що також сканується багато інших незвичайних портів.

2.2.5. Satan

Satan – це набір програм на C і Perl і є попередньою версією SAINT. Вони доволі схожі за призначенням і конструкцією. Основна відмінність між ними полягає в конкретних вразливостях, які вони сканують. Як і SAINT, Satan підтримує три рівні сканування: легкий, звичайний і важкий. Приклад сканування списку вразливостей у важкому режимі:

- експорт NFS до непривілейованих програм;
- експорт NES через програму відображення портів;
- доступ до файла пароля NIS;
- доступ REXD;
- доступ до файлів TFTP;
- віддалений доступ до оболонки;
- необмежений експорт NES;
- необмежений доступ до X-сервера;
- домашній каталог FTP із можливістю запису;
- кілька вразливостей Sendmail;
- кілька вразливостей FTP.

Оскільки Satan досліджує вразливі місця, використовуючи той самий порядок, що наведено вище, їх можна визначити на основі послідовного шаблону мережевого трафіка, який генерує програма.

2.3. Атаки на підвищення привілеїв

Атаки на підвищення привілеїв – це атаки, під час яких зловмисник використовує помилку в програмному забезпеченні, щоб отримати доступ до ресурсу, який зазвичай був би захищений від програми або користувача, що призводить до ескалації або використання поточних рівнів привілеїв для виконання зловмисної дії з більшими привілеями, ніж передбачалося розробником програми та системним адміністратором. Відомі атаки цього типу можна загалом розділити на дві категорії:

1. Вертикальна ескалація привілеїв, або від користувача до суперкористувача: користувач із нижчими привілеями зі звичайним обліковим записом користувача в системі використовує деякі недоліки для оцінки функцій або вмісту, зарезервованого для користувачів з вищими привілеями, або суперкористувачів (root).

2. Горизонтальна ескалація привілеїв, або від некористувача до користувача: звичайний користувач зі звичайним обліковим записом або без будь-якого облікового запису в системі використовує певну вразливість у системі, щоб оцінити функції чи вміст, зарезервований для інших звичайних користувачів.

У цьому розділі проілюстровано деякі приклади добре відомих атак на підвищення привілеїв.

2.3.1. Атаки переповнення буфера

Буфер – це безперервна виділена частина пам'яті, наприклад, масив або вказівник мовою С. Зазвичай у буфері немає автоматичної перевірки меж. Переповнення буфера відбувається, коли програма або процес намагається зберегти більше даних у буфері, ніж він призначений. Оскільки буфери створюються для зберігання обмеженої кількості даних, додаткова інформація має кудись подітися, спричиняючи переповнення в сусідніх буферах, і в такий спосіб пошкоджуючи та перезаписуючи дійсні дані, що зберігаються в них. Атаки переповнення буфера використовують цю вразливість і розміщують код, який зловмисник намагається виконати, у область переповнення буфера. Потім зловмисник перезаписує адресу повернення функцій, щоб вона вказувала назад на буфер і виконувала призначений код, щоб ініціювати певні дії, як-от створення оболонки з правами root, пошкодження файлів користувача, зміна даних або розкриття конфіденційної інформації.

Добре відома атака переповнення буфера була виявлена в Microsoft Outlook і Outlook Express у 2000 р. Через програмну помилку, зроблену Microsoft, зловмисники можуть виконувати будь-який тип коду, який вони бажають, на комп'ютері-жертві, просто надіславши повідомлення електронної пошти. На відміну від інших типових вірусів електронної пошти, користувачі не можуть захистити себе, не відкриваючи вкладені файли, оскільки механізм заголовка повідомлення програми мав недолік у реалізації, через який відправники можуть переповнювати область сторонніми даними, і процес буде активовано, коли одержувач електронної пошти завантажить повідомлення з сервера. Деякі інші добре відомі атаки на переповнення буфера для підвищення привілеїв включають Sendmail, який переповнює буфер у процедурі декодування MIME служби sendmail (SMTP) на машинах Linux, і IMAP, який використовує помилку в коді автентифікації транзакції входу служби IMAP на машинах Linux.

Типові заходи протидії атакам переповнення буфера включають написання захищеного коду, анулювання виконання стека, захищені інструменти компілятора та динамічні перевірки під час виконання. Однак жоден із них не може повністю вирішити проблему переповнення буфера через традиційну структуру, що постачається мовою програмування С, і погану практику програмування розробником програмного забезпечення.

2.3.2. Атаки з неправильною конфігурацією

Кожна система безпеки має бути налаштована адміністратором для певних параметрів, щоб відображати частину політики безпеки, яку вона виконує, і щоб гарантувати, що система забезпечує функціональні можливості, необхідні користувачам. Зазвичай це означає ввімкнення кількох параметрів. Потрібні функції працюють добре після того, як адміністратор вибере потрібний варіант. Однак проблема може виникнути, коли адміністратор через деякий час забуде вимкнути ці параметри, коли вони непотрібні. Будь-яка неправильна або слабка конфігурація

може бути використана зловмисниками, щоб пропустити бар'єр безпеки або дізнатися про порушення безпеки. Як наслідок, неправильна конфігурація або відсутність патча стали однією з найбільш значних вразливостей, з якими останнім часом стикаються підприємства, і аналіз показує, що атаки з неправильною конфігурацією все ще залишаються серед успішних атак WLAN.

Деякі добре відомі приклади атак із неправильною конфігурацією – словник і FTP-запис. Під час атаки за словником (також відомої як груба сила) зловмисник знає про ім'я користувача та робить повторні спроби вгадати пароль зі списку можливих паролів. Якщо адміністратор не змінить ім'я користувача та пароль за замовчуванням, вгадати можливі паролі стане дуже легко. Атаку за словником можна виявити та запобігти їй, налаштувавши максимальну кількість невдалих спроб входу для кожної служби або додатково вставивши певну затримку між двома послідовними спробами входу. Інша атака неправильної конфігурації, атака FTP-Write, використовує анонімний обліковий запис, що є звичайним режимом входу, який надається службами FTP. Якщо анонімний кореневий каталог FTP або його підкаталоги належать обліковому запису FTP або знаходяться в тій самій групі, що й обліковий запис FTP, і вони не захищені від запису, зловмисник зможе додати файли, наприклад, rhosts, щоб отримати локальний доступ до системи. Відстежуючи анонімні облікові записи FTP і перевіряючи будь-які спроби створити файли в кореневій папці FTP, ми можемо легко виявити такі вторгнення.

2.3.3. Атаки в умовах гонки

Процесори працюють за принципом одного окремого кроку за раз. У сучасних операційних системах багатозадачність реалізуються шляхом швидкого перемикання одного або кількох часових інтервалів. До того ж апаратні або програмні переривання можуть випереджати інші процеси. Умова змагання виникає, коли вихідні дані та/або результат процесу неочікувано і критично залежать від послідовності чи часу виконання інших конкретних конкуруючих інструкцій.

Добре відомим прикладом є стара атака входу в систему Unix, у якій, коли створюється новий процес входу, існує короткий проміжок часу, коли новий процес виконується в режимі пріоритету (ядро або кореневий режим) і ще не переключено на звичайний режим користувача. У цей час, якщо користувач-людина неодноразово натискав клавішу виходу під час входу в систему, можливо, буде запобігти переходу від root до користувача, дозволяючи людині завершити доступ до всієї системи. У старій атаці входу в систему Unix її виникнення повністю залежить від того, чи відбулася обробка ключа виходу до або після переходу у звичайний режим користувача.

Деякі інші місця для атак із умовою гонки передбачають відкриття та перевірку оболонки або пакетного файла, запуск підпрограми, перевірку пароля чи перевірку імені користувача. У деяких операційних системах виконується початкова перевірка, щоб переконатися, що оболонка або пакетний файл безпечні. Після підтвердження файл передається іншому процесу для запуску. Протягом цього короткого проміжку часу зловмисник може мати можливість замінити перевірений файл на інший, у такий спосіб дозволяючи вільне виконання команд, виконання яких має бути заборонено.

Першою лінією захисту від атак із умовами гонки є належна практика написання програмного забезпечення, наприклад, усвідомлення того, що таке атомарні та неатомарні операції в макросах прикладних програм, сценаріях оболонки та програмах, написаних локально чи користувачами. До того ж кінцеві користувачі системи повинні завжди звертати увагу на оновлення безпеки, які виправляють діри в умовах гонки.

2.3.4. Атаки «людина посередині»

Атака «людина посередині» (MITM) – це форма активного підслуховування, за якої зловмисник контролює всю розмову між жертвами, встановлюючи незалежний зв'язок із кожною жертвою, передаючи повідомлення між жертвами та змушуючи жертв повірити, що вони розмовляють безпосередньо один з одним через приватне з'єднання. Щоб здійснити успішну атаку MITM, зловмисник повинен мати можливість перехопити всі повідомлення, що передаються між двома жертвами, ввести нові та імітувати кожну кінцеву точку так, щоб інша була задоволена. Хоча більшість криптографічних протоколів або певна форма механізмів автентифікації кінцевої точки можуть запобігти атакам MITM, вони відбуваються безпосередньо за багатьох обставин, наприклад, атака на користувачів на публічній точці бездротового доступу, яку здійснює власник.

Типовий приклад атаки MITM проілюструємо так:

Припустимо, Майк хоче поговорити з Джеком. Щоб почати це спілкування, Майк повинен попросити у Джека відкритий ключ. Атака MITM відбувається, якщо третя особа, Боб, може перехопити відкритий ключ, надісланий Джеком Майку. У цьому випадку Боб може підслухати розмову, надіславши підроблене повідомлення Майку, яке нібито надійшло від Джека, разом із власним відкритим ключем. Майк, який вважає, що цей відкритий ключ належить Джеку, шифрує своє повідомлення ключем Боба та надсилає зашифроване повідомлення назад Джеку. Знову Боб перехоплює повідомлення, розшифровує його, зберігає копію та повторно шифрує за допомогою відкритого ключа, який Джек спочатку надіслав Майку. Коли Джек отримує щойно зашифроване повідомлення, він вважає, що воно походить від Майка. У цьому прикладі атаки будь-яка особиста інформація Майка та Джека може бути крадіжкою Боба, що робить можливим підвищення привілеїв.

Надійний механізм шифрування є найкращим засобом протидії атакам типу «людина посередині», наприклад, використання SSH замість Telnet, використання механізму шифрування файлів (наприклад, PGP або Entrust) або контрольних сум сеансу тощо.

2.3.5. Атаки соціальної інженерії

Соціальна інженерія – це метод атаки, за якого зловмисник намагається ввести жертву в оману шляхом агресивних переконань або використання інших навичок міжособистісного спілкування, щоб розкрити конфіденційну інформацію для доступу до захищеної критично важливої інформаційної системи. Атаку соціальної інженерії можна розділити на дві категорії, а саме фізичну та психологічну.

Типові приклади у фізичній категорії включають зони на робочому місці та смітник. Типові приклади психологічної категорії включають удаване переконання по телефону або онлайн-комунікації.

На робочому місці зловмисник може отримати інформацію для автентифікації, просто стоячи на місці та спостерігаючи, як співробітник вводить свій пароль, або просто запитуючи у працівника інформацію під приводом, що вона потрібна для усунення помилок системи. У сміттєвих контейнерах зловмисник може зібрати величезну кількість інформації, включно з потенційно телефонними книгами кредитних спілок, посібниками з політики, організаційними діаграмами, списками співробітників, системними посібниками, роздруківками конфіденційних даних або імен для входу та паролів, а також застарілим обладнанням.

Ще одним популярним методом атак соціальної інженерії є телефонний та онлайн-зв'язок. Зловмисники можуть просто зателефонувати до кол-центрів або довідкових служб і зробити вигляд, що телефонують із кредитної спілки. Обманюючи телефонну систему чи оператора, спритний зловмисник може імітувати когось, хто має владу чи релевантність, і поступово витягувати інформацію з користувача, або спритний зловмисник може маскуватися під законного користувача, якому потрібно скинути пароль. Зазвичай більшість співробітників колл-центрів і довідкових служб просто відповідають на запитання та переходять до наступного телефонного дзвінка, оскільки вони мінімально навчені у сфері безпеки, що створює величезну дірку в безпеці.

Атаки соціальної інженерії, які проводяться онлайн-комунікаціями, містять широку категорію. Найпопулярнішим є фішинг електронної пошти. Схоже, що фішингова атака на електронну пошту походить із законного джерела із запитом перевірки інформації, у якому електронний лист зазвичай містить посилання на шахрайську вебсторінку з відповідними логотипами та вмістом, який виглядає досить законним і вимагає від користувача ввести все, від контактної інформації до імені користувача та пароля.

Загалом автоматичне виявлення атак соціальної інженерії складне, оскільки вони не є вразливими місцями чи помилками в системі. Незалежно від того, скільки зусиль ви витрачаєте на технології та пристрої безпеки, найслабшою ланкою будь-якої системи безпеки зазвичай є людський фактор. Як результат, найкращим способом запобігання атакам соціальної інженерії є постійне навчання користувачів щодо можливого спуфінгу, з яким вони можуть зіткнутися.

2.4. Атаки на відмову в обслуговуванні (DoS) і розподілену відмову в обслуговуванні (DDoS)

Метою атак на відмову в обслуговуванні (DoS) є втручання в нормальну роботу мережевих служб шляхом затоплення, виснаження та перевантаження ресурсів цільової мережі або хоста. Ресурсами можуть бути пропускна здатність мережі, пропускна здатність маршрутизатора для пересилання пакетів, сервери імен, пам'ять / обчислювальна потужність на серверах або структури даних операційних систем. Під час DoS-атак зловмисники зазвичай генерують велику кількість безглуздого трафіка, наприклад, незавершені TCP-з'єднання, неправильно сформовані IP-пакети, створені ботами запити до вебсторінок та деякі інші ретельно

розроблені методи, що призводить до припинення роботи служби чи програми або перешкоджання іншим користувачам послуги або програми.

Розподілені атаки на відмову в обслуговуванні (DDoS) розробляються розподіленим способом. У реальних мережах передбачуваними жертвами DoS-атак зазвичай є потужні сервери зі швидким мережевим підключенням. Проте зловмисник DoS у більшості випадків має лише обмежені обчислювальні та мережеві ресурси. Отже, щоб провести успішну DoS-атаку проти потужного сервера-жертви, розумний зловмисник розширює матеріальне забезпечення та поширює сценарій атаки на кількох проміжних хостах. У спільноті хакерів ці проміжні хости називаються зомбі або ботами. Обчислювальні можливості кожного зомбі можуть бути такими ж обмеженими, як і власний хост зловмисника. Агрегації з добре організованою структурою для всіх ресурсів зомбі, однак, достатньо, щоб перевантажити потужні сервери з дуже швидкими з'єднаннями.

Більшість розподілених мережевих атак використовують недоліки безпеки, конфігурації або програмного забезпечення на зомбі-хостах, щоб контролювати їх з метою розподілених атак. Щоб організувати ефективну розподілену атаку, яку зазвичай важко виявити та запобігти їй, зловмисник може застосувати широкий спектр топологій, шаблони зв'язку та стратегії атак, засновані на деяких наявних характеристиках, як-от анонімність, легкість розгортання та доступність проміжних хостів. Як наслідок, стандартні визначення та спільна таксономія розподілених мережевих атак необхідні для проведення ефективного їх аналізу.

2.4.1. Підходи до виявлення атак DoS і DDoS

Виявленню DoS- і DDoS-атак в академічній літературі приділено менше уваги, ніж виявленню вторгнень. Деякі наявні системи виявлення вторгнень, як-от Snort [6], застосовують сигнатури добре відомих атак DoS і DDoS для їх ідентифікації. Сигнатури моделюються та створюються під час аналізу кожної окремої атаки, щоб унікально позначити зловмисний трафік. Інші відстежують зміни в нормальній поведінці хоста та мережі, і будь-яке відхилення буде виявлено та повідомлено як атака. Наявні системи, наприклад, Bro, застосовують такий спосіб виявлення атак DoS і DDoS на основі (статистичних) змін у звичайній поведінці програм і протоколів.

Високий обсяг трафіка біля цільового сервера може бути ознакою DDoS-атак. Такий приклад проілюстровано Лу та Траоре, коли вперше була запропонована нова метрика на основі трафіку під назвою /PTraffic шляхом вивчення основного принципу DDoS-атак. Алгоритм виявлення викидів на основі моделі суміші Гауса (GMM) використовується для аналізу значення /PTraffic і прийняття рішень про вторгнення відповідно до результату виявлення викидів. Запропонований метод було оцінено в живому мережевому середовищі, і експериментальні результати показують, що підхід може не тільки ефективно виявляти DDoS-атаки, але й забезпечувати ефективну відповідь на ці атаки. Однак тільки високого трафіка недостатньо для доказу DDoS-атак. Такий метод запропоновано Lakhina et al., де моделі трафіка спочатку створюються у високовимірному просторі, а потім аналіз головних компонент використовується для визначення лінійного підпростору нормальних моделей трафіка. Основним обмеженням цього підходу є те, що він не є

адаптивним і не може бути узагальненим. Наприклад, коли вебсервер із невеликим трафіком стає вебсайтом із високим трафіком, цей підхід повідомляє про сповіщення та позначає це як аномалію, навіть якщо DDoS-атаки не було. Отже, удосконалення цього підходу покладається на поєднання виявлення аномально високих обсягів трафіка поблизу цільового сервера з деякими правилами, які визначають, чи є відповідний трафік насправді атакою, чи просто сплеском трафіка шляхом перевірки вмісту трафіка.

Замість того, щоб вказувати DoS- і DDoS-атаки поблизу об'єкта жертви, деякі підходи намагаються виявити шаблони атак поблизу джерел трафіка (наприклад, зомбі чи бота). Такий типовий приклад запропоновано Mirkovic et al. Виявлення поблизу джерела має багато переваг, як-от надання потенціалу для повного блокування DDoS-трафіка з мережі, запобігання зараженню цільової системи-жертви або зменшення впливу DDoS-атак на всю мережу. Однак високий рівень помилок може обмежити застосування цього методу в реальності, оскільки хибна фільтрація трафіка, створеного з джерел, призведе до побічних збитків для невинних клієнтів.

Незалежно від підходів виявлення DDoS-атак поблизу джерела чи цілі, функції трафіка, показники чи шаблони завжди необхідні для того, щоб відрізнити DDoS-атаки від звичайного трафіка. Gil та ін. пропонують багаторівневе дерево онлайн-статистики пакетів, яке називається MULTOPS, для виявлення атак шляхом використання кореляції швидкості вхідних і вихідних пакетів на різних рівнях агрегації префіксів підмережі. Мережеві пристрої з MULTOPS можуть виявляти поточні DoS-атаки, що споживають пропускну здатність, на основі значної непропорційної різниці між швидкістю пакетів, що надходять до жертви чи зловмисника. Розглянемо типову аномалію мережевого трафіка. Поведінка DDoS також виявляється за допомогою набору статистичних моделей трафіка, як-от зміни кількості пакетів TCP SYN, порівняно з пакетами TCP FIN (RST); складність трафіка за Колмогоровом і спектральний аналіз трафік, коли він проходить за посиланнями. Деякі інші роботи з виявлення DoS і DDoS можна знайти в [1].

2.4.2. Запобігання та реагування на атаки DoS та DDoS

Запобігання та реагування на DoS- та DDoS-атаки спрямовані на забезпечення гарантованого доступу до ресурсів, локалізацію зловмисників та зниження інтенсивності атаки. Щоб досягти цього, вони зазвичай проводяться систематично (тобто вони, ймовірніше, можуть відбутися в місці, де розташовано хост-джерело, цільовий сервер або пристрої основної системи).

Ранні підходи в пристроях базової мережі покладаються на правила, які блокують аномальний трафік від подальшого просування. Приклади включають вхідну / вихідну фільтрацію і перевірку RPF (переадресація зворотного шляху), за якої пакети з підробленими IP-адресами не передадуться. Зовсім нещодавно протокол зворотного зв'язку і розподілений міжмережевий екран реалізовані для втручання в зловмисний трафік, ініційований DDoS-атаками. Використовуючи механізм зворотного зв'язку, маршрутизатор поблизу сервера-жертви відкидає шкідливі пакети та відповідає за доставку пакетів у місце, де відбувається атака, щоб джерела могли вжити відповідних заходів проти DDoS-атак. Основним обмеженням цього методу є те, що він не може пом'якшити відмову в обслуговуванні на цільовому хості, а

лише зменшить робоче навантаження на маршрутизатор. Розподілений брандмауер тісно пов'язаний з механізмами зворотного зв'язку. Його головна перевага полягає в тому, що пакети будуть відкидатися в кінці пункту призначення, і так для нього не споживається пропускна здатність проміжних каналів. Існують і інші підходи до запобігання та реагування, що застосовуються на базі.

Було зроблено багато спроб ідентифікувати хости-атаки та зупинити потоки зловмисного трафіка, розташовані на найближчих до цільових серверах-жертвах. Для відстеження IP-пакетів від адресата до їх джерел, незважаючи на IP-спуфінг, було запропоновано кілька методів, які зазвичай називають IP-відстеженням. Хоча сигнатури атаки можна охарактеризувати так, для ефективного визначення джерела атаки потрібне широкомасштабне розгортання через інтернет. Враховуючи рівень довіри між різними інтернет-органами, здається, що на практиці це не відбувається.

Міркович та ін. пропонує систему під назвою D-WARD, розташовану на вихідному мережевому маршрутизаторі (LAN або прикордонному маршрутизаторі), яка може автономно виявляти та видаляти потоки DDoS, що походять із цієї мережі. Хоча цей підхід здається привабливим заходом проти DDoS-атак, як ми обговорювали раніше, помилкова фільтрація джерела трафіка завдасть серйозної побічної шкоди звичайним клієнтам.

Повністю звільнитися від атак DDoS шляхом розгортання систем в окремих місцях важко, тому пропонується поєднання локальних відповідей і спільних міждоменних відповідей для ефективного вирішення проблеми затоплення DoS і DDoS. Махаджан та ін. запропонували сукупний контроль перевантаження та техніку відштовхування для ідентифікації та придушення потоків атак. Відштовхування відіграє важливу роль у реагуванні агрегату вище за течією, поки фактично не відбудеться перевантаження. До того ж виявлення агрегації трафіка базується лише на IP-адресах призначення.

2.4.3. Приклади атак DoS і DDoS

У цьому розділі пояснюється кілька прикладів справжніх добре відомих атак DoS або DDOS. Більшість із них використовується для використання пропускної здатності висхідної лінії зв'язку або пропускної здатності сервера.

– TCP-SYN Flood: Ідея атаки SYN Flood полягає у зловживанні процесом встановлення з'єднання TCP. Звичайне встановлення TCP-з'єднання передбачає тристоронній процес рукостискання. По-перше, клієнт посилає на сервер пакети запиту з бітом SYN. По-друге, сервер виділяє блок керування TCP, щоб підготуватися до отримання наступного пакету, і надсилає клієнту пакет відповіді з бітами SYN / ACK. На цьому етапі сервер залишається в напіввідкритому стані. По-третє, клієнт посилає на сервер пакети з бітом ACK. Після того, як сервер отримує пакети від клієнта, встановлюється нормальне TCP-з'єднання. Атака SYN Flood генерує велику кількість напіввідкритих TCP-з'єднань, щоб заповнити блок керування TCP, щоб вичерпати доступні ресурси. Отже, цільовий хост стає нездатним приймати більше вхідних TCP-з'єднань. Це створює відмову в обслуговуванні для законних запитів. Хоча flood-атака TCP-SYN є доволі старою, вона все ще дуже популярна.

Надсилання пакетів із сильно розподілених джерел або з підробленими IP-адресами ускладнює розробку контрзаходів. Деякі наявні інструменти, які можуть виконувати атаки TCP-SYN, включають Phatbot / Agobot, TEN2K, Stacheldraht, Trinity, TFN і Shaft. Подібною атакою з затопленням TCP-SYN є атака затоплення TCP-ACK, у якій зловмисник надсилає один SYN і кілька ACK із підробленою IP-адресою джерела, щоб перевантажити сервер. Ця атака може бути просто відфільтрована брандмауером SPI, оскільки пакети затоплення не належать до встановленого з'єднання TCP. Прикладом наявних інструментів для виконання атаки TCP-ACK є Mstream.

– ICMP / UDP Flood Attack: UDP і ICMP є протоколами без збереження стану, за допомогою яких пакети UDP або ICMP можуть передаватися між двома машинами з високою швидкістю. Велика кількість пакетів з протоколом UDP і ICMP, спрямованих на сервер, може викликати його перевантаження через масові відповіді. Підробка IP-адреси ускладнює відстеження та зупинку, але, здається, загроза зменшується завдяки використанню вихідної фільтрації. Деякі наявні інструменти атаки UDP і ICMP включають SDBot / SpyBot і Trinoo.

– Ping смерті: зловмисник надсилає на цільовий хост пакет exo-запиту ICMP, що містить дуже великий обсяг інформації, наприклад, дані великого розміру. Тоді буфер ядра цільового хоста буде переповнено, якщо він спробує відповісти на цей запит ICMP. Як наслідок, хост вийде з ладу після цієї атаки. Ping of death належить до категорії фрагментних пакетних атак, які зазвичай надсилають пакети з неправильною довжиною або зміщенням. Наразі більшість операційних систем і брандмауерів здатні обробляти та видаляти такі неправильно сформовані пакети. Trinity є типовим інструментом для подібних атак.

– Smurf: атака Smurf здійснюється за допомогою протоколу ICMP. У звичайних умовах хост-джерело A надсилає повідомлення exo-запиту хосту призначення B, а потім B надсилає exo-повідомлення A. Коли відбувається атака smurf, хост A підробляє IP-адресу за допомогою хоста C, а потім надсилає повідомлення exo-запиту на широкомовну адресу, щоб постраждалий хост C отримував усі exo-повідомлення. Як наслідок, посилання та маршрутизатори до хосту C можуть бути засмічені переливним трафіком, і C не зможе отримувати запити від інших користувачів. Атаку Smurf можна легко виявити, спостерігаючи за поведінкою трафіка, коли велика кількість exo-відповідей надсилається до певної системи без жодного exo-запиту, що надходить від системи-жертви. Типовим інструментом для виконання атаки smurf є TFN2K.

– Таблиця процесів: таблиця процесів використовує вразливість, через яку мережеві служби виділяють новий процес для кожної вхідної реалізації TCP / IP. Можливо, що таблиця процесів цільового хоста буде заповнена декількома екземплярами мережевих серверів, а отже, інші команди не можуть бути виконані на цільовому хості. Багато служб Unix, наприклад, smtp, imap і finger, вразливі до цього типу атак. Щоб уникнути цієї атаки, вразливі служби повинні мати стійкість до швидких послідовних запитів, коли навантаження на систему перевищує певний рівень.

– UDPStorm: під час атаки UDPStorm зловмисник генерує нескінченний потік даних між двома портами UDP на одному або різних хостах. Внаслідок чого хост

не може пропонувати послуги іншим користувачам, а мережа може бути перевантаженою або сповільненою. Змоделювати поведінку такої атаки доволі легко, і її можна повністю запобігти за допомогою шлюзу, який блокує підроблені IP-пакети.

– Syslogd: ця атака використовує недолік реалізації в syslogd сервера Solaris v2.5. Демон syslogd виконує DNS-пошук IP-адреси джерела вхідного повідомлення. Якщо не знайдено жодної IP-адреси, що відповідає дійсному запису DNS, служба syslogd припинить роботу з повідомленням про помилку Segmentation Fault. Подібно до атаки syslogd, атака Teardrop також використовує недоліки в реалізації IP-протоколу в процесі повторного збирання IP-сегментів, що перекриваються. Краплеподібна атака виводить систему з ладу за допомогою незвичайної фрагментації IP-пакетів, що спричиняє повний збій машини (синій екран) або втрату підключення до мережі на вразливих машинах. Зловмисник надсилає два або більше фрагментів, які неможливо зібрати належним способом, маніпулюючи значенням зміщення пакета, і змушує комп'ютер жертви перезавантажувати або зупиняти свою діяльність. Оновлення операційних систем за допомогою патчів безпеки може ефективно запобігти такому типу атак.

– Mailbomb: як типовий приклад DoS-атак на рівні програми, атаки Mailbomb переповнюють чергу SMTP-сервера або квоту на поштову скриньку окремої особи, надсилаючи величезну кількість повідомлень на поштовий сервер (наприклад, SMTP або POP). Виявлення атаки mailbomb є доволі суб'єктивним і повністю залежить від конфігурацій сервера, як-от дозволена кількість або швидкість вхідних повідомлень для певного користувача або групи користувачів, історія середнього використання поштової служби, розмір квоти тощо.

– Apache2: зловмисник надсилає запит, що містить багато HTTP-заголовків, на цільові вебсервери Apache. Сервер використовуватиме все більше процесорного часу для обробки цих заголовків, може відмовити в обслуговуванні іншим користувачам і, зрештою, призвести до збою через брак пам'яті. Виявлення цієї атаки доволі просте, тому що типовий HTTP-запит зазвичай містить не більше 20 заголовків, і будь-яке відхилення, що перевищує цю межу, є аномальним.

Наведені вище приклади показують, що механізм DoS- і DDoS-атак може бути дуже різним, починаючи від зловживання або надмірного використання законної функції протоколу чи системи (наприклад, Mailbomb і Smurf), створюючи некоректні пакети, які порушують реалізацію стеку TCP / IP (наприклад, Teardrop і Ping of Death) або використання недоліків у наявних реалізаціях програм (наприклад, Apache2 і Syslogd). Навіть більше, нові DDoS-атаки стають дедалі інтелектуальнішими, наприклад, флеш-натовпи за допомогою http «get» і «request» на вебсервері, у яких дуже важко відрізнити трафік, створений реальними нормальними користувачами, від шкідливого трафіка, створеного ботами або зомбі, оскільки вони поводяться подібно.

2.5. Атаки хробаків

Комп'ютерні хробаки відтворюють себе від системи до системи без використання файла хосту та виконують зловмисні дії, наприклад, використовують ресурси комп'ютера та вимикають систему. На відміну від вірусів, які вимагають

розповсюдження зараженого хост-файла, хробаки існують як окремі сутності й не приєднуються до інших файлів.

Хробак Morris є одним з перших комп'ютерних хробаків, поширених через інтернет [1]. Основна ідея хробака Морріса та останніх частих атак хробака в Інтернеті полягає в тому, щоб націлити на вразливості в процесах демона важливих мережевих служб за допомогою використання статичного переповнення буфера. Комп'ютерні хробаки стали однією з найсерйозніших загроз сучасному Інтернету з моменту появи першого хробака. У 2001 р. Nimda і Code Red швидко поширилися через інтернет і спричинили зараження понад 250 000 серверів, згідно з даними Групи реагування на надзвичайні ситуації з комп'ютерами (CERT). Деякі приклади нещодавніх хробаків включають Slammer, Blaster і Nachi.

Рис. 1. Зображення, згенеровані ШІ на тему атак вірусів і хробаків. URL: https://aigallery.app/

2.5.1. Моделювання та аналіз поведінки хробаків

Деякі дослідники використовують традиційні епідемічні моделі для моделювання поширення інтернет-хробаків та аналізу їх масштабів. Joukov et al. дослідили наявних хробаків і запропонували деякі методи стримування хробаків. Мур та ін. застосовували методи моделювання для вивчення поведінки розповсюдження CodeRed і пов'язаних з ним хробаків. З великою кількістю машин, заражених хробаками, симулюються різні сценарії, що включають блокування вмісту та внесення адрес у чорний список. Замість того, щоб пропонувати конкретні методи виявлення хробаків у загальнодоступному інтернеті, вони припускають, що виявлення та розповсюдження результатів виявлення займе фіксований проміжок часу. Відповідно стримування потім здійснюється шляхом блокування заражених IP-адрес або створення підписів, які запобігають будь-якому подальшому поширенню хробака з пристроїв стримування. Результати моделювання показують, що стримувати деякі типи хробаків надзвичайно важко, якщо не неможливо, за розумних припущень, оскільки жоден час реакції не буде достатньо швидким, щоб захистити від поширення епідемії. Однак існують підходи до захисту великих безмасштабних мереж від шкідливого коду, який швидко саморозповсюджується, як-от хробаки [1].

Величина загрози хробаків проаналізована Staniford et al. У своєму дослідженні вони використовують традиційну модель епідемії для вивчення поширення Code Red одразу після оригінального інциденту Code Red у 2001 р. Вони стверджують та наполягають, що добре спроєктований активний інтернет-хробак може відігравати значну роль у війні між державами, і потрібна розробка аналога Національ-

ного центру контролю захворювань (NCDC) для загроз кібербезпеці кожної країни, які стосуються комп'ютерних вірусів і хробаків.

2.5.2. Виявлення та моніторинг атак хробаків

Основна мета – виявлення та зупинка інтернет-хробаків на ранніх стадіях їх активності. Zou та ін. запропонували архітектуру системи для моніторингу хробаків, яка складається з центру попередження про зловмисне програмне забезпечення (MWC) і розподілених моніторів. Основна ідея запропонованої системи полягає в моніторингу вхідних / вихідних сканувань невикористаних адрес / портів у точках доступу мереж, а потім зібрана інформація агрегується в центральній точці для аналізу. Фільтр Калмана використовується для динамічної оцінки параметрів поширення хробака. Більше інформації про результати моделювання системи можна знайти у [1]. Інша подібна система моніторингу хробаків представлена Berk et al., яка базується на колекції повідомлень ICMP Destination Unreachable, створених маршрутизаторами для пакетів на неіснуючі IP-адреси. Такі дані ICMP фактично є тими самими даними, що й дані, зібрані моніторингом вхідного сканування в [1]. Зустрічаючи пакети на невикористані IP-адреси, маршрутизатори локальних мереж можуть або надсилати повідомлення ICMP до системи моніторинг, або надсилати таку інформацію до MWC.

Архітектура розподіленого виявлення хробаків під назвою Netbait запропонована Брентом та ін. [7] Netbait – це система обробки запитів щодо даних системи виявлення вторгнень (IDS), у якій агрегується інформація зонду, зібрана локальними IDS на географічно розподіленому наборі взаємодіючих машин. Інформацію, що зберігається на Netbait, можна ділитися на глобальному рівні за допомогою розподіленої архітектури обробки запитів за допомогою операторів SQL. Запитуючи та збираючи дані з багатьох вузлів, провайдери та мережеві адміністратори можуть мати різні погляди на поточну інфекцію, а потім автоматично створювати чорні списки для ідентифікації заражених машин. Для розгортання такої розподіленої системи в реальних мережах необхідний певний рівень довіри та координації. Однак це не завжди так через рівень довіри між різними інтернет-органами. Через це все ще залишаються серйозні побоювання щодо реальної масштабованості системи. Деякі інші підходи з цього питання можна знайти в [1].

2.5.3. Стримування хробаків

Мета стримування хробака – порушити постійне поширення хробака в кінцевих системах або в мережевій інфраструктурі після його виявлення. Стеніфорд досліджував стримування випадкових скануючих хробаків (наприклад, Code Red, Slammer або Blaster) у великомасштабній корпоративній мережі. Стверджується, що захист від черв'яків може бути дуже ефективним у запобіганні зараженню скануючими черв'яками за умови, що щільність вразливостей у мережі нижче критичного порогу. Якщо вище цього порогу, хробаки можуть поширюватися та заразити підприємства в усьому світі, але стримування може обмежити це зараження до певного рівня. Стверджується, що нещодавні хробаки є переважно випадковими скануючими хробаками, тому більшість хробаків, які з'явилися за останні три роки, можна легко стримувати.

Toth та ін. запропонували систему для виявлення та реагування на розповсюдження хробаків на основі історії підключень, записаної кожним хостом у мережі. Центральна станція моніторингу збирає всю інформацію про з'єднання, а потім будує графік з'єднань для аналізу підозрілих шаблонів, які можуть вказувати на активність хробака. Шаблони, які вказують на поширення хробаків, мають відповідати деяким особливим властивостям, наприклад: одна й та сама вразливість неодноразово використовується на різних хостах; вузол призначення певного з'єднання відкриває подібне з'єднання з тим самим портом і таким же корисним навантаженням на інший хост через короткий період часу; або скомпрометований хост, який має намір знайти нові хости-жертви, підключившись до служб за випадковими IP-адресами. Після виявлення хробаків правила брандмауера, розгорнуті на межі мережі, будуть активовані через трансляцію.

Замість отримання швидкої відповіді системи для запобігання поширенню хробаків Вільямсон пропонує стійку інфраструктуру для автоматичного уповільнення атак і виграшу часу для реакції людини. Пропоноване обмеження вірусів обмежує швидкість дозволених підключень до нових хостів. Враховуючи можливі помилки, система затримує з'єднання, призначені для нових хостів, а не розриває їх. Чим більше некорельований трафік, тим більше він затримується. За словами Вільямсона, випадкові помилки виявлення призводять до невеликих затримок, однак поведінка, схожа на вірус, буде відкладена. Зазвичай швидкість підключення до нових машин менша за певну дозволену швидкість, що забезпечує повільне розповсюдження вірусів і зменшення розповсюдження хробака на порядок, якщо його розгортати повсюдно.

LaBrea намагався уповільнити зростання хробаків на базі TCP, розгортаючи зонди на нерозподілені адреси та віртуально переводячи такі з'єднання в постійний стан. Потік, який використовується для ініціювання зонду, буде заблоковано, тому швидкість зараження хробаком зменшиться. Модифікація коду хробака може легко обійти цей підхід [1].

2.5.4. Приклади добре відомих атак хробаків

У цьому розділі ми обговорюємо характеристики деяких добре відомих атак хробаків, які нещодавно виникли внаслідок широкомасштабного спалаху в інтернеті та вразили велику кількість систем і служб.

– Blaster: також відомий як Lovsan, хробак Blaster поширився на комп'ютерах з операційними системами Windows XP і Windows 2000 у серпні 2003 р. Черв'як використовує вразливість переповнення буфера в службі DCOM RPC і закодований для запуску SYN-затоплення порту 80 вебсайта windowsupdate.com. Через це корпорація Майкрософт тимчасово закрила цей сайт з оновлення програмного забезпечення, щоб мінімізувати можливі збитки від хробака. Пізніше спалах хробака було припинено за допомогою виправлення, випущеного Microsoft.

– Nachi: хробак Nachi використовує вразливість у службі Microsoft RPC. На відміну від хробака Blaster, Nachi допомагає клієнту завантажувати та встановлювати патчі безпеки з вебсайта Microsoft, що, здається, не завдає шкоди. Однак він впливає на мережевий трафік, перезавантажує заражену систему без згоди опе-

ратора. У серпні 2003 р. спалах Nachi заразив банкомати двох основних фінансових установ, що змусило банківські IDS відключити багато банкоматів від мережі. Постефект просто еквівалентний успішній DoS-атаці.

– Slammer: хробак Slammer, запущений у січні 2003 р., заразив 75 000 жертв лише за 10 хвилин і вважається одним із швидко розповсюджуваних інтернет-хробаків. Він використовує дві помилки переповнення буфера в Microsoft SQL Server, а потім реплікується, надсилаючи пакети на випадково згенеровані IP-адреси, шукаючи ймовірні хости-жертви. Черв'як різко сповільнює інтернет-трафік, тому що: 1) багато маршрутизаторів вийшли з ладу через трафік бомбардування, створюваний зараженими серверами; і 2) більша частина пропускної здатності інтернету споживається маршрутизаторами, коли вони намагаються оновити свої таблиці маршрутизації під час збою або повернення збійних.

2.6. Атаки маршрутизації

Атаки маршрутизації використовують недоліки та вразливі місця в дизайні та реалізації маршрутизаторів. Поширена атака на маршрутизацію – це злам, який спрямований на операції обміну інформацією про маршрутизацію. Більшість складних атак або суттєвих атак DoS спочатку базуються на атаках маршрутизації на інфраструктуру IP-маршрутизації. Оскільки заміна поточної незахищеної інфраструктури маршрутизації не завжди можлива, додатковий підхід покладається на виявлення атак маршрутизації та виконання відповідних контрзаходів [28]. Існує два загальні протоколи, які широко використовуються в реалізації інтернет-маршрутизаторів, а саме протокол відкритого найкоротшого шляху (OSPF) і протокол межового шлюзу (BGP).

OSPF є, мабуть, найпоширенішим алгоритмом маршрутизації Interior Gateway Protocol (IGP) ієрархічного протоколу зв'язку, у якому дерево найкоротших шляхів обчислюється за допомогою методу, заснованого на алгоритмі Дейкстри. База даних станів зв'язків створена для підтримки дерева топології мережі в кожному маршрутизаторі, і ідентичні копії бази даних періодично оновлюватимуться шляхом заповнення на всіх маршрутизаторах у кожній зоні, що підтримує OSPF. Природний спадкоємець RIP (протоколу інформації про маршрутизацію), OSPF є менш вразливим до зовнішніх атак, оскільки він не використовує TCP або UDP, а використовує IP безпосередньо, усуваючи в такий спосіб потребу у функціях TCP або UDP. BGP є основним протоколом маршрутизації в інтернеті, який працює, зберігаючи таблицю IP-мереж, що вказують на доступність мережі як усередині автономної системи (AS – це сукупність IP-мереж під керуванням однієї сутності), так і поза AS. BGP – це векторний протокол шляху, який не використовує традиційні показники IGP. Рішення щодо маршрутизації приймаються на основі шляху, мережевих політик або правил.

2.6.1. Атаки OSPF

Деякі приклади інсайдерських атак OSPF обговорюються в [8] з метою оцінки системи виявлення вторгнень на основі протоколу реального часу, які включають атаку Seqt++, атаку максимального віку та атаку максимального порядкового номера.

– Seq++: під час атаки seq++ хакер отримує екземпляр реклами стану посилання (LSA), змінює показник стану посилання, збільшує порядковий номер LSA на 1 і перераховує контрольні суми LSA та OSPF перед тим, як змінений LSA буде повторно введено в систему маршрутизації. Інші маршрутизатори розглядатимуть модифікований LSA як більш свіжий через його збільшений порядковий номер. Змінений LSA нарешті поширюється назад до свого ініціатора, а потім ініціатор повинен протистояти (відповідно до протоколу) за допомогою нового LSA, що містить правильну інформацію про статус зв'язку та новіший порядковий номер. Якщо в цьому випадку зловмисник продовжує генерувати seq++ LSA, топологія мережі стане нестабільною.

– Максимальний вік: на відміну від атаки seq++, атака максимального віку змінює вік LSA на максимальний вік (тобто 1 година), а потім повторно вводить його в систему маршрутизації. Змінений LSA змусить усі маршрутизатори видалити відповідний LSA зі своїх таблиць топології мережі. У подібний спосіб ініціатор видаленого LSA нарешті отримає MaxAge LSA і відповідно відіб'ється за допомогою нового LSA, що містить правильну інформацію про статус посилання та більш свіжий порядковий номер. Якщо зловмисник продовжує генерувати LSA максимального віку, атака може призвести до нестабільної топології мережі.

– Максимальний порядковий номер: на відміну від атаки seq++ і максимального віку, атака максимального порядкового номера встановлює порядковий номер LSA на максимальний порядковий номер (7FFFFFFFh). Відповідно до протоколу автор відкине LSA з максимальним порядковим номером, а потім заповнить новий LSA, що містить правильну інформацію про статус зв'язку з найменшим порядковим номером: 80000001h. Якщо протокол OSPF реалізований у належний спосіб, необмежена генерація такого LSA з максимальним порядковим номером, отже, призводить до нестабільної топології мережі. Якщо видалення MaxSeq LSA пропущено в маршрутизаторі, MaxSeq LSA залишатиметься в таблиці топології кожного маршрутизатора протягом однієї години, перш ніж досягне свого максимального віку.

2.6.2. Атаки BGP

Відповідно до [1], основні цілі типової атаки BGP зазвичай включають чорну порожнину, перенаправлення, підривну дію та нестабільність. Blackholing робить префікс недоступним для великої частини інтернету. Переспрямування змушує трафік, що надходить до певної мережі, проходити інший шлях, а потім досягти скомпрометованого пункту призначення. Subversion може відстежувати та змінювати дані, коли трафік змушений проходити через певне посилання. Нестабільність може спровокувати послаблення маршруту у вихідних маршрутизаторах і, отже, спричинити перебої в з'єднанні. «Чорна порожнина», перенаправлення та нестабільність можуть бути легко досягнуті, коли скомпрометований маршрутизатор модифікує, видаляє або вводить підроблені оновлення BGP, оскільки в цьому випадку інші маршрутизатори отримують неправильне уявлення про умови та топологію мережі.

Існує 7 поширених механізмів, якими зазвичай користуються зловмисники BGP, а саме помилкові оновлення та викрадення префіксів, деагрегація, супереч-

ливі рекламні оголошення, модифікації оновлень, випадкове перемикання посилань, нестабільність і збої сеансу BGP, спричинені перевантаженням. Під час помилкових оновлень і викрадення префіксів AS оголошує або створює недійсний маршрут або префікс, яких у неї насправді немає. Деагрегація розбиває адресний блок на певні довші префікси з вищими параметрами, і внаслідок цього зловмисник може оголосити фальшиві маршрути, яким надається перевага перед легітимними маршрутами до цієї мережі. Загалом суперечлива реклама (advertising – реклама, – у контексті комп'ютерної мережі, є характеристикою маршрутизатора для трансляції мережевих оновлень і змін) є законною технікою інженерії міждоменного трафіка, за якої різні повідомлення про маршрутизацію надсилаються однією AS різним вузлам BGP. Зловмисник може використати це, перенаправляючи трафік на зловмисний маршрутизатор або на іншу AS, що призводить до перевантаження в цій AS. У модифікаціях оновлень скомпрометований маршрутизатор перенаправляє трафік, щоб створити проблеми для вихідної AS. Цю атаку важко виявити, оскільки ділові відносини та політика між AS та/або постачальниками послуг зберігаються конфіденційними та не передаються третім особам. Через ненавмисне перемикання зв'язку зламаний маршрутизатор навмисно перекриває маршрут до блоку адреси жертви так, що сусідні носії BGP пом'якшують ці маршрути. Отже, мережа-жертва залишатиметься недоступною протягом періоду згасання маршруту. Нестабільність може бути викликана багатьма факторами, як-от кількість тайм-аутів сеансів BGP через перевантаження каналу, перезавантаження маршрутизатора або неодноразові фізичні збої зв'язку, а також зміни маршрутизації чи політики через таймер MinRouteAdver і спосіб, у який BGP досліджує альтернативні шляхи. Атаки на нестабільність зазвичай здійснюються шляхом використання одного або кількох захоплених маршрутизаторів. Помилки сеансу BGP, спричинені перевантаженням, використовують помилку реалізації, яка полягає в тому, що велике перевантаження, яке переносить однорангові сеанси BGP у посиланнях, призведе до уповільнення сеансів BGP на основі TCP. Тому багато з цих сеансів буде зрештою перервано, що призведе до скасування тисяч маршрутів. Коли сеанси BGP знову резервуються, маршрутизатори повинні обмінюватися повними таблицями маршрутизації, створюючи у такий спосіб великі обсяги трафіка BGP і спричиняючи значні затримки конвергенції маршрутизації.

ВИСНОВКИ

Злам або несанкціоноване використання системи, тобто вторгнення, є поширеною атакою на мережі та комп'ютери, яка може принести дуже великі як матеріальні, так і нематеріальні втрати, а з іншого боку, зловмисники можуть отримати великі незаконні прибутки, або вирішити інші свої задачі. З розвитком мереж, збільшенням кількості комп'ютерів та користувачів, пристроїв інтернету речей та розумної інфраструктури загроза вторгнень зростає. Водночас вразливими можуть бути усі компоненти інформаційно-комунікаційних систем. І хоча деякі з описаних загроз уже відійшли у минуле, постійно з'являються нові несподівані виклики, тому однією зі стратегій є постійне відстеження поточного стану ситуації з загрозами вторгнень.

КОНТРОЛЬНІ ЗАПИТАННЯ

1. Що таке вторгнення? Дайте кілька визначень з огляду на мету вторгнення.
2. Що таке виявлення вторгнення?
3. Ознаки наявності чи наслідків вторгнення.
4. Що таке запобігання вторгненню?
5. Чому системи виявлення та запобігання вторгненню об'єднують?
6. Основні об'єкти, на які можуть здійснюватися атаки вторгнення.
7. Що роблять інструменти (системи) виявлення вторгнень (IDS)?
8. За якими ознаками виявляються вторгнення?
9. З якими вадами чи особливостями пов'язана можливість здійснення вторгнень?
10. Які загальні математичні методи виявлення вторгнень?

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Ghorbani A. A., Lu W., Tavailaee M. Network Intrusion Detection and Prevention. Concepts and Techniques. Springer Science+Business Media, LLC 2010. 242 p.
2. Технології захисту інформації в інформаційно-телекомунікаційних системах: навч. посіб. / А. В. Жилін, О. М. Шаповал, О. А. Успенський; ІСЗЗІ КПІ ім. Ігоря Сікорського. Київ: КПІ ім. Ігоря Сікорського, Вид-во «Політехніка», 2021. 213 с.
3. Янко А. С., Макаренко О. І. Концепція системи виявлення та запобігання вторгнень до мережі. *Системи управління, навігації та зв'язку.* 2022. № 2. С. 59–67.
4. Толюпа С., Лукова-Чуйко Н., Шестак Я. Засоби виявлення кібернетичних атак на інформаційні системи. *Інфокомунікаційні технології та електронна інженерія*. 2021. Вип. 1, № 2. С. 19–31.
5. Огляд математичних методів у системах виявлення та попередження кіберзагроз / Н. О. Лисенко, В. Б. Мазуренко, А. І. Федорович, Д. С. Астахов, В. І. Стаценко. *Актуальні проблеми автоматизації та інформаційних технологій.* 2021. Том 25. С. 91–102.
6. SNORT Users Manual 2.9.16. The Snort Project. URL: http://manual-snort-org.s3-website-us-east-1.amazonaws.com/ 2024
7. Brent N., Lee G., Weatherspoon H. Netbait: a distributed worm detection service, Tech. Report IRB-TR-03-033, Intel Research Berkeley. September 2003.
8. Chang H.-Y., Wu S. F., Jou Y. Frank. Real-time protocol analysis for detecting link-state routing protocol attacks. *ACM Transactions on Information and System Security.* 2001 (TIS-SEC). № 4(1). P. 1–36.