Хмельницький національний університет
Факультет інформаційних технологій
Кафедра комп'ютерної інженерії та інформаційних систем

ДИПЛОМНА РОБОТА МАГІСТРА

Галузь знань 12 – Інформаційні технології

Спеціальність 123 –Комп'ютерна інженерія

на тему «Метод побудови резильєнтних IT-інфраструктур в умовах здійснення атак мережного типу»

КвРКІ 16005.16.18 ПЗ

Виконав: студент 2 курсу, група КІ2м-20-1 _____ Заграй А.О.
Підпис Ініціали, прізвище

Керівник доктор техн. наук, професор _____ Лисенко С.М.
Науковий ступінь, вчене звання Підпис Ініціали, прізвище

До захисту допускаю:
Зав. кафедри КІІС, д.т.н., проф.
Т.О. Говорущенко
__ _____ 2022 р.

Хмельницький, 2022

ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

Факультет ІНФОРМАЦІЙНИХ ТЕХОЛОГІЙ

Кафедра КОМП'ЮТЕРНОЇ ІНЖЕНЕРІЇ ТА ІНФОРМАЦІЙНИХ СИСТЕМ

Освітній рівень МАГІСТР

Галузь знань 12 ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ

Спеціальність 123 КОМП'ЮТЕРНА ІНЖЕНЕРІЯ

Освітня програма ОСВІТНЬО-НАУКОВА ПРОГРАМА «КОМП'ЮТЕРНА ІНЖЕНЕРІЯ ТА ПРОГРАМУВАННЯ»

ЗАТВЕРДЖУЮ

Зав. кафедри Т.О.Говорущенко

" 01 " 09 2021 р.

ЗАВДАННЯ
НА ДИПЛОМНИЙ ПРОЕКТ (РОБОТУ)

Заграй Альоні Олексіївні

Прізвище, ім'я, по батькові студента

Тема проекту (роботи) Метод побудови резильєнтних ІТ-інфраструктур в умовах здійснення атак мережного типу

1. Керівник проекту (роботи) д.т.н., професор Лисенко С.М.

Прізвище, ім'я, по батькові, науковий ступінь, вчене звання

Затверджена наказом ректора університету додаток від 06.01.2022 р. № 1

2. Строк подання студентом проєкту (роботи) на кафедру 03.05.2022

3. Вихідні дані до проекту (роботи) Завдання на дипломне проектування

4. Зміст пояснювальної записки (перелік питань, які потрібно розробити)

Оцінка резильєнтності ІТ-інфраструктур в умовах здійснення атак мережного типу

Моделювання показників безпеки для оцінки резильєнтності ІТ-інфраструктур в умовах здійснення атак мережного типу

Удосконалений метод побудови резильєнтних ІТ-інфраструктур в умовах здійснення атак мережного типу

Програмно-апаратний засіб забезпечення побудови резильєнтних ІТ-інфраструктур в умовах здійснення атак мережного типу

5. Перелік графічного матеріалу (із зазначенням обов'язкових креслень)

6. Консультанти розділів дипломного проекту (роботи)

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Нормоконтроль	Лисенко С.М., доцент кафедри КІСП		
Антиплагіат	Нічепорук А.О., старший викладач кафедри КІСП		

7. Дата видачі завдання «06» вересня 2021 р.

КАЛЕНДАРНИЙ ПЛАН

№з/п	Назва етапів (розділів) кваліфікаційної роботи	Термін виконання етапів проекту (роботи)	Примітка
1	Вибір напряму дослідження та узгодження тематики КРМ з керівником	06.09.2021	виконано
2	Ознайомлення з предметною областю; формулювання мети та задач дослідження; визначення об'єкта та предмета дослідження	05.10.2021	виконано
3	Робота над розділом 1 – аналіз відомих моделей, методів за темою; постановка задачі	05.11.2021	виконано
4	Робота над розділом 2 – розробка моделей і методів для вирішення поставленої задачі	05.12.2021	виконано
5	Робота над науковою статтею	15.12.2021	виконано
6	Робота над розділом 3 – розробка алгоритмів та технологій, їх аналіз	17.02.2022	виконано
7	Робота над розділом 4 – проектування ПЗ для вирішення поставленої задачі	05.04.2022	виконано
8	Узгодження отриманих; оформлення пояснювальної записки згідно вимог	18.04.2022	виконано
10	Попередній захист ДРМ	26.04.2022	виконано
11	Захист ДРМ на засіданні ЕК	06.05.2022	

Студент _____ А.О. Заграй
Підпис Ініціали, прізвище

Керівник проекту (роботи) _____ С.М. Лисенко
Підпис Ініціали, прізвище

РЕФЕРАТ

Тема дипломної роботи: Метод побудови резильєнтних ІТ-інфраструктур в умовах здійснення атак мережного типу

Автор роботи: Заграй А.О.

Керівник роботи: д.т.н., професор Лисенко С.М.

Пояснювальна записка: 89 с., 18 рис., 5 табл., 3 дод., 88 джерел.

ІТ-ІНФРАСТРУКТУРИ, КІБЕРАТАКА, МЕРЕЖНІ АТАКИ, РЕЗИЛЬЄНТНІСТЬ.

Об'єктом дослідження є процес побудови резильєнтних ІТ-інфраструктур в умовах здійснення атак мережного типу.

Предметом дослідження є моделі, удосконалений метод та програмно-технічні засобами побудови резильєнтних ІТ-інфраструктур в умовах здійснення атак мережного типу.

Метою дипломної роботи є забезпечення резильєнтного функціонування ІТ-інфраструктур в умовах здійснення атак мережного типу.

Для розв'язання поставлених задач використовувалися такі методи, як основні положення системного аналізу, моделювання, нечіткої логіки, методів аналізу даних, теорії математичної статистики, теорії дискретної математики.

Наукова новизна отриманих результатів:

– Удосконалено метод побудови резильєнтних ІТ-інфраструктур в умовах здійснення атак мережного типу, який на відміну від відомих враховує розрахунках ймовірності мережевої атаки на ІТ-інфраструктуру;

– Набули подальшого розвитку програмно-технічні засоби побудови резильєнтних ІТ-інфраструктур в умовах здійснення атак мережного типу, які забезпечують резильєнтне функціонування ІТ-інфраструктур в у мовах атак

Практична значимість отриманих результатів полягає у розробці засобів оцінки резильєнтності ІТ-інфраструктур в умовах здійснення атак мережного типу, які забезпечуватимуть таку оцінку з високою точністю.

Зв'язок роботи з науковими програмами, планами, темами. Дослідження, представлені у кваліфікаційній роботі, проводились в рамках держбюджетної НДР Хмельницького національного університету 1Б-2021 «Самоорганізована розподілена система виявлення зловмисного програмного забезпечення в комп'ютерних мережах» (ДР № 0121U109936) 2021-2022 рр.

За темою дипломної роботи опубліковано тези у Всеукраїнській науково-практичній конференції Актуальні Проблеми Комп'ютерних Наук (АПКН-2021) . Було взято участь у Всеукраїнській науково-практичній конференції Актуальні Проблеми Комп'ютерних Наук.

Дата 29.04.22 Підпис студента 

ЗМІСТ

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ

ЗПЗ – зловмисне програмне забезпечення

IDS – система виявлення вторгнень

CVSS – Common Vulnerability Scoring System, загальна система оцінки вразливостей

ЖЦ – життєвий цикл

ПК – персональний комп’ютер

ІС – інформаційна система

КС – комп’ютерна система

APT – розширена стійка атака

КМ – комп’ютерна мережа

ВСТУП

Комп’ютерні мережі сьогодні відіграють роль нервової системи в багатьох критично важливих інфраструктурах, державних і військових організаціях і підприємствах. Захист таких критично важливих мереж означає більше, ніж просто виправлення відомих вразливостей і розгортання брандмауерів або IDS.

Належні показники необхідні для оцінки рівня безпеки мереж і надання рішень, що покращують безпеку. Однак, без урахування невідомих вразливостей нульового дня, побудова резильєнтних IT-інфраструктур в умовах здійснення атак мережного типу є неможливою.

Тому задача побудови резильєнтних IT-інфраструктур в умовах здійснення атак мережного типу є сьогодні надзвичайно важливою.

Метою кваліфікаційної роботи магістра роботи є забезпечення резильєнтного функціонування IT-інфраструктур в умовах здійснення атак мережного типу.

Поставлена мета досягається розв’язанням наступних задач:

1. Розглянути поняття резильєнтного функціонування та оцінка резильєнтності іт-інфраструктур в умовах здійснення атак мережного типу,
2. Дослідити відомі методи оцінки резильєнтності мережі в умовах атак.
3. Проаналізувати та дослідити показники резильєнтності мережі в умовах здійснення атак, зокрема показник різноманітності мережі, метрику покриття атаки мережі.
4. Здійснити моделювання показників безпеки для оцінки резильєнтності іт-інфраструктур в умовах здійснення атак мережного типу.
5. Виконати формальне моделювання різноманітності мережі
6. Побудувати модель різноманітність на основі найменших зусиль для атаки, модель імовірнісної різноманітності мережі та імовірнісну модель мережевого різноманіття.
7. Розробити удосконалений метод побудови резильєнтних IT-інфраструктур в умовах здійснення атак мережного типу.

8. практично реалізувати програмно-апаратних засобів удосконаленого методу побудови резильєнтних ІТ-інфраструктур в умовах здійснення атак мережного типу з тестуванням на проникнення.

Об'єкт дослідження – процес побудови резильєнтних ІТ-інфраструктур в умовах здійснення атак мережного типу.

Предмет дослідження – моделі, удосконалений метод та програмно-технічні засобами побудови резильєнтних ІТ-інфраструктур в умовах здійснення атак мережного типу.

Методи дослідження. Для розв’язання поставлених задач використовуються основні положення системного аналізу, методів аналізу даних, теорії дискретної математики, теорії комп’ютерних мереж та систем.

Наукова новизна отриманих результатів:

1. Удосконалено метод удосконалений метод побудови резильєнтних ІТ-інфраструктур в умовах здійснення атак мережного типу, який на відміну від відомих враховує розрахунках ймовірності мережевої атаки на ІТ-інфраструктуру, ймовірності атаки на основі загальної системи оцінки вразливостей, і який забезпечують резильєнтне функціонування ІТ-інфраструктур в у мовах атак.

2. Набули подальшого розвитку програмно-технічні засоби побудови резильєнтних ІТ-інфраструктур в умовах здійснення атак мережного типу, які забезпечують резильєнтне функціонування ІТ-інфраструктур в у мовах атак.

Практична цінність. В результаті виконаного наукового дослідження було розроблено програмно-технічні засоби побудови резильєнтних ІТ-інфраструктур в умовах здійснення атак мережного типу, які забезпечують резильєнтне функціонування ІТ-інфраструктур в у мовах атак.

Зв’язок роботи з науковими програмами, планами, темами. Дослідження, представлені у кваліфікаційній роботі, проводились в рамках держбюджетної НДР Хмельницького національного університету 1Б-2021 «Самоорганізована розподілена система виявлення зловмисного програмного забезпечення в комп’ютерних мережах» (ДР № 0121U109936) 2021-2022 рр.

За темою дипломної роботи опубліковано тези у Всеукраїнській науково-практичній конференції Актуальні Проблеми Комп’ютерних Наук (АПКН-2021) [1]. Було взято участь у Всеукраїнській науково-практичній конференції Актуальні Проблеми Комп’ютерних Наук.

1 ОЦІНКА РЕЗИЛЬЄНТНОСТІ ІТ-ІНФРАСТРУКТУР В УМОВАХ ЗДІЙСНЕННЯ АТАК МЕРЕЖНОГО ТИПУ

Сьогодні сучасні комп'ютерні мережі відіграють надзвичайно велику роль у багатьох ІТ-інфраструктурах: критичних, урядових та військових організаціях та підприємствах [1-3].

Захист таких критично важливих мереж означає більше, ніж просто виправлення відомих вразливостей та розгортання брандмауерів або IDS.

Відповідні показники необхідні для оцінки рівня безпеки мереж та забезпечення розширених рішень щодо безпеки, а саме рівня резильєнтності іт-інфраструктур – здатності передбачати мережні кібератаки, протистояти їм та мати здатність до відновлення після їх здійснення [1-5].

Однак, не враховуючи невідомих вразливостей нульового дня (англ. Zero-day/0day), показників безпеки недостатньо, щоб чисельно відобразити справжній рівень безпеки мережі [4-7].

Сучасні комп'ютерні мережі відіграють роль нервової системи в багатьох критичних інфраструктурах, державних і військових організаціях, на підприємствах. Захист такої критичної мережі місії означає більше, ніж просто виправлення відомих вразливостей та IDS. Резильєнтність мережі проти потенційних атак нульового дня, що експлуатують невідомі вразливості, не менш важлива. Гучний інцидент ботнету Stuxnet, в якому використовуються чотири нульові денні вразливості для цільових промислових систем управління, наочно продемонстрував реальне значення оцінки та підвищення безпеки мереж від атак нульового дня.

Оскільки не можна поліпшити те, що не можна виміряти, відсутність ефективних показників безпеки є ключовою перешкодою на шляху до розробки систематичних підходів до оцінки і, отже, підвищення відносної ефективності рішень безпеки. У усталених областях, таких як фізична наука, першим важливим кроком у напрямку вивчення будь-якого предмета є пошук принципів чисельного

розрахунку та практичних методів вимірювання певної якості, пов'язаної з ним. Поняття метрики придумано в різних формах в різних контекстах. Метрики можуть бути ефективним інструментом для менеджерів безпеки, щоб визначити ефективність різних компонентів своїх програм безпеки; безпека конкретної системи, продукту або процесу. Метрики також можуть допомогти визначити рівень ризику в тому, щоб не вжити певних заходів, і таким чином надати керівництво в пріоритетності виконання.

Загалом показники безпеки поділяються на чотири категорії [8] показників безпеки процесів, метрики призначені для вимірювання процесів і процедур, показники мережевої безпеки, метрики, які оцінюють рівень безпеки на цілих мережах, показники безпеки програмного забезпечення, метрики, які вимірюють можливість мають потоки в програмних додатках, і люди метрики безпеки, метрики, які включають експертів-людей:

1. Показники безпеки процесу: цей тип показників призначений для вимірювання процесів і процедур. Це передбачає високу корисність політики і процесів безпеки, але зв'язок між метриками і рівнем безпеки чітко не визначений, наприклад, відсоток системи з перевіреним контролем безпеки, або кількість виявлених ризиків і їх тяжкості.

2. Показники мережевої безпеки: Існуючі показники в цій категорії, як правило, залежать від конкретних продуктів (наприклад, пожежних стін, IDS, перевірки моделі) і широко використовуються для мережевих систем. Ці показники, як правило, ілюструються діаграмами, наприклад, для кількості заблокованих вірусів, кількості застосованих патчів і аналізу графів.

3. Показники безпеки програмного забезпечення: Цей тип показників зазвичай має багато обмежень (наприклад, номер рядка коду (LOC), точки функцій (FPs), висока складність) і є контекстно-чутливими, екологічними та архітектурно-залежними, наприклад, розміром та складністю, дефектами (тяжкість, тип) з плином часу.

4. Показники безпеки людей: Ця категорія включає показники безпеки, в яких

беруть участь експерти-люди.

Існує багато дослідницьких зусиль щодо показників мережевої безпеки (глава 2 більш детально розгляне пов'язану роботу). Примітно, що в [9] запропоновано метрику, засновану на концепції MTTC (Середній час до компромісу) як вимірювання середніх зусиль, необхідних для зловмисників, щоб скомпрометувати мережу сукупні метричні оцінки неподільних дефектів, отриманих із Загальної системи оцінювання вразливостей (CVSS) як показник для загальної безпеки мереж.

Однак одним з ключових обмежень більшості існуючих зусиль є відсутність розгляду невідомих вразливостей нульового дня. Існуючі зусилля щодо показників мережевої безпеки зазвичай призначають числові оцінки вразливостям на основі відомих фактів про вразливості. Така методологія більше не застосовується, якщо розглядати атаки нульового дня. Насправді, популярна критика минулих зусиль щодо показників безпеки полягає в тому, що не можливо мати справу з невідомими вразливостями, які, як правило, вважаються незмірними [10]. На жаль, з урахуванням невідомих вразливостей, показник безпеки буде мати лише сумнівне значення в кращому випадку, оскільки він може визначити конфігурацію мережі, щоб бути більш безпечною, ніж вона є насправді. Таким чином, потрапляємо в агностицизм, що безпека не підпадає під сумнів, поки не зможна виправити всі потенційні спалахи безпеки; хоча на той час, більше не потрібні показники безпеки.

1.1 Відомі методи оцінки резильєнтності мережі в умовах атак

Сьогоднішні проблеми безпеки в мережевій системі викликані не тільки поодинокими вразливостями, але і поєднанням мульти-вразливостей між декількома хостами [11]. Перш ніж визначати показник безпеки на мережевому рівні, попередня умова полягає в тому, щоб зрозуміти, як вразливості хостів можуть бути об'єднані як шляхи атаки. Граф атаки є добре встановленою моделлю для цієї мети. Навіть добре керовані мережі все ще можуть зіткнутися з проблемою захисту від мережевих атак. Багато служб безпечніше, коли працюють поодинці, однак,

кілька служб можуть допомогти один одному виконати уразливості, що експлуатуються. Вивчення або сканування хоста індивідуально є неефективним в оцінці безпеки мереж; відносини між хостами і вразливості повинні розглядатися для вимірювання безпеки. З цією метою в [12] запропоновано метод, використовуючи перевірку моделі для аналізу можливостей мережі vulner. Перевірка моделі розглядається як система генерації шляху атаки з вхідною інформацією з мережевого середовища. Чотири основні описи вибираються як входи в перевірку моделі, щоб з'ясувати, чи існує атакуючий шлях. З визначенням передумов і цілей, модельні шашки дають конкретні шляхи атаки досліджуваних мережевих систем. Це перша робота, пов'язана з атакуючими графами, яка відкриває нову еру аналізу мережевої безпеки. Спостерігаючи, що знайти один шлях атаки в мережевій системі недостатньо, в [13] пропонують ідею пошуку всіх атакуючих шляхів в інтегрованій системі. Ця робота використовує алгоритм перевірки моделі для заміни колишнього ручного методу графа атаки. Пропонується кілька алгоритмів, які допомагають генерувати графи атаки більш ефективно, [14] зменшити складність цього аналізу від експоненціального до многочлена, припускаючи, що зловмиснику ніколи не потрібно повертатися, щоб відновити втрачений привілей.

В [15] описано інструментарій, який використовується для переналаштування наборів програм та скриптів оболонки, щоб допомогти адміністраторам перевірити потенційні дірки безпеки в системах. В роботі будують систему паролів, яка використовується для виявлення потенційних проблем безпеки і звітування перед системними адміністраторами. В [16] представлено графічний метод аналізу вразливостей мережі. Ця система аналізу визначає дуже ймовірні шляхи атаки, які можуть бути використані для тестування системи при застосуванні певних змін, наприклад, зміни конфігурації, розгортання IDS. В [17] представлено корисну модель-шашки, яка допоможе реалізувати графи атаки та перевірити показники мережевої безпеки. В [18] розроблено інструмент графа атаки з графічним інтерфейсом для зв'язування вразливостей і налаштування графів атак. Крім того, в

[19] подано інструментарій для генерації та аналізу графів атак.

1.2 Показники резильєнтності мережі в умовах здійснення атак

В [20] подано моделювання мінімальних зусиль, необхідних зловмиснику, щоб використовувати вразливості як метрику для оцінки рівня безпеки систем. В [21] запропонувати мінімальні зусилля, необхідні для виконання кожного вторгнення мережного типу як метрика. Щоб розглядати шлях атаки як показник безпеки, в [22] запропоновано метрику для вимірювання сили безпеки мережі, що класифікує силу найслабшого зловмисника (який має найнижчі здібності, щоб скомпрометувати цю систему). За допомогою індивідуального алгоритму цей показник обчислює мінімальні набори необхідних початкових атрибутів для зловмисника з найнижчою здатністю успішно скомпрометувати систему. Ці метрики розроблені з точки зору нападників.

Використання найменших можливостей зловмисника в якості вимірювання для систем не може представляти середню здатність нападників. В [23] використовувати середній час до компромісу, які навчаються на фізичній безпеці, як метрика безпеки для вимірювання систем. Особа, яка приймає рішення, дізнається про поліпшення, просто розрахував MMTC, який чим довше, тим безпечніше. Розглядаючи інвестиції та скорочення очікуваних збитків, в [24] розроблено кількісну метрику управління ризиками, яка використовує зменшення очікуваних втрат для вимірювання успіху інвестицій у інформаційну безпеку.

Більшість з перерахованих вище показників відносно прості і розроблені на ранній стадії дослідження метрики мережевої безпеки. Прості метрики легко застосовувати і розуміти, але можуть не підходити для складних і постійно мінливих мережевих середовищ. В [25] визначено показник стійкості до атаки для оцінки та порівняння безпеки різних конфігурацій мережі. Показник стійкості до атаки вимірює здатність систем захищатися від атак, що може запропонувати адміністраторам потенційні кращі конфігурації в мережевих системах.

В [26] запропонувати агрегування ймовірності вразливості з загальної системи підрахунку балів вразливостей (CVSS). Ймовірність уразливості вказує на успішні показники експлуатації для даної вразливості. Однак ця робота фокусується тільки на вразливостях, які ігнорують взаємозалежність всередині графа атаки. Щоб вирішити цю проблему, в [27] подано модель на основі баєсових мереж для агрегування вразливостей всередині мережевих систем. Баєсові мережі визначаються змінними і дугами, що представляють умовну незалежність серед змінних. Однак формальне поняття цієї метрики не визначено в даній роботі. Також розширено роботу від баєсових мереж до динамічних баєсових мереж, щоб впоратися з постійно мінливими мережевими середовищами. Хоча динамічна ситуація має деякі обмеження, чітке визначення в баєсовій мережі надихає пізніше працювати в метриках мережевої безпеки.

В [28] запропоновано метричні, обчислювальні мінімально необхідні ресурси для компрометації системи для цілей зміцнення. Попередні методи загартовування зазвичай порушують шляхи атаки, щоб зміцнити мережу, однак ця робота зосереджена на пошуку початкового необхідного набору для нападників. Таким чином, дії атаки буде легко перерваний, вимкнувши початкові умови поліпшити це рішення, визначивши чітку нотацію мінімальної вартості затвердіння мережі на основі графів атак.

Показники безпеки також розробляються для конкретних застосувань, таких як інформаційно-теоретичні метрики для вимірювання ефективності IDSs [29] побудувати формальну структуру з використанням теорії інформації для аналізу та кількісної оцінки ефективності IDS. Робота представляє формальну модель IDS, аналізуючи інформаційно-теоретичний підхід.

Подібні роботи існують і в інших сферах, наприклад, деякі з принципів проектування пропонуються для розробки метрик для довіри [30]. Автентифікація сутностей у великомасштабній системі завжди використовує автентифікацію, що означає, що кожна сутність може автентифікувати наступний шлях. Ця технологія була поширена на кілька шляхів. В [31] запропоновано метрики, щоб отримати

довіру, надану набором шляхів.

1.3 Показник різноманітності мережі

Дослідження показників безпеки останнім часом привернуло багато уваги. На відміну від існуючої роботи, спрямованої на вимірювання обсягу мережевої безпеки [32] різноманітність мережі Метрики зосереджені на різноманітності як одній конкретній властивості мереж, які можуть вплинути на керованість. Тим не менш, робота запозичена з популярної метрики безпеки програмного забезпечення, поверхні атаки [33], загальної ідеї зосередження уваги на інтерфейсах (віддалено доступні ресурси), а не на внутрішніх деталях (наприклад, локальних додатках). Найменш атакуюча метрика складності дайвера,заснована на зусиллях, походить від метрики безпеки k-zero day [34] і імовірнісна метрика різноманітності базується на метриці ймовірності атаки. Інша помітна робота оцінює показники безпеки проти реальних атак у контрольованому середовищі [35], що забезпечує майбутній напрямок для кращої оцінки роботи. Одне з обмежень роботи полягає у високій складності аналізу графа ресурсів; моделі високого рівня залежностей ресурсів можуть забезпечити більш грубі, але більш ефективні рішення для моделювання різноманітності.

Ідея використання дизайнерської різноманітності для відмовостійкості досліджувалася вже давно. Підхід програмування N-версії генерує $N \geq 2$ функціонально еквівалентних програм і порівнює їх результати для визначення несправної версії [36], з метриками, визначеними для вимірювання різноманітності програмного забезпечення та несправностей. Основне обмеження дизайнydiversity полягає у високій складності створення різних версій, що може не виправдати вигоди. Використання різноманітності дизайну як механізму безпеки також привернуло багато уваги. Загальні принципи різноманітності дизайну, як показано, застосовуються і до безпеки в [37]. Система N-Variant extends ідея N-v система для виявлення вторгнень [39], а концепція поведінкової відстані виводить її за рамки

вихідного голосування. Різні методи рандомізації були використані для автоматичного створення різноманітності [40].

На додаток до різноманітності дизайну та генерованої різноманітності, недавня робота використовує опортуністичні різноманітні, які вже існують серед різних програмних систем. Оцінюється практичність використання різноманітності ОС для толерантності до вторгнень та продемонстровано доцільність використання опортуністичного різноманіття, вже існуючого між різними ОС для терпимості вторгнень [35]. Різноманітність також застосовується до толерантних систем вторгнення, які зазвичай реалізують деякі види візантійської толерантної розлому (BFT) реплікації як рішення для відмов tolerance [36]. Загальна архітектура для впровадження вторгнених веб-серверів на основі принципів надмірності та диверсифікації вводиться в [37]. Різноманітність компонентів, що не є придатними для заповнення (COTS), використовується для забезпечення неявної еталонної моделі, замість явної моделі, яка зазвичай потрібна, для виявлення аномалій на веб-серверах/ Diversity може відігравати важливу роль у вирішенні різних проблем безпеки в хмарних обчисленнях, таких як використання різних органів влади для фіциентної розшифровки та відкликання в хмарному сховищі та використання різноманітних політик доступу для підвищення безпеки хмарних даних.

1.4 Метрика покриття атаки мережі

Концепція поверхні атаки спочатку пропонується для конкретного програмного забезпечення і вимагає специфічної для домену експертизи для формулювання та реалізації [38]. Пізніше концепція генер-алезується з використанням формальних моделей і стає застосовною до всього програмного забезпечення [39]. Крім того, він уточняються і застосовується до великомасштабного програмного забезпечення, і його розрахунку можуть допомогти автоматично згенеровані графи викликів. Поверхня атаки привернула значну увагу протягом багатьох років. Він використовується як метрика для оцінки

системи передачі повідомлень Android, в ядрі, а також служить основою в Moving Target Defense, яка в основному спрямована на зміну поверхні атаки з плином часу [40]. Інші прагнуть розширити сферу застосування цієї концепції в інших областях, таких як шестисторонні поверхні атаки між користувачами, сервісами та хмарними системами, а також наближення ударної поверхні для сучасних автомобілів [41]. Дослідження з автоматизації розрахунку поверхні атаки є ще однією цікавою областю, наприклад, COPES використовує статичний аналіз з байт-коду для розрахунку поверхні атаки та для забезпечення softw на основі дозволів[9]. Трасування стека зі звітів про аварійне завершення роботи користувача використовується для автоматичного наближення поверхні атаки [42]. Незважаючи на такий величезний інтерес до концепції поверхні атаки, наскільки відомо, існує мало роботи над формально визначенням поверхні атаки на мережевому рівні. Також досліджено кореляцію між поверхнею атаки та вразливостями, наприклад, використання точок входу поверхні атаки та досяжності для оцінки ризику вразливості. Дослідження про зв'язок між поверхнею атаки і щільністю вразливості наведено в [43], хоча результат заснований лише на двох випусках HTTP-сервера Apache, що дає мало ключу до загального існування такої кореляції. Що стосується показників безпеки в цілому, існують зусилля зі стандартизації щодо вразливості якessment, включаючи Загальну систему підрахунку балів вразливостей (CVSS) [44], яка вимірює вразливості в ізоляції. Зусилля NIST щодо стандартизації показників безпеки також наведені в [45]. Дослідження показників безпеки привернуло багато уваги останнім часом. Більш ранні роботи включають метрику з точки зору часу і зусиль на основі марковської моделі [46]. Зовсім недавно пропонується кілька показників безпеки шляхом об'єднання показників CVSS на основі графів атаки [47]. Мінімальні зусилля, необхідні для виконання кожного вторгнення мережного типу, використовуються як метрика в [48]. Середня метрика часу на компроміс пропонується на основі моделі державного простору хижака (SSM), що використовується в біологічних науках. Хоча ці показники в основному розроблені для відомих вразливостей, менше роботи здатні впоратися з атаками

нульового дня. Кілька винятків включають емпіричне дослідження загальної кількості вразливостей нульового дня, доступних в один день на основі існуючих даних [49], зусилля з замовлення різних додатків в системі серйозністю кон-послідовностей, що мають вразливість одного нульового дня, а останнім часом k-zero day модель безпеки [50] і модель мережевого різноманіття обидва намагаються змоделювати ризик вразливостей нульового дня, але їх загальне обмеження полягає у відсутності можливостей для розрізнення ймовірності того, що різні ресурси мають такі вразливості. Метрика поверхні атаки net-work відрізняє ресурси з ймовірністю атаки від трьох вимірів концепції поверхні атаки і агрегує ймовірність атаки на рівні мережі, щоб змоделювати ризик вразливостей нульового дня, які вирішують обмеження від попередніх робіт.

1.5 Моделі вразливостей

У літературі були вивчені дві основні моделі виявлення вразливостей (VDM): одна зосереджена на вивченні особливостей, які корелюють з вразливими компонентами в додатку з м'якою посудом; інша зосереджена на використанні математичних моделей для відповідності моделі виявлення вразливості історичним даним для прогнозування майбутнього числа вразливостей для одного додатка.

В [51] проаналізовано можливість прогнозування вразливого компонента у Windows Vista за допомогою логістичної регресії для five груп метрик, відбивних, складних, охоплення, залежності та організаційної структури компанії. Бінарні результати були оцінені з десятикратною перехресною перевіркою, яка дає точність нижче 67% і нагадує нижче 21%. В [52] вивчено показники активності розробників та вразливості програмного забезпечення. Точність і відкликання з баєсової мережевої предиктивної моделі становить від 12%-29% до 32% до 56% відповідно. В [53] вивчено взаємозв'язок між метриками програмного забезпечення та вразливими компонентами в 14 веб-додатках з відкритим кодом. Кореляція рангу Спірмена обчислюється між метриками і ресурсами безпеки in-dicator (SRI), який

визначається автором і отриманий зі сканерів безпеки. В [54] виконано бінарну фікацію classi за допомогою логістичної регресії з десятикратною перехресною перевіркою для аналізу взаємозв'язку між складністю, кодовою та розробнико-діяльністю (CCD) метриками та вразливостями. Також вивчено зв'язок між 18 метриками складності, метриками коду five та метрикою історії несправностей та вразливими компонентами. Відкликання і точність цього дослідження 83% і 11% відповідно.

В [55] проаналізовано вплив мета-даних у сховищах коду за допомогою кодових мет-ріків для прогнозування вразливих комітів. Точність VCCFinder становить 60%, коли відкликання становить 24%. Також вивчено зв'язок між м'якими wє метриками і vulner- здатною функцією в існуючих атаках мережного типу. Всього в їх дослідженні було розглянуто 183 вразливості з Національної бази даних вразливостей для ядра Linux і HTTP-сервера Apache. В [56] додати токен, який генерується з вихідного коду в дослідженні, щоб визначити вразливі компоненти. Такж порівняно прогностичні повноваження між метриками програмного забезпечення та майнінгом тексту в прогнозуванні вразливих компонентів та вивчено вивчити вразливості, які важко відтворюються (HRV) на рівні коду, і досягають точності 82% та відкликання 84% для класифікації вразливих файлів у файли, схильні до HRV або не схильні до HRV.

Mathematic VDM фокусується на моделюючому процесі виявлення вразливих програмних вразливостей шляхом оцінки кількості вразливостей з часом. Існуючі моделі є Lin-ear [57], Експоненціальна, Alhazmi Malaiya Logistic (AML) та модель, заснована на зусиллях. VDMs, як правило, математичні моделі з параметрами, в яких реальні історії вразливості необхідні для отримання моделі. Ці моделі є специфічними для програмних додатків через те, що реальні дані історії вразливості необхідні в створенні моделі. Як правило, великі дані історії необхідні для отримання краще пристосованої моделі.

На відміну від існуючих моделей, підходи , які використовуються в роботі збирають функції від показників five, щоб прогнозувати кількість вразливостей у

рівні додатків (а не вразливого компонента). На відміну від того факту, що математичні VDMs вимагають великої кількості історичних даних, моделі, розроблені під час роботи вивчають важливі особливості від метрик, щоб від залежності прогнозу від історичних даних.

1.6 Показники безпеки

В [58] змодельованоь мінімальні зусилля, необхідні зловмиснику для використання вразливостей, як показник для оцінки рівня безпеки систем. Та запропоновано мінімальні зусилля, необхідні для виконання кожного вторгнення мережного типу як метрику. Розглядаючи шлях атаки як показник безпеки, а також запропоновано метрику для вимірювання міцності безпеки мережі, переоцінюючи силу найслабшого зловмисника (який має найменші можливості скомпрометувати цю систему). За допомогою налаштованого алгоритму ця метрика обчислює мінімальні набори необхідних початкових атрибутів, щоб зловмисник з найменшими можливостями успішно скомпрометувати систему. Ці показники розроблені з точки зору зловмисників.

Використання найнижчих здібностей зловмисника як вимірювання для систем не може відображати середні здібності зловмисників. В [59] використано середній час на компроміс, який вивчає фізичну безпеку, як показник безпеки для вимірювання систем. Особа, яка приймає рішення, дізнається рішення щодо покращення, просто обчислюючи MMTC, що довше, тим безпечніше. Розглядаючи інвестиції та скорочення очікуваних збитків, подано кількісний показник управління ризиками, який використовує зменшення очікуваних втрат для вимірювання успіху інвестицій в інформаційну безпеку.

Більшість з наведених вище показників є відносно простими і розроблені на ранній стадії дослідження показників безпеки мережі. Прості метрики легко застосувати та зрозуміти, але можуть не підійти для складних і постійно мінливих мережевих системних середовищ.

В [60] визначено показник стійкості до атак для оцінки та порівняння безпеки різних конфігурацій мережі. Показник стійкості до атак вимірює здатність систем захищатися від атак, що може підказати адміністраторам потенційні кращі конфігурації мережевих систем, та запропоновано агрегувати ймовірності вразливості за Загальною системою оцінки вразливостей (CVSS). Імовірність уразливості вказує на успішне використання даної вразливості. Однак ця робота зосереджена лише на вразливостях, які ігнорують взаємозалежності всередині графа атаки. Щоб усунути це обмеження, надано модель на основі байєсівських мереж для агрегування вразливостей всередині мережевих систем. Байєсівські мережі визначаються вузлами, що представляють змінні, і дугами, що представляють умовну незалежність між змінними. Проте формальне поняття цієї метрики в цій роботі не визначено розширено від байєсівських мереж до динамічних байєсівських мереж, щоб впоратися з постійно мінливими мережевими середовищами. Хоча динамічна ситуація має деякі обмеження, чітке визначення в байєсівській мережі надихає подальшу роботу з показниками безпеки мережі.

В [61] запропоновано метрику, що обчислює мінімально необхідні ресурси для компрометації системи з метою зміцнення. Попередні методи посилення зазвичай розривають шляхи атаки, щоб посилити мережу, однак ця робота зосереджена на пошуку початкового необхідного набору для зловмисників. Таким чином, дії атаки можна було б легко зламати, відключивши початкові умови, які покращують це рішення, визначаючи чітке позначення мінімальної вартості зміцнення мережі на основі графів атак.

Метрики безпеки також розробляються для конкретних застосувань, таких як теоретико-інформаційні метрики для вимірювання ефективності IDS [62], де подано формальну структуру з використанням теорії інформації для аналізу та кількісної оцінки ефективності IDS. У цій роботі представлена формальна модель IDS, що аналізує інформаційно-теоретичний підхід.

Подібні роботи існують і в інших областях, наприклад, деякі з принципів проектування пропонуються для розробки метрик довіри. Автентифікаційні

сутності у великомасштабній системі завжди використовують автентифікацію, що означає, що кожен об’єкт може автентифікувати наступний шлях. Ця технологія була поширена на кілька шляхів. Райтер та ін. запропонували метрики для оцінки впевненості, яку надає набір шляхів.

1.7 Показник різноманітності мережі

Дослідження показників безпеки останнім часом привертають велику увагу. На відміну від існуючих робіт, які мають на меті виміряти рівень безпеки мережі [63], показники різноманітності мережі зосереджуються на різноманітності як на одній особливій властивості мереж, яка може впливати на безпеку. Тим не менш, робота запозичує популярну метрику безпеки програмного забезпечення, поверхню атаки, загальну ідею фокусування на інтерфейсах (віддалено доступних ресурсах), а не на внутрішніх деталях (наприклад, локальних додатках). Метрика різноманітності на основі найменшої атаки походить із метрики безпеки n-нульового дня, а імовірнісна метрика різноманітності заснована на метриці ймовірності атаки. Інша помітна робота оцінює показники безпеки проти реальних атак у контрольованому середовищі, що дає змогу краще оцінити роботу в майбутньому. Одне з обмежень роботи полягає у високій складності аналізу графа ресурсів; високорівневі моделі залежностей від ресурсів можуть надати більш грубі, але ефективніші рішення для моделювання різноманітності.

Ідея використання різноманітності дизайну для відмовостійкості досліджувалася протягом тривалого часу. Підхід до програмування K-версій генерує $K \geq 2$ функціонально еквівалентних програм і порівнює їх результати для визначення несправної версії з метриками, визначеними для вимірювання різноманітності програмного забезпечення та несправностей [64]. Основне обмеження різноманітності дизайну полягає у високій складності створення різних версій, що може не виправдовувати переваги. Використання різноманітності дизайну як механізму безпеки також привертає велику увагу. У [65] показано, що

загальні принципи різноманітності дизайну також застосовні до безпеки. Система K-версій розширює ідею програмування K-версії для виявлення вторгнень, а концепція поведінкової дистанції виносить її за межі вихідного голосування. Для автоматичного створення різноманітності використовувалися різні методи рандомізації.

На додаток до різноманітності дизайну та створеної різноманітності, останні роботи використовують опортуністичне різноманіття, яке вже існує серед різних програмних систем. Оцінено практичність використання різноманітності ОС для стійкості до вторгнень, а в [66] продемонстровано можливість використання опортуністичного розмаїття, яке вже існує між різними ОС, для дотримання вторгнень. Різноманітність також застосовувалася до систем стійкості до вторгнень, які зазвичай реалізують деякі види реплікації Візантійської Відмовостійкість (BFT) як відмовостійкі рішення. Загальна архітектура для реалізації стійких до вторгнень веб-серверів на основі принципів резервування та диверсифікації представлена в [67]. Розмаїття готових компонентів (COTS) використовується для забезпечення неявної еталонної моделі замість звичайної явної моделі для виявлення аномалій у веб-серверах. Різноманітність може відігравати важливу роль у вирішенні різних проблем безпеки в хмарних обчисленнях, наприклад, використання різноманітних повноважень для ефективного розшифрування та відкликання в хмарному сховищі та використання різноманітних політик доступу для підвищення безпеки хмарних даних.

1.8 Метрика поверхні мережевих атак

Концепція поверхні атаки спочатку була запропонована для конкретного програмного забезпечення та потребує спеціального досвіду для формулювання та реалізації. Згодом концепція узагальнюється за допомогою формальних моделей і стає застосовною до всього програмного забезпечення. Крім того, він уточнюється і застосовується до великомасштабного програмного забезпечення, і його

обчислення може сприяти автоматично згенерованим графам викликів [67]. Поверхня атаки протягом багатьох років привертала значну увагу. Він використовується як метрика для оцінки системи передачі повідомлень Android, у хвості ядра, а також служить основою для Moving Target Defense, яка в основному спрямована на зміну поверхні атаки з часом. Інші мають на меті розширити сферу дії цієї концепції в інших областях, таких як шестисторонні поверхні атаки між користувачами, службами та хмарними системами, а також наближення поверхні атаки для сучасних автомобілів. Дослідження щодо автоматизації обчислення поверхні атаки є ще однією цікавою областю, наприклад, COPES використовує статичний аналіз з байт-коду для обчислення поверхні атаки та для захисту програмного забезпечення на основі дозволів.

Трасування стеку зі звітів користувачів про збої використовується для автоматичного визначення поверхні атаки. Незважаючи на такий величезний інтерес до концепції поверхні атаки, наскільки відомо, існує мало робіт щодо формального визначення поверхні атаки на рівні мережі. Кореляція між поверхнею атаки та вразливими місцями також була досліджена, наприклад, використання точок входу на поверхню атаки та доступності для оцінки ризику вразливості.

Дослідження про взаємозв’язок між поверхнею атаки та щільністю вразливості наведено в [68], хоча результат базується лише на двох версіях HTTP-сервера Apache, що дає мало уявлень про загальне існування такої кореляції.

Що стосується показників безпеки загалом, здійснюють зусилля зі стандартизації оцінки вразливостей, включаючи загальну систему оцінки вразливостей (CVSS), яка вимірює вразливості окремо. З

усилля NIST щодо стандартизації показників безпеки також наведено в [69]. Дослідження показників безпеки останнім часом привертають велику увагу. Робота включала метрику з точки зору часу та зусиль на основі моделі Маркова]. Зовсім недавно було запропоновано кілька показників безпеки шляхом комбінування оцінок CVSS на основі графів атак. Мінімальні зусилля, необхідні для виконання кожного вторгнення мережного типу, використовуються як метрика в [70].

Запропоновано метрику середнього часу до компромісу на основі моделі простору стану хижака (SSM), що використовується в біологічних науках. Хоча ці показники здебільшого розроблені для відомих вразливостей, менша кількість робіт здатна впоратися з атаками нульового дня. Кілька винятків включають емпіричне дослідження загальної кількості вразливостей нульового дня, доступних за один день на основі наявних даних, спробу впорядкувати різні програми в системі за серйозністю наслідків наявності однієї уразливості нульового дня, а нещодавно модель безпеки k-нульового дня і модель мережевого різноманіття [65] намагаються моделювати ризик вразливостей нульового дня, але їх загальним обмеженням є відсутність можливості розрізняти різна ймовірність наявності таких вразливих місць. Метрика поверхні мережевої атаки розрізняє ресурси з імовірністю атаки від трьох вимірів у концепції поверхні атаки та об'єднує ймовірність атаки на рівні мережі для моделювання ризику вразливостей нульового дня, що усуває обмеження з попередніх робіт.

1.9 Модель виявлення вразливостей

У літературі досліджувалися дві основні моделі виявлення вразливостей (VDM); один зосереджується на вивченні функцій, які корелюють з уразливими компонентами в програмному додатку; інший зосереджений на використанні математичних моделей, щоб узгодити модель виявлення вразливостей з історичними даними для прогнозування майбутньої кількості вразливостей для однієї програми.

В [71] аналізують можливість прогнозування вразливих компонентів у ОС Windows за допомогою логістичної регресії для п'яти груп показників, відтоку, складності, покриття, залежності та організаційної структури компанії. Бінарні результати були оцінені за допомогою десятикратної перехресної перевірки, яка дає точність нижче 67% і відкликання нижче 21%, а також вивчають показники активності розробників і вразливості програмного забезпечення. Точність і

відкликання з прогнозної моделі мережі Байєса становить від 12%-29% і від 32% до 56% відповідно. В [72] вивчають взаємозв'язки між програмними метриками та вразливими компонентами в 14 веб-додатках з відкритим кодом. Рангова кореляція Спірмена обчислюється між метрикою та індикатором ресурсів безпеки (SRI), який визначений автором і отриманий зі сканерів безпеки. Також виконують бінарну класифікацію за допомогою логістичної регресії з десятикратною перехресною перевіркою, щоб проаналізувати взаємозв’язок між показниками складності, відтоку коду та активності розробника (CCD) та вразливими місцями. Точність цього дослідження становлять 83% і 11% відповідно.

В [73] аналізують вплив метаданих у сховищах коду за допомогою метрик коду, щоб передбачити уразливі коміти. Точність VCCFinder становить 60%, коли відкликання становить 24% та вивчають зв'язок між показниками програмного забезпечення та вразливою функцією в існуючих атаках мережного типу. Всього в їхньому дослідженні було досліджено 183 вразливості з національної бази даних уразливостей для ядра Linux і HTTP-сервера Apache. В [74] додали маркер, згенерований із вихідного коду, у дослідження, щоб ідентифікувати вразливі компоненти, порівнюють можливості прогнозування між програмними метриками та інтелектуальним аналізом тексту при прогнозуванні вразливих компонентів. Також вивчають важко відтворювані вразливості (HRV) на рівні коду, і досягають точності 82% і відкликання 84% для класифікації вразливих файлів на файли, схильні до HRV або не схильні до HRV.

Математичний VDM зосереджується на моделюванні процесу виявлення вразливостей програмного забезпечення шляхом оцінки кількості вразливостей з часом. Існуючими моделями є лінійна [75], експоненціальна, модель Alhazmi Malaiya Logistic (AML) та модель на основі зусиль. VDM зазвичай є математичними моделями з параметрами, в яких для отримання моделі потрібні реальні історії вразливостей. Ці моделі є специфічними для програмних додатків у зв’язку з тим, що для створення моделі необхідні дані реальної історії вразливостей. Зазвичай для отримання кращої моделі потрібні великі дані історії [76-82].

На відміну від існуючих моделей, підходи , які використовуються під час роботи збирають функції з п’яти різних показників для прогнозування кількості вразливостей на рівні програми (не уразливий компонент) [83-85]. Незважаючи на те, що математичні VDM вимагають великої кількості історичних даних, моделі, які використовуються в роботі вивчають важливі особливості метрики, щоб залежати від прогнозів від історичних даних [86-88].

1.10 Висновки та постановка задачі

В даному розділі розглянуто поняття резильєнтного функціонування та оцінка резильєнтності іт-інфраструктур в умовах здійснення атак мережного типу.

Також в розділі було досліджено відомі методи оцінки резильєнтності мережі в умовах атак.

Проаналізовано та досліджено показники резильєнтності мережі в умовах здійснення атак, зокрема показник різноманітності мережі, метрику покриття атаки мережі. В розділі досліджено моделі вразливостей, показники безпеки, показник різноманітності мережі, метрику поверхні мережевих атак, а також модель виявлення вразливостей.

В розділі зроблено висновки щодо необхідності розроблення удосконалений метод побудови резильєнтних ІТ-інфраструктур в умовах здійснення атак мережного типу.

2 МОДЕЛЮВАННЯ ПОКАЗНИКІВ БЕЗПЕКИ ДЛЯ ОЦІНКИ РЕЗИЛЬЄНТНОСТІ ІТ-ІНФРАСТРУКТУР В УМОВАХ ЗДІЙСНЕННЯ АТАК МЕРЕЖНОГО ТИПУ

2.1 Формальне моделювання різноманітності мережі

Однак на вищому рівні абстракції, як глобальній властивості всієї мережі, концепції різноманітності мережі та її впливу на безпеку приділяється обмежена увага. У цій темі робимо перший крок до формального моделювання різноманітності мережі як показника безпеки з метою оцінки стійкості мереж щодо атак нульового дня.

Описуємо кілька випадків використання, щоб мотивувати дослідження та проілюструвати різні вимоги та проблеми в моделюванні різноманітності мереж.

2.2 Особливості застосування моделі

Описуємо кілька випадків використання, щоб мотивувати дослідження та проілюструвати різні вимоги та проблеми в моделюванні різноманітності мереж.

Випадок використання 1: Stuxnet та SCADA Security. Stuxnet є одним із перших шкідливих програм, які використовують декілька (чотири) різні атаки нульового дня [76].

Це вказує на те, що в критично важливих системах, таких як наглядовий контроль та збір даних (SCADA), у цьому випадку, ризик атак нульового дня та численних невідомих вразливостей є реальним, і, отже, адміністраторам мережі знадобиться систематичний спосіб оцінки такого ризику.

Однак це складне завдання через відсутність попередніх знань про вразливості або методи атаки.

Більш уважний погляд на стратегії атаки Stuxnet покаже, як різноманітність мережі може допомогти тут. Stuxnet націлений на програмовані логічні контролери

(ПЛК) в системах управління газопроводами або електростанціями [29], які переважно програмуються за допомогою машин Windows, не підключених до мережі.

Stuxnet використовує багатоетапний підхід, спочатку заражаючи комп'ютери Windows, що належать третім сторонам (наприклад, підрядникам), потім поширюючись на внутрішні КС Windows через локальну мережу, і, нарешті, охоплюючи останній крок за допомогою знімних флеш накопичувачів.

Очевидно, що ступінь різноманітності програмного забезпечення вздовж потенційних шляхів атаки, що ведуть від периметра мережі до ПЛК, можна розглядати як критичну метрику стійкості мережі до загроз, подібних до Stuxnet.

Мета розділу полягає в тому, щоб забезпечити ретельне вивчення таких показників різноманітності мережі.

Варіант використання 2: Розповсюдження атак мережного типу. Щоб зробити обговорення більш конкретним, звернемося до поточного прикладу, показаного на рисунку 2.1.

Розглядаючи випадок, головне, що турбує це можливе поширення атак мережного типу або ботів всередині мережі.

Тут поширена думка, що можливо просто підрахувати кількість (відсоток) окремих ресурсів у мережі як різноманітність.

Наприклад, припустимо, що хости 1, 2 і 3 є веб-серверами, на яких запущено IIS, і всі мають доступ до файлів, збережених на хості 4.

Очевидно, що наведена вище метрика на основі кількості вказує на відсутність різноманітності та пропонує замінити IIS іншим програмним забезпеченням, щоб запобігти атака мережного типу від зараження всіх трьох одночасно.

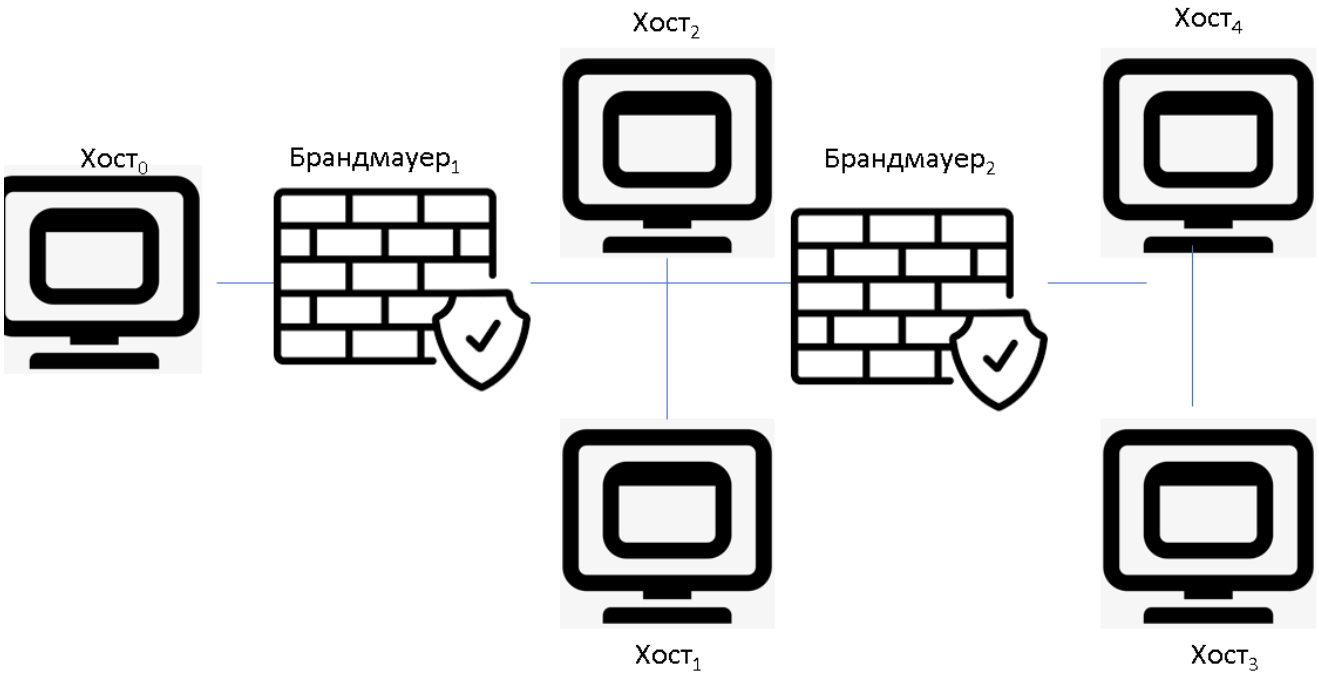


Рисунок 2.1 – Схема мережі

Однак легко помітити, що, навіть якщо атака мережного типу може заразити лише один веб-сервер після такої диверсифікації (наприклад, він може заражати IIS, але не Apache), він все одно може поширюватися на всі чотири хости через спільну мережу на хості 4 (наприклад, він може заражати певні виконувані файли, що зберігаються на хості 4, до яких згодом звертаються всі веб-сервери).

Причина того, що цей наївний підхід зазнає невдачі в цьому випадку, полягає в тому, що він ігнорує існування причинно-наслідкових зв’язків між ресурсами (через спільну мережу).

Випадок використання 3: Цілеспрямована атака.

Припустимо, що нас більше турбує цілеспрямована атака на сервер зберігання даних, хост 4.

Після обговорення вище, інтуїтивним рішенням є диверсифікація ресурсів на будь-якому шляху, що веде до критичного активу (хост 4), наприклад, між хостами 1 (або 2). , 3) і хост 4.

Хоча це достовірне спостереження, усвідомлення цього вимагає ретельного

вивчення причинно-наслідкових зв’язків між різними ресурсами, оскільки хост 4 настільки безпечний, наскільки безпечний найслабший шлях (що представляє найменшу атаку), що веде до нього.

Запропонуємо формальну метрику, засновану на такій інтуїції, у розділі 3.4.

З іншого боку, найменша атака сама по собі дає лише часткову картину. Припустимо, що тепер хости 1 і 2 диверсифіковані для запуску IIS і Apache відповідно, а брандмауер 2 дозволить лише хостам 1 і 2 досягти хоста 4.

Хоча найменші зусилля атаки не змінилися, ці зусилля з диверсифікації фактично надали зловмисникам більше можливостей для досягти хоста 4 (використовуючи IIS або Apache).

Тобто неправильне розмаїття насправді може зашкодити безпеці. Це буде відображено імовірнісною метрикою в Розділі 3.5.

Випадок використання 4: MTD. Moving Target Defense (MTD) можна розглядати як інший підхід до застосування різноманітності до безпеки, оскільки вона диверсифікує ресурси за часовим виміром [49].

Однак більшість існуючих робіт з MTD спираються на інтуїтивне уявлення про різноманітність, що може призвести до оманливих результатів.

Наступний випадок демонструє корисність запропонованих показників, зокрема для MTD.

Розглянемо випадок і припустимо, що хости 1 і 2 є веб-серверами, хост 3 — сервером додатків, а хост 4 — сервером баз даних. MTD намагатиметься досягти кращої безпеки, змінюючи в часі програмні компоненти на різних рівнях.

Поширеною помилкою тут є те, що поєднання різних компонентів на різних рівнях збільшить різноманітність, а ступінь різноманітності дорівнює добутку різноманітності на цих рівнях.

Однак зазвичай це не так. Наприклад, одна помилка на сервері додатків (хост 3) може призвести до ін’єкції SQL, яка скомпрометує сервер бази даних (хост 4) і, як наслідок, витоку пароля користувача.

Крім того, як і в попередньому випадку, збільшення різноманітності з часом

може фактично надати зловмисникам більше можливостей знайти недоліки.

2.3 Показник мережевого різноманіття

Хоча поняття мережевого різноманіття привертає обмежену увагу, його аналог в екології, біорізноманітті та його позитивний вплив на стабільність екосистеми досліджували протягом багатьох десятиліть [77].

Багато випадків потенційно можуть бути запозичені з багатої літератури про біорізноманіття, розглядаючи розділ, зосередимося на адаптації існуючих математичних моделей біорізноманіття для моделювання мережевого різноманіття.

Зокрема, кількість різних видів в екосистемі відома як видове багатство [78]. Аналогічно, враховуючи набір різних типів ресурсів P (розглянемо подібність ресурсів пізніше) у мережі, називаємо потужність | P | багатство ресурсів у мережі. Очевидним обмеженням цього показника багатства є те, що він ігнорує відносну кількість кожного типу ресурсу.

Наприклад, два набори $\{p_1, p_1, p_2, p_2\}$ і $\{p_1, p_2, p_2, p_2\}$ мають однакове багатство 2, але чітко різні рівні різноманітності.

Щоб усунути це обмеження, індекс Шеннона-Вінера, який, по суті, є ентропією Шеннона з використанням натурального логарифма, використовується як індекс різноманітності для групування всіх систем з однаковим рівнем різноманітності.

Експонента індексу різноманітності розглядається як ефективна числова метрика.

Ефективне число в основному дозволяє завжди вимірювати різноманітність з точки зору кількості однаково поширених видів, навіть якщо насправді ці види можуть бути не однаково поширеними.

У наступному рівнянні запозичено це поняття, щоб визначити ефективне багатство ресурсів і перший показник різноманітності.

Ефективна d_1-різноманітність у мережі H з набором хостів $G = \{g_1,g_2,...,g_n\}$,

набором типів ресурсів P = $\{p_1,p_2,...,p_m\}$ і відображенням ресурсу res(.): G → 2^P (тут 2^P позначає набір ступенів P), нехай t = $\sum_{i=1}^{k}|res(g_1)|$ (загальна кількість екземплярів ресурсу), і нехай $r_j = \frac{|\{g_i:r_j \in res(g_i)\}|}{t}|$ ($1 \le i \le n, 1 \le j \le m$) (відносна частота кожного ресурсу).

Визначаємо різноманітність мережі як $d_1 = \frac{p(H)}{t}$, де p(H) – це ефективне багатство ресурсів мережі, визначене як

$$p(H) = \frac{1}{\prod_1^k r_i^{r_i}}. \quad (2.1)$$

Одним з обмежень ефективної метрики на основі чисел є те, що подібність між різними типами ресурсів не враховується, а всі типи ресурсів вважаються абсолютно різними, що нереалістично (наприклад, одну і ту ж програму можна налаштувати на виконання абсолютно різних ролей , наприклад Nginx як зворотний проксі-сервер або веб-сервер відповідно, і в цьому випадку їх слід розглядати як різні ресурси з великою схожістю).

Тому, запозичуємо чутливу до подібності метрику біорізноманіття, нещодавно введену в [62], щоб повторно визначити багатство ресурсів.

З цим новим визначенням вищенаведена метрика різноманітності d_1 тепер може обробляти схожість між ресурсами.

Припустимо, що функція подібності задана як v(.) : [1,m]×[1,m] → [0,1] (більше значення означає більшу подібність і v(i,i) = 1 для всі $1 \le i \le m$), нехай $vr_i = \sum_{j=1}^{m} v(i,j)r_j$. Визначаємо ефективне багатство ресурсів мережі, враховуючи функцію подібності, як

$$p(H) = \frac{1}{\prod_1^k vr_i^{r_i}}. \quad (2.2)$$

Ефективний показник різноманітності мережі d_1 на основі багатства

підходить лише для тих випадків, коли всі ресурси можуть розглядатися однаково, а причинно-наслідкові зв’язки між ресурсами або не існують, або їх можна безпечно ігнорувати.

З іншого боку, ця метрика також може використовуватися як будівельний блок всередині інших показників різноманітності мережі, у тому сенсі, що можливо просто сказати «кількість окремих ресурсів», не турбуючись про нерівномірний розподіл типів ресурсів або схожість між ресурсами, завдяки до концепцій ефективного багатства, наведених у визначенні 1 і 2.

Було показано, що вплив біорізноманіття на стабільність екосистеми критично залежить від взаємодії різних видів всередині харчової мережі [69].

Хоча така взаємодія, як правило, має форму взаємозв’язку «живлення» між різними видами, що не стосується безпосередньо комп’ютерних мереж, це спостереження надихнуло на моделювання різноманітності на основі структурних відносин між ресурсами, які будуть детально описані в найближчі розділи.

2.4 Показник різноманітності мережі на основі найменших зусиль

Розглянемо, як моделюється різноманітність мережі на основі найменших зусиль атаки. Евристичний алгоритм пошуку d_2 описаний у [55].

2.4.1 Модель різноманітність на основі найменших зусиль для атаки

Щоб моделювати різноманітність на основі найменших зусиль для атаки, розглядаючи причинно-наслідкові зв’язки між різними ресурсами, спочатку потрібна модель таких зв’язків і можливих атак нульового дня.

Модель, яка використовується в роботі подібна до моделі графа атаки [6], хоча модель зосереджена на віддалено доступних ресурсах (наприклад, сервісах або додатках, які доступні з інших хостів у мережі), які будуть розглядатися як заповнювачі для потенційного нульового дня вразливості замість відомих

вразливостей, як у графах атак.

Щоб побудувати інтуїцію, переглянемо рисунок 1, зробивши наступні припущення. Доступ із зовнішнього брандмауера 1 дозволений до хосту 1, але заблокований до хосту 2; доступи з хоста 1 або 2 дозволені до хосту 3, але блокуються до хоста 4 брандмауером 2; хости 1 і 2 надають послугу http; хост 3 надає службу ssh; Хост 4 надає послуги як http, так і rsh. сервіси.

На рисунку 2 зображено відповідний граф ресурсів, який синтаксично еквівалентний графу атак, але моделює атаки нульового дня, а не відомі вразливості.

Кожна пара в відкритому тексті є умовою безпеки (наприклад, підключення <джерело, призначення> або привілей <привілей, хост>), і кожна трійка всередині поля є потенційним використанням ресурсу <ресурсу, вихідного хосту, місця призначення >; край вказують від попередніх умов до вторгнення мережного типу нульового дня (наприклад, від <0,1> і <user,0> до <http,0,1>), а від цього вторгнення мережного типу до його пост-умов (наприклад, від <http,0,1> до <user,1>).

Атаки мережного типу або умови, пов’язані з брандмауером 2, пропущені для простоти.

Просто розглядаємо ресурси різних типів як абсолютно різні (їх подібність можна обробити, використовуючи ефективне багатство ресурсів, наведене у Визначенні 2).

Крім того, використовується консервативний підхід, розглядаючи всі ресурси (сервіси та брандмауери) як потенційно вразливі до атак нульового дня.

Граф ресурсів для мережі з набором хостів G, набір ресурсів P із відображенням ресурсів res(.) : $G \rightarrow 2^P$, набір атак мережного типу нульового дня $E = \{<p, g_s, g_d> \mid g_s \in G, g_d \in G, p \in res(g_d)\}$ та їх попередні та постумови C, граф ресурсів – це орієнтований граф $H(E \cup C, P_p \cup P_i)$, де $P_p \subseteq C \times E$ і $P_i \subseteq E \times C$ – відношення до та після умов відповідно.

Далі розглянемо, як зловмисники можуть потенційно атакувати критичний мережевий актив, змодельований як умова цілі, з найменшими зусиллями.

На рисунку 2.2, дотримуючись простого правила, згідно з яким атака мережного типу може бути виконано, якщо всі передумови задоволені, і виконання цього вторгнення мережного типу призведе до виконання всіх постумов, можемо спостерігати шість шляхів атаки, як показано в таблиці 1 (другий і третій стовпці наразі можна проігнорувати, і будуть пояснені незабаром).

Таблиця 2.1 – Шлях атаки

Шлях атаки	Кількість кроків# кроків	Кількість ресурсів
1. (http,0,1) —> (ssh, 1,4) —> (rsh.4.5)	3	3
2. {http A), 1) —> (ssh, 1,4) —> {http, 4,5)	3	2
3. (http, 0,1) —> { http , 1,2) —> (ssh ,2,4) —> (rsh,4,5)	4	3
4. (http, 0,1) —> { http , 1,2) —> (ssh ,2,4) —> (http,4,5)	4	2
5. (firewall,Q,F) —> (http, 0,2) —> (ssh, 2,4) —> (rsh,4,5)	4	4
6. (firewall.0.F) —> (http,0,2) —> { ssh,2,4) —> (http,4,5)	4	3

Шлях атаки для графа ресурсів $H(E \cup C, P_p \cup P_i)$ називаємо $C_1 = \{c : c \in C,(\nexists e \in E)(<e, c> \in P_i)\}$ множиною початкових умов.

Будь-яка послідовність операцій нульового дня e_1, e_2,..., e_k називається шляхом атаки в H, якщо $(\forall i \in [1,k])(<c, e_i> \in P_p \rightarrow (c \in C_i \vee (\exists\, j \in [1,i- 1])(<e_j , c> \in P_i)))$, і для будь-якого $c \in C$ використовуємо seq(c) для набору шляхів атаки $\{e_1, e_2,..., e_k : <e_k , c> \in P_i\}$.

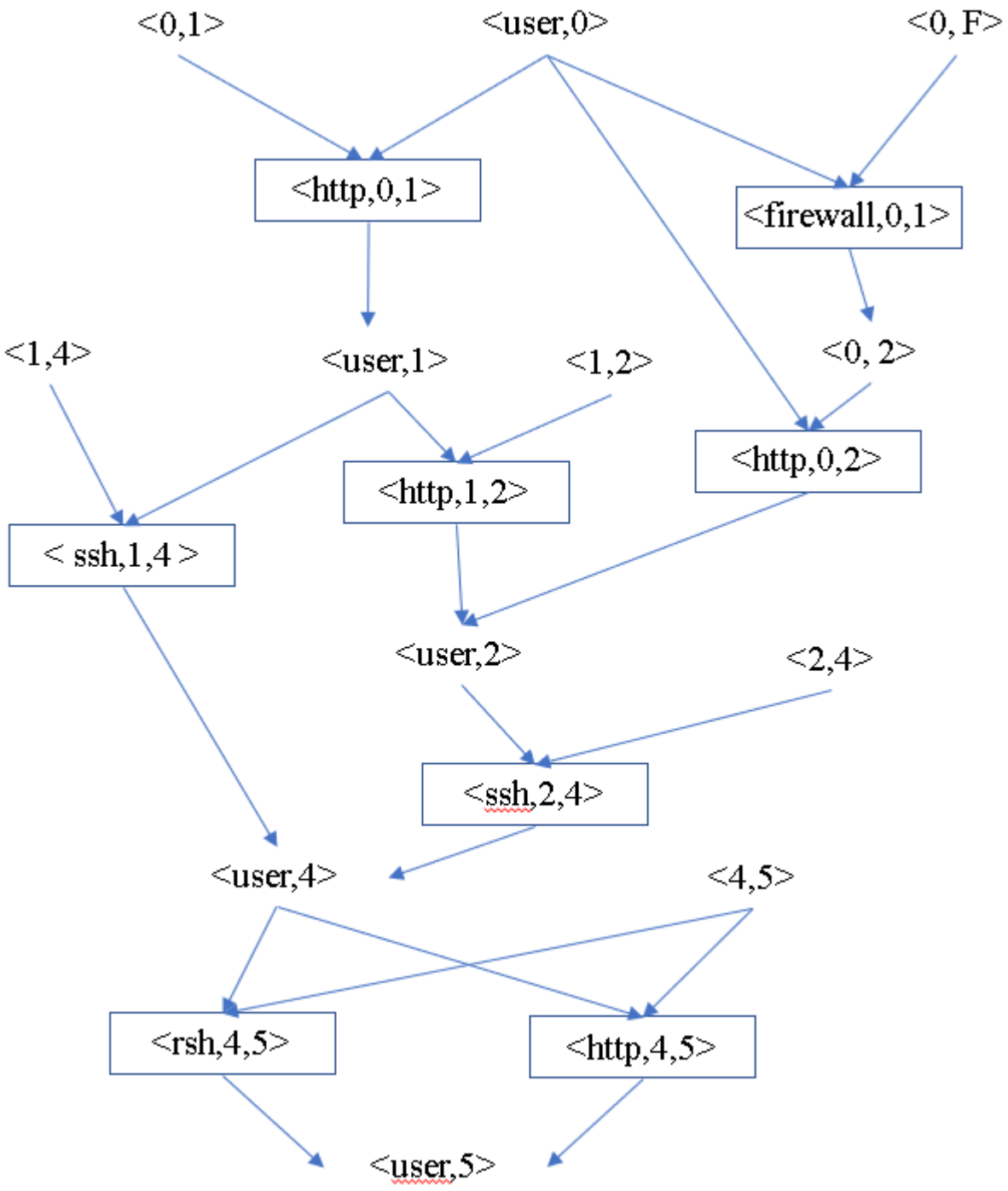


Рисунок 2.2 – Приклад графа ресурсів

Тепер розглядаємо, як можна визначити різноманітність на основі найменшого нападу (найкоротший шлях).

Насправді існує кілька можливих способів вибору таких найкоротших шляхів і визначення метрики, як проілюстровано на прикладі нижче.

По-перше, як показано в другому стовпці таблиці 1, шляхи 1 і 2 є найкоротшими з точки зору кроків (тобто, кількість шкідливих атак мережного типу нульового дня).

Очевидно, що шляхи не відображають найменших зусиль для атаки, оскільки шлях 4 насправді може зайняти менше зусиль, ніж шлях 1, оскільки зловмисники можуть повторно використовувати свій код вторгнення мережного типу, інструменти та навички, використовуючи ту саму службу http на трьох різних хостах.

Далі, як показано в третьому стовпці, шляхи 2 і 4 є найкоротшими з точки зору кількості окремих ресурсів (або ефективного багатства).

Це здається більш оптимальним, оскільки фіксує заощаджені зусилля при повторному використанні атак мережного типу.

Однак, хоча шляхи 2 і 4 мають однакову кількість різних ресурсів (2), вони чітко відображають різну різноманітність.

Іншим, здавалося б, прийнятним рішенням є заснування на мінімальному відношенні $\frac{\text{\# ресурсів}}{\text{\# кроків}}$ (яке наведено шляхом 4 у прикладі), оскільки таке співвідношення відображає потенційні покращення з точки зору різноманітності (наприклад, співвідношення $\frac{2}{4}$ шлях 4 вказує на 50% потенційного покращення різноманітності).

Однак можемо легко уявити, що дуже довгий шлях атаки мінімізує такий коефіцієнт, але не відображає найменші зусилля атаки (наприклад, шлях атаки з 9 кроками і 3 різними ресурсами дасть відношення $\frac{1}{3}$, менше $\frac{2}{4}$, але очевидно вимагає більше зусиль, ніж шлях 4).

Врешті, ще один варіант — вибрати найкоротший шлях, який мінімізує як кількість окремих ресурсів (шлях 2 і 4), так і наведене вище співвідношення $\frac{\text{\# ресурсів}}{\text{\# кроків}}$ (шлях 4).

Однак при ближчому розгляді буде виявлено, що, хоча шлях 4 і представляє найменші зусилля для атаки, він не являє собою максимальну кількість потенційного покращення різноманітності, тому що як тільки починаємо диверсифікувати шлях 4, найкоротший шлях може змінитися на шлях 1 або 2.

На основі цих обговорень визначаємо мережеве різноманіття, поєднуючи перші два варіанти вище.

Зокрема, різноманітність мережі визначається як співвідношення між мінімальною кількістю окремих ресурсів на шляху та мінімальною кількістю кроків на шляху (зверніть увагу, що це можуть бути різні шляхи).

Повертаючись до прикладу вище, бачимо, що шляхи 2 і 4 мають мінімальну кількість окремих ресурсів (два), а також шляхи 1 і 2 мають мінімальну кількість кроків (три), тому різноманітність мережі в прикладі дорівнює $\frac{2}{3}$ (зверніть увагу, що це відношення ніколи не перевищить 1). Інтуїтивно, чисельник 2 позначає поточний рівень надійності мережі проти атак мережного типу нульового дня (не більше 2 різних атак), тоді як знаменник 3 позначає максимальний потенціал надійності мережі (витримує не більше 3 різних атак) за рахунок збільшення кількості різноманітність (від $\frac{2}{3}$ до 1).

Більш формально представляємо другий показник різноманітності мережі у визначенні 5 (зауважте, що для простоти розглядаємо лише одну ціль для представлення даного критичного активу, що не є обмеженням, оскільки кілька умов цілі можна легко обробити шляхом додавання кілька розглядуваних фіктивних умов [3]).

d_2-Різноманітність враховуючи граф ресурсів H(E ∪C,P_p∪P_i) та умову мети c_h∈C, для кожного c∈C і q∈seq(c) позначимо P(q) для {p:p ∈ P,p. з'являється у q} різноманіття мережі визначається як (де min(.) повертає мінімальне значення в наборі)

$$d_2 = \frac{min_{q \in seq(c_h)} |P(q)|}{min_{q' \in seq(c_h)} |\acute{q}|}. \quad (2.3)$$

2.5 Імовірнісна різноманітність мережі

Розглянемо імовірнісний показник для визначення ефекту різноманітності на основі середніх зусиль атаки шляхом поєднання всіх шляхів атаки.

Спочатку визначимо важливі обмеження в цій моделі, а потім розглянемо оновлену модель для їх усунення.

2.5.1 Імовірнісна модель мережевого різноманіття

Розглянемо імовірнісну модель мережевого різноманіття, яка визначає різноманітність мережі як співвідношення між двома ймовірностями, а саме, ймовірністю того, що певні критичні активи можуть бути скомпрометовані, і такою ж імовірністю, але з додатковим припущенням, що всі екземпляри ресурсу є різними (що означає, що зловмисники не можуть повторно використовувати жодний атака мережного типу).

Обидві ймовірності представляють ймовірність атаки щодо умов цілі, яку можна змоделювати за допомогою байєсівської мережі, побудованої на основі графа ресурсів [32].

Наприклад, рисунок 2.3 демонструє цю модель на основі запущеного прикладу (для простоти показана лише частина прикладу).

Ліва сторона представляє випадок, коли ефект повторного використання вторгнення мережного типу не враховується, тобто два екземпляри служби http вважаються різними.

Права сторона розглядає цей ефект і моделює його як умовну ймовірність того, що нижня служба http може бути використана, враховуючи, що верхня вже експлуатується (позначена пунктирною лінією).

Дві таблиці умовних ймовірностей (CPT) ілюструють ефект повторного використання вторгнення мережного типу http (наприклад, ймовірність 0,9 у

правому CPT) і його невикористання (наприклад, ймовірність 0,08 у лівому CPT), відповідно. Рознесення мережі в розглянутому випадку буде розраховано як відношення $d_3 = \frac{0.0064}{0.072}$.

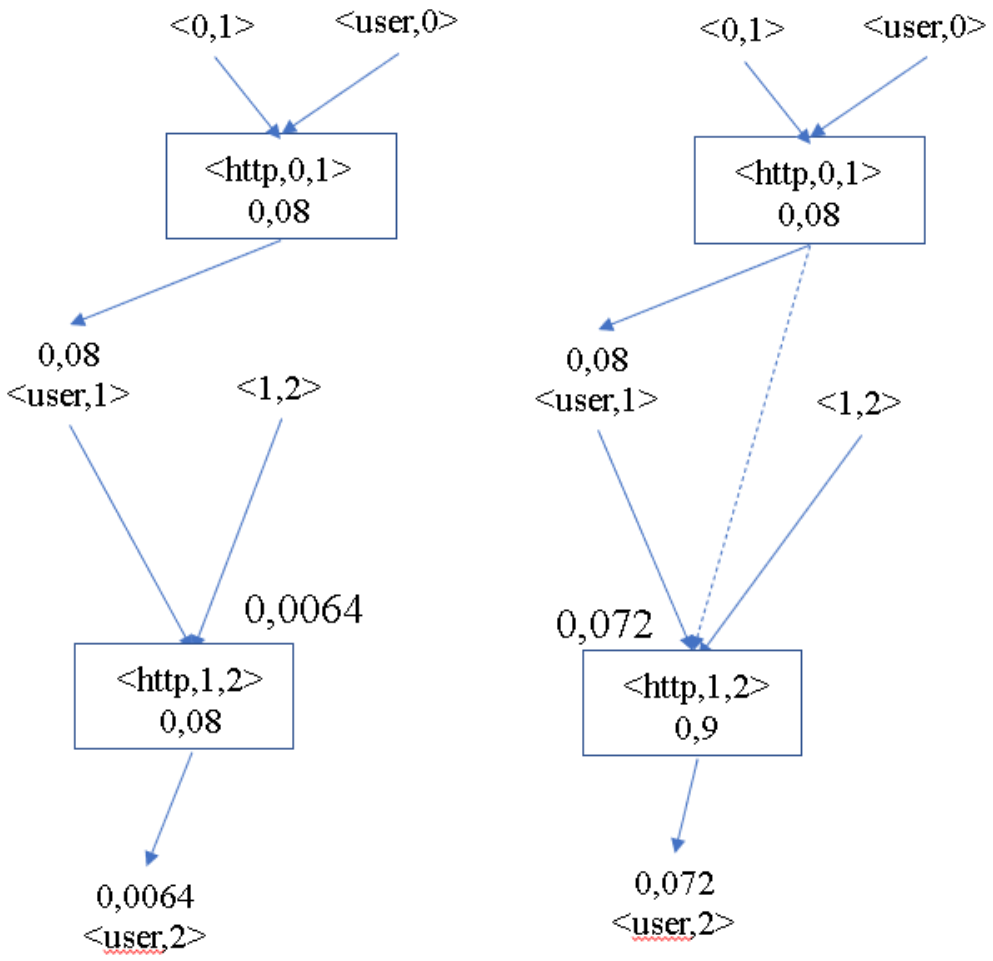


Рисунок 2.3 – Моделювання різноманітності мережі за допомогою байєсівських мереж

Зрозуміло, що наведена модель має певні обмеження, коли під час моделювання було повернуто кілька недійсних результатів (більше 1).

Точніше, у наведеній вище моделі моделювання ефекту повторного використання атак мережного типу як умовної ймовірності (те, що ресурс може бути використаний, враховуючи, що деякі інші екземпляри того ж типу вже

експлуатуються) по суті, передбачає повний порядок для різних екземплярів того самого ресурсу. введіть будь-який граф ресурсів, що містить значні обмеження.

Наприклад, на рисунку 2.4 (пунктир і поле, а також таблицю CPT на даний момент можна ігнорувати), хоча повторно використаний http атака мережного типу <http,1,2> (після використання <http,0,1>) може оброблятися за допомогою наведеної вище моделі шляхом додавання пунктирної лінії, що вказує на неї від її предка <http,0,1>, той самий метод не працюватиме для іншого потенційно повторно використаного http вторгнення мережного типу <http,0,2>, оскільки не існує певного порядку між <http,0,1> і <http,0,2>, що означає, що зловмисник може досягти <http,0,2> до або після досягнення <http,0,1>.

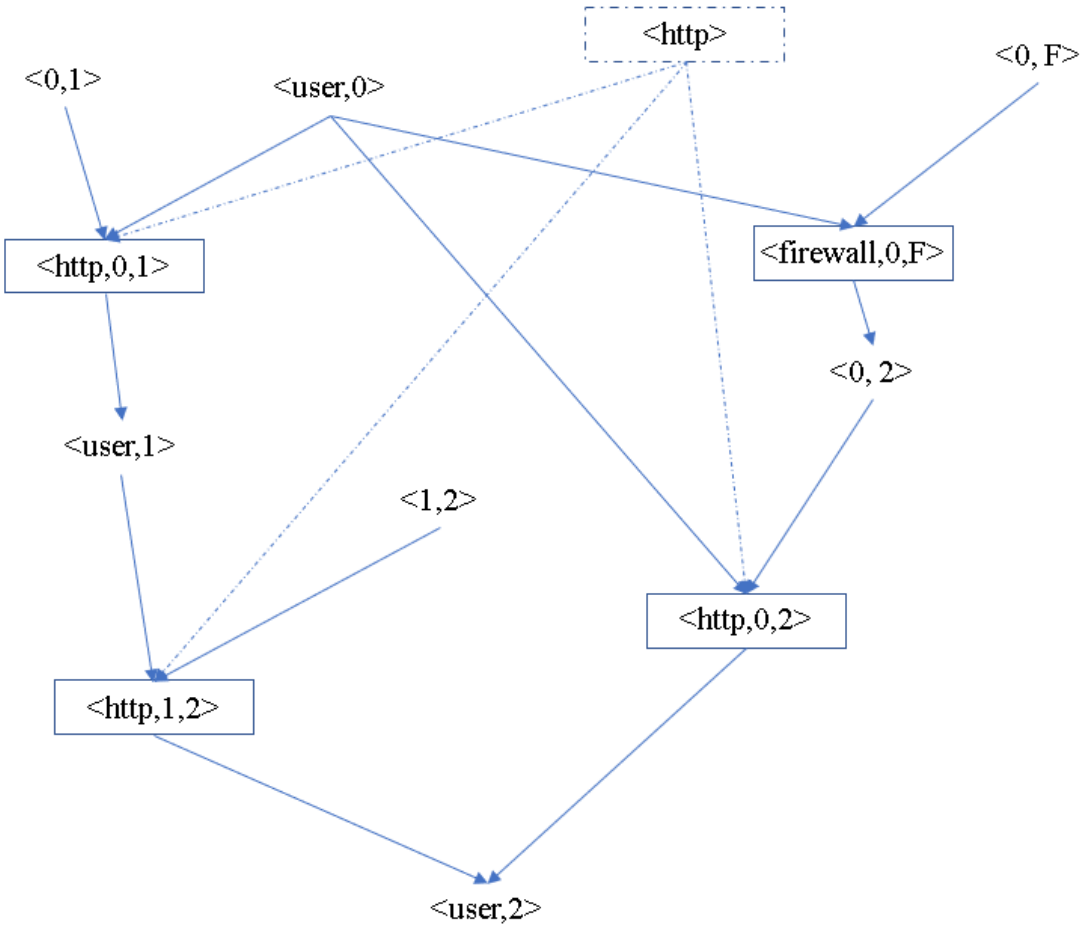


Рисунок 2.4 – удосконалена модель

Тому не можливо легко припустити, що один із них буде використаний

першим.

Враховуючи, що модель графа ресурсів визначається на основі байєсівської мережі, яка за визначенням вимагає ациклічних графів, також не можливо додати двонаправлені пунктирні лінії між атаками мережного типу.

Іншим пов'язаним обмеженням є те, що, якщо атаки мережного типу вважаються частково замовленими, ймовірність атаки не обов'язково буде найнижчою, якщо всі ресурси вважаються різними. Наприклад, на рисунку 4 зловмисник може досягти умови <user,2> двома шляхами: <http,0,1> → <http,1,2> і <firewall,0,F> → <http,0, 2>. Інтуїтивно, ймовірність атаки буде вищою, якщо вважатиметься, що http-атаки мережного типу в двох шляхах є різними, оскільки тепер у зловмисника буде більше вибору для досягнення цільової умови <user,2>.

Ці обмеження будуть розглянуті в наступних розділах за допомогою оновленої моделі.

2.5.2 Перебудова метрики d_3

Щоб усунути вищезгадані обмеження вихідної метрики d_3, можливим шляхом є перероблення моделі повторного використання атак мережного типу того самого типу ресурсу. Інтуїтивно, те, що дозволяє зловмиснику з більшою ймовірністю досягти успіху в експлуатації раніше використаного типу ресурсів, — це знання, навички або поведінка атаки, які він/вона отримав.

Тому, замість безпосереднього моделювання випадкових зв'язків між повторно використаними атаками мережного типу, явно моделюємо такі переваги зловмисника як окремі події та моделюємо їх ефект збільшення ймовірності успіху в наступних атаках мережного типу як умовні ймовірності.

Точніше, новий батьківський вузол, загальний для атак мережного типу того ж типу ресурсу, буде додано до графа ресурсів, як показано на рисунку 4 за допомогою пунктирних ліній і прямокутника.

Цей загальний батьківський вузол представляє подію, коли зловмисник має

можливість використовувати цей тип ресурсів. Однак, на відміну від вузлів, що представляють початкові умови, які будуть розглядатися як докази для розрахунку апостеріорної ймовірності умови мети, подія, коли зловмисник може використати тип ресурсів, не буде вважатися спостережуваним.

Додавання спільного батьківського вузла до атак мережного типу того ж типу ресурсу призведе до імовірнісної залежності між дочірніми вузлами, так що задоволення одного дочірнього вузла збільшить ймовірність інших, що моделює ефект повторного використання атак мережного типу.

Наприклад, на рисунку 4 пунктирна лінія вказує на новий вузол <http>, що представляє подію, коли зловмисник має можливість використовувати http-ресурси. Пунктирні лінії відображають умовні ймовірності того, що зловмисник може використати кожен екземпляр http, а таблиця CPT показує приклад такої умовної ймовірності для <http,1,2>.

Гранична ймовірність 0,08, призначена для <http>, представляє ймовірність того, що зловмисник може використовувати ресурси http, а умовна ймовірність 0,9, призначена для <http,1,2>, представляє ймовірність того, що той самий зловмисник використає цей конкретний екземпляр.

Існування такого спільного батьківського елемента введе залежність між цими http атаками мережного типу, так що задоволення одного збільшує ймовірність інших.

У визначенні другий набір умовних ймовірностей представляє ймовірність того, що зловмисник здатний використовувати кожен тип ресурсів.

Третій і четвертий набори разом представляють семантику графа ресурсів. Нарешті, п’ятий набір представляє умовну ймовірність того, що атака мережного типу може бути виконано, коли задовольняються його передумови (включаючи умову, яка представляє відповідний тип ресурсу).

d_3 Різноманітність для графу ресурсів $H(E \cup C, P_p \cup P_i)$, нехай $P' \subseteq P$ — набір типів ресурсів, кожен з яких спільний принаймні двома атаками мережного типу в E, і нехай $P_s = \{(p,<p,g_s,g_d>) : p \in P' <p ,g_s,g_d> \in E\}$ (тобто ребра від типів ресурсу до

екземплярів ресурсу).

Побудуємо байєсівську мережу B = (H' (E ∪C ∪P' 0 ,P_p ∪P_i ∪P_s),θ), де H' отримують шляхом введення P' і P_s в граф ресурсів H і розглядаючи кожен вузол як дискретний випадковий змінна з двома станами T і F, а θ — набір параметрів байєсівської мережі, заданий таким чином.

I. R(c = T) = 1 для всіх початкових умов c ∈ CI .

II. R(p = T) наведено для всіх спільних типів ресурсів p ∈ P' .

III. $R(e \mid \exists c_{<c,e>\in Pp} = F) = 0$ (тобто атака мережного типу не може бути виконаний, поки не будуть задоволені всі його попередні умови).

IV. $R(c \mid \exists e_{<c,e>\in Pi} = T) = 1$ (тобто постумова може бути задоволена лише будь-яким вторгненням мережного типуом).

V. $R(e \mid \forall c_{<c,e>\in PpUPs} = T)$ задані для всіх e ∈ E (тобто ймовірність успішного виконання вторгнення мережного типу, коли всі його передумови будуть задоволені).

Для будь-якого c_h ∈ C мережеве різноманіття d_3 визначається як $d_3 = \frac{r'}{r}$, де $r = R(c_h \mid \forall c_{c\in CI} = T)$ (тобто умовна ймовірність виконання c_h за умови, що всі початкові умови є true), а r' позначає мінімально можливе значення r, коли деякі ребра видаляються з P_s (тобто найнижча ймовірність атаки, якщо припустити, що певні типи ресурсів більше не використовуються атаками мережного типу).

На рисунку 2.5 показано два простих приклади, у яких перший зображує зв'язок між двома атаками мережного типу (у тому сенсі, що обидва верхні атаки мережного типу повинні бути виконані до того, як буде досягнуто нижнього вторгнення мережного типу), тоді як другий – відношення диз'юнкції (будь-який з двох верхніх атак мережного типу). подвиги самі по собі можуть призвести до нижчого подвигу).

В обох випадках, припускаючи $c_h = <c_3,1>$, ймовірність $r = R(c_h \mid \forall c_{c\in CI} = T)$ показана на рисунку. Тепер розглянемо, як обчислити нормалізуючу константу r' . Для лівого випадку ймовірність $r = R(c_h \mid \forall c_{c\in CI} = T)$ буде мінімізована, якщо видалимо обидва ребра з верхнього вузла (v_1) до двох його дочірніх (тобто ці два

атаки мережного типу не довше використовувати той самий тип ресурсу).

Можна розрахувати, що r' = 0,0064, а отже, різноманіття $d_3 = \frac{0,0064}{0,0648}$ у цьому випадку. Правий випадок є більш цікавим, оскільки виявляється, що p уже зведено до мінімуму, оскільки видалення ребер із верхнього вузла (v_1) призведе лише до більшого значення r (оскільки зловмисник матиме два різні шляхи для досягнення нижчого вторгнення мережного типу), яке можна обчислити як 0,1536.

Отже, різноманітність у цьому випадку дорівнює $d_3 = \frac{0,0792}{0,0792}$, тобто покращення різноманітності не підвищить (насправді це зашкодить) безпеці в цьому випадку.

Цей приклад також підтверджує попереднє спостереження, що припущення, що всі ресурси є різними, не обов'язково призводить до найменшої ймовірності атаки.

Наведений вище приклад також призводить до спостереження, що нормалізуюча константа r' не завжди може бути простою для обчислення, оскільки знаходження випадку, у якому p мінімізується, по суті означає, що потрібно оптимізувати різноманітність мережі для підвищення її безпеки, яка сама по собі є цікавим майбутнім. напрямок.

Замість цього пропонуємо наближений варіант нормалізації константи r' на основі наступних спостережень із наведеного вище прикладу.

На рисунку 2.5 бачимо, що права частина містить два шляхи атаки, що ведуть до умови цілі <c_3,1> (оскільки кожного з верхніх атак мережного типу достатньо, щоб привести до нижнього вторгнення мережного типу).

Раніше було показано, що видалення штрихових ліній лише збільшить ймовірність p (досягнення умови цілі).

Однак можемо легко побачити, що незалежно від того, видалимо пунктирні лінії чи ні, ймовірність p завжди була б мінімізована, якби був лише один шлях (наприклад, видаливши <v_1,2,1> з рисунку).

Зауважте, що для лівої сторони вже існує лише один шлях, оскільки обидва

верхні атаки мережного типу необхідні для досягнення нижнього вторгнення мережного типу, тому p мінімізується, коли два верхні атаки мережного типу вважаються різними.

Зрозуміло, що безпека мережі ніколи не може перевищувати випадок, коли на графі ресурсів залишається лише найкоротший шлях (з точки зору кількості кроків), а жоден ресурс не використовується на шляху.

Нормалізуюча константа r' у визначенні 6 завжди задовольняє $r' \geq r''$, де r'' — ймовірність $R(c_h \mid \forall c_{c \in CI} = T)$, розрахована на найкоротшому шляху атаки в термінах кроків (див. Розділ 3.4.1) .

Доведення (ескіз): Доводимо результат математичною індукцією на i, кількість кроків на найкоротшому шляху.

Базовий випадок $i = 1$ є тривіальним. Для індуктивного випадку припустимо, що результат справедливий для будь-якого графа ресурсів з найкоротшим шляхом не більше n.

Враховуючи граф ресурсу H, найкоротший шлях якого має $n + 1$ кроків, нехай набір атак мережного типу, які безпосередньо примикають до умови цілі c_h, дорівнює E_{n+1}.

Очевидно, $R(c_h \mid \forall c_{c \in CI} = T) \geq R(e \mid \forall c_{c \in CI} = T)$ виконується для всіх $e \in E_{n+1}$, оскільки c_h можна задовольнити будь-яким використанням в E_{n+1} (і ймовірність диз'юнкція подій не може бути меншою за ймовірність будь-якої події).

Не втрачаючи загальності, припустимо, що $e_{n+1} \in E_{n+1}$ є вторгненням мережного типуом поруч із c_h на найкоротшому шляху, і маємо $R(c_h \mid \forall c_{c \in CI} = T) \geq R(e_{n+1} \mid \forall c_{c \in CI} = T)$. Нехай E_n — множина атак мережного типу, найближча до e_{n+1}, $E \subseteq E_n$ — множина атак мережного типу на найкоротшому шляху, а $e_n \in E$ — атака мережного типу, розташований поруч із e_{n+1}.

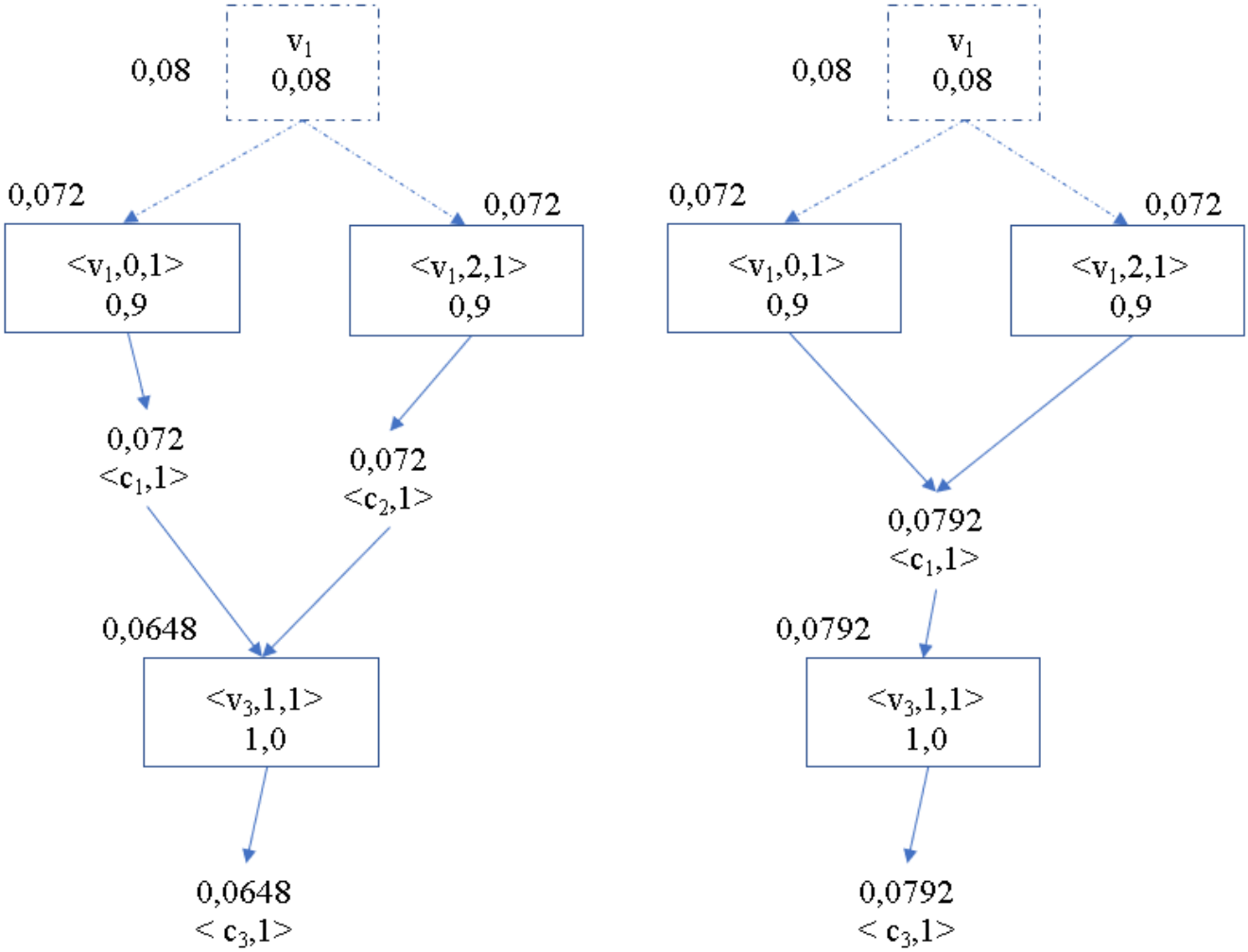


Рисунок 2.5 – Два приклади застосування d_3

Між атаками мережного типу в E та будь-яким іншим вторгненням мережного типуом в $E_n \backslash E$ щодо e_{n+1} не може бути жодних кон'юнктивних зв'язків, тому що інакше найкоротший шлях мав би більше ніж n + 1 кроків, що суперечить припущенню.

Отже, маємо, що $R(c_h \mid \forall c_{c \in CI} = T) \geq R(e_{n+1} \mid \forall c_{c \in CI} = T) \geq R(e_n \mid \forall c_{c \in CI} = T) \cdot R(e_{n+1} \mid \forall c_{<c,e> \in Pp \cup UPs} = T)$.

Тоді, згідно з індуктивною гіпотезою, маємо, що $R(e_n \mid \forall c_{c \in CI} = T)$ має бути не меншою за ту саму ймовірність, розраховану на найкоротшому шляху (довжини n), і, отже, закінчуємо доказ.

Наведений вище результат спрощує застосування d_3, оскільки найкоротший

шлях можна легко отримати за допомогою евристичного алгоритму, згаданого в [31].

Застосовуємо підхід до прикладу виконання, як показано на рисунку 2.2.

На основі таблиці 1 перший і другий шляхи атаки мають найменшу кількість кроків. Ліворуч на рисунку 2.6 зображено перший шлях.

Нормуючу константу можна розрахувати на основі цього шляху як $r' = 5{,}12 * 10^{-4}$. Правий бік зображує застосування моделі для повторного використання атак мережного типу, яка додає загальний батьківський ресурс для того самого типу ресурсів, представлений пунктирними лініями та квадратами.

Є два типи ресурсів, які повторно використовуються в цьому графу ресурсів, http і ssh. Застосовуючи вищеописаний метод, отримуємо ймовірність атаки p = 0,0052, і тому мережеве рознесення можна розрахувати як

$$d_3 = \frac{r'}{r} = \frac{5{,}12*10^{-4}}{0{,}0052}.$$

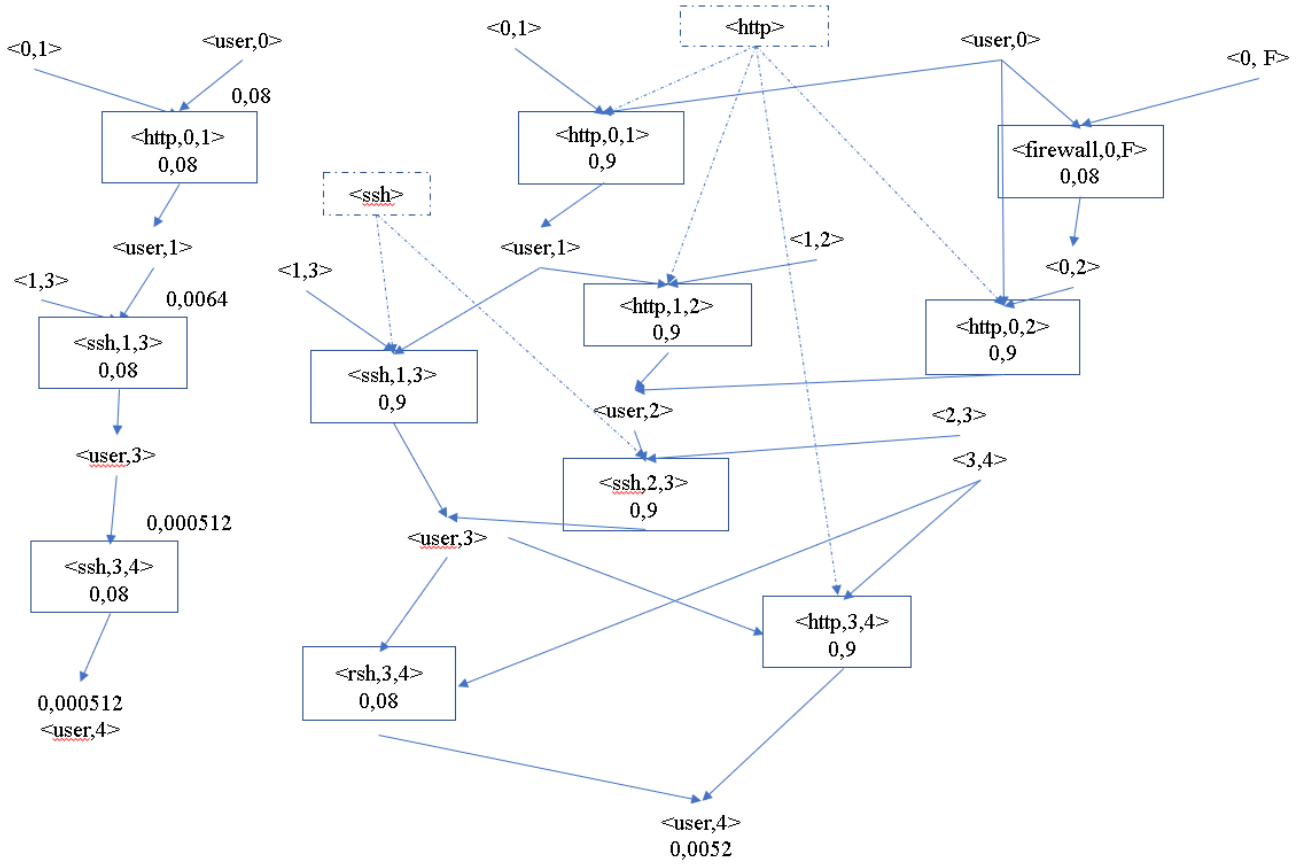


Рисунок 2.6 – Застосування d_3 до робочого прикладу

2.6 Застосування показників різноманітності мережі

Запропоновані показники різноманітності мережі засновані на абстрактних моделях мереж і атак.

Не менш важливо, як створити такі моделі для даної мережі.

У розділі обговорюються різні практичні питання щодо застосування метрик і надається приклад із створення екземплярів моделей.

2.5.3 Рекомендації щодо створення моделей мережевого різноманіття

Щоб застосувати запропоновані показники різноманітності мережі, необхідно зібрати необхідну вхідну інформацію.

Описуємо, як такі вхідні дані можуть бути зібрані з даної мережі, а також обговорюємо практичність і масштабованість.

Показник різноманітності d_1.

Щоб створити екземпляр d_1, потрібно зібрати інформацію про

1) хости (наприклад, комп'ютери, маршрутизатори, комутатори, брандмауери),

2) ресурси (наприклад, віддалено доступні послуги), і

3) схожість між ресурсами.

Інформація про хости та ресурси зазвичай вже доступна адміністраторам у вигляді карти мережі.

Сканування мережі допоможе зібрати або перевірити інформацію про активні послуги.

Також може знадобитися ретельна перевірка конфігурацій хоста (наприклад, статусу служб і правил брандмауера), оскільки сканування мережі може не виявити служби, які на даний момент вимкнені або приховані механізмами безпеки (наприклад, брандмауерами), але можуть бути знову ввімкнені один раз механізми безпеки скомпрометовані.

Збір та оновлення такої інформації для великої мережі, безумовно, вимагає значного часу та зусиль.

Для спрощення таких завдань існують автоматичне мережне сканування або інструменти на базі хостів.

Більше того, фокусування на віддалено доступних ресурсах дозволяє моделі залишатися відносно керованою та масштабованою, оскільки більшість хостів зазвичай мають лише кілька відкритих портів, але десятки чи навіть сотні локальних програм.

Проблема полягає в тому, щоб визначити подібність різних, але пов'язаних ресурсів.

Показник d_2-різноманітності.

Щоб створити екземпляр метрики різноманітності мережі d_2, заснованої на найменших зусиль, потрібно зібрати наступне, на додаток до того, що вже вимагає d_1,

1) підключення між хостами,
2) умови безпеки, які необхідні або підлягають ресурсам (наприклад, привілеї, довірчі відносини тощо), і
3) критичні активи.

Інформація про підключення зазвичай вже доступна як частина карти мережі.

Перевірити таку інформацію може допомогти мережевий сканер.

Ретельне вивчення конфігурацій хоста (наприклад, правил брандмауера) і параметрів програми (наприклад, політики автентифікації) зазвичай достатньо для визначення вимог для доступу до ресурсу (передумови), а також оцінки рівнів привілеїв програм і потужності ізоляція навколо таких програм виявить наслідки компрометації ресурсу (пост-умови).

Критичні активи можна визначити на основі потреб і пріоритетів організації.

Кількість додаткової інформації, необхідної для застосування d_2, можна порівняти з тим, що необхідно для d_1, оскільки ресурс зазвичай має невелику кількість попередніх і постумов.

Зібравши таку інформацію, можемо побудувати граф ресурсів, використовуючи існуючі інструменти для побудови традиційних графів атак через їх синтаксичну еквівалентність, і, як відомо, останній є практичним для реалістичних застосувань [46, 47].

Показник d_3-різноманітності

Щоб створити екземпляр імовірнісної метрики різноманітності мережі d_3, потрібно зібрати наступне, на додаток до того, що вже потрібно для d_2:

1. Граничні ймовірності спільних типів ресурсів.
2. Умовні ймовірності того, що ресурси можуть бути скомпрометовані, коли задоволені всі передумови.

Обидві групи ймовірностей представляють ймовірність того, що зловмисники можуть зламати певні ресурси.

Для кожного типу ресурсу може бути присвоєно різну ймовірність, якщо її можна оцінити на основі досвіду чи репутації (наприклад, історії минулих вразливостей, знайдених у тому самому чи схожому ресурсі).

Коли така оцінка неможлива або бажана, можна призначити те саме номінальне значення, як показано нижче.

Оскільки вразливість нульового дня зазвичай інтерпретується як уразливість, невідома чи оголошена публічно, її можна охарактеризувати за допомогою базових показників CVSS [73], як уразливість з недоступним рівнем усунення, непідтвердженою впевненістю звіту та максимальним загальним базовим балом. (і, отже, отримати консервативне значення метрики).

Таким чином, отримуємо номінальне значення 0,8, перетворюючи на ймовірність 0,08.

Для довідки найнижчий існуючий бал CVSS [18] на даний момент становить 1,7, тому 0,08 є досить низьким для вразливості нульового дня.

Після визначення ймовірностей застосування d_3 означає побудову байєсівських мереж і створення ймовірнісних висновків на основі мереж, чого можна досягти за допомогою багатьох існуючих інструментів (наприклад,

використовуємо OpenBayes [33]).

Хоча добре відомий факт, що висновки з використанням байєсівських мереж, як правило, нерозв’язні, результати моделювання показали, що конкретного висновку, необхідного для застосування метрики d_3, насправді можна досягти за розумних обчислювальних витрат [31].

2.6 Висновок

В даному розділі здійснено моделювання показників безпеки для оцінки резильєнтності іт-інфраструктур в умовах здійснення атак мережного типу. Зокрема, виконано формальне моделювання різноманітності мережі, подано особливості застосування моделі. В розділі розглянуто показник мережевого різноманіття, натхненного біорізноманіттям та показник різноманітності мережі на основі найменших зусиль. В розділі також представлено модель різноманітність на основі найменших зусиль для атаки. Розглянута імовірнісна різноманітність мережі та імовірнісна модель мережевого різноманіття. Надано рекомендації щодо створення моделей мережевого різноманіття.

3 УДОСКОНАЛЕНИЙ МЕТОД ПОБУДОВИ РЕЗИЛЬЄНТНИХ ІТ-ІНФРАСТРУКТУР В УМОВАХ ЗДІЙСНЕННЯ АТАК МЕРЕЖНОГО ТИПУ

3.1 Основи удосконаленого методу побудови резильєнтних ІТ-інфраструктур в умовах здійснення атак мережного типу

З метою розв’язку поставленої задачі було розроблено удосконалений метод побудови резильєнтних ІТ-інфраструктур в умовах здійснення атак мережного типу.

З цією метою важливо розглянути аспект функціонування резильєнтних систем. Для критично важливих комп’ютерних мереж (наприклад, тих, які відіграють роль нервової системи в критичних інфраструктурах, урядових або військових організаціях і підприємствах), адміністратори безпеки зазвичай виходять за рамки традиційних механізмів безпеки, таких як брандмауери та IDS.

Занепокоєння щодо перспективи розширеної стійкої загрози (APT) та прихованого зловмисного програмного забезпечення ІТ-інфраструктур зазвичай спонукає їх зрозуміти стійкість своїх мереж проти потенційних атак нульового дня, які використовують раніше невідомі вразливості.

Однак, хоча існує багато стандартів і показників для вимірювання відносної серйозності відомих вразливостей ІТ-інфраструктур.

З цією метою багатообіцяючим рішенням є концепція поверхні атаки, яка спочатку була запропонована для вимірювання ступеня безпеки програмного забезпечення ІТ-інфраструктур за трьома вимірами, а саме, точки входу та виходу (тобто методи, що викликають функції введення-виводу), канали (наприклад, TCP і UDP), а також ненадійні елементи даних (наприклад, записи реєстру або файли конфігурації).

Оскільки поверхня атаки покладається на такі внутрішні властивості програмного забезпечення ІТ-інфраструктур, що не залежать від зовнішніх факторів, як-от розкриття вразливостей або доступність атак мережного типу, вона,

природно, охоплює як відомі, так і невідомі вразливості [67] і стає хорошим кандидатом для розуміння загрози нульового дня. напади.

На відміну від вихідної метрики поверхні атаки, яка формально та кількісно визначена для одного програмного забезпечення IT-інфраструктур, більшість додатків на вищих рівнях абстракції (наприклад, на рівні мережі) обмежені інтуїтивним і якісним поняттям.

Прийняття такого неточного поняття неминуче втрачає більшу частину сили вихідної концепції у формальних і кількісних міркуваннях про ймовірність того, що система містить вразливі місця IT-інфраструктур.

Для розроблення удосконалений метод побудови резильєнтних IT-інфраструктур в умовах здійснення атак мережного типу розглянемо концепцію поверхні атаки до рівня мережі як формально визначену метрику безпеки, а саме поверхню мережевої атаки, для оцінки стійкості мереж до потенційних атак нульового дня на IT-інфраструктуру.

Існують дві основні проблеми підняття поверхні атаки до рівня мережі IT-інфраструктур.

По-перше, модель поверхні атаки покладається на додавання для агрегування балів, що несумісно з причинно-наслідковими зв’язками між різними ресурсами всередині мережі.

По-друге, існує парадокс, що єдиний спосіб уникнути дорогого розрахунку поверхні атаки – це виконати цей розрахунок.

З цією метою необхідним є розроблення методу, що базується на евристичних алгоритмах для вирішення цієї задачі з метою покращення ефективності запропонованих раніше.

Для побудови методу розглянемо ситуацію, показану на рисунку 3.1 (топологія модельованої комп’ютерної мережі). Припустимо, що зовнішній брандмауер дозволяє всі вихідні запити на з’єднання, але блокує всі вхідні запити до поштового сервера (h2) і файлового сервера (h3), включаючи запити від комп’ютерів у класі (h25); внутрішній брандмауер дозволяє всі вихідні запити з h4,

але блокує всі вхідні запити, крім запитів з h2.

Також припускаємо, що основна турбота полягає в захисті хосту адміністратора (h4), що містить критичні активи. Виходячи з таких припущень, можемо легко побачити, що зловмисник на h0 потенційно може піти шляхом атаки, наприклад, h1 → h2 → h4, щоб скомпрометувати h4.

Тепер розглянемо питання щодо можливості застосування концепції поверхні атаки, яка визначена лише для кожного окремого ресурсу, до такої мережі, щоб обчислити рівень їїбезпеки (наприклад, у термінах h4).

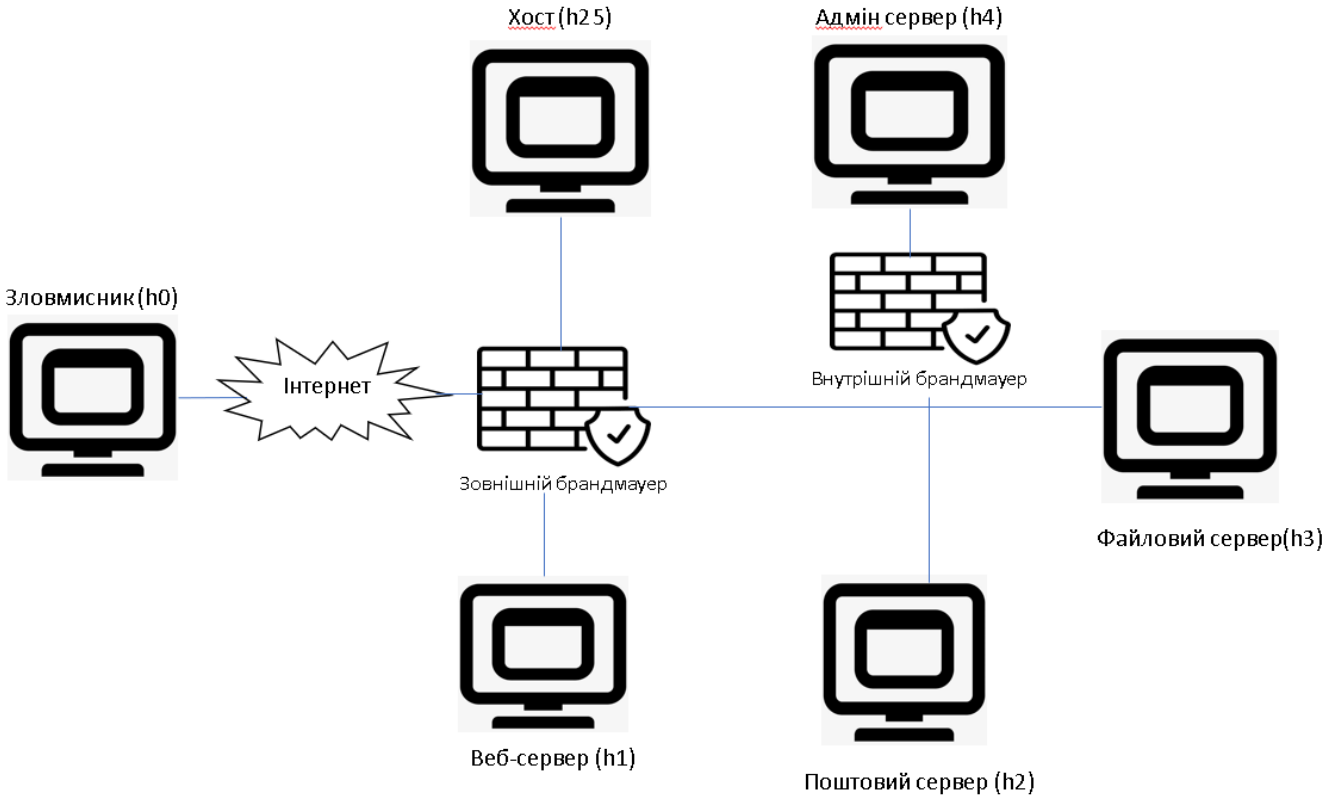


Рисунок 3.1 – Топологія модельованої комп’ютерної мережі приклад

Два очевидні рішення полягають у тому, щоб безпосередньо застосувати метрику, або розглядаючи всю мережу як єдину IT-інфраструктуру, або спочатку застосувавши її до кожного ресурсу окремо, а потім додавши усі отримані результати разом.

Оскільки операція додавання є асоціативною, обидва рішення дають

однаковий результат, тобто загальну кількість методів, каналів і ненадійних елементів даних відповідно.

Проблема полягає в тому, що така операція додавання несумісна з причинно-наслідковими зв'язками між мережевими ресурсами IT-інфраструктури, які можуть бути як кон'юнктивними, так і диз'юнктивними.

Застосування цього на шляху атаки, наприклад, h1 → h2 → h4, є менш значущим.

Оскільки це означає, що довший шлях атаки дасть більшу поверхню атаки на IT-інфраструктуру (менш безпечний), але довший шлях зазвичай вимагає більше зусиль від зловмисників (більш безпечний), що є протиріччям.

Таким чином, проблема полягає в тому, як об’єднати поверхню атаки мережевих ресурсів IT-інфраструктури, поважаючи їх причинно-наслідкові зв’язки.

Також існує проблема розрахунку поверхні атаки. Тому важливо дослідити можливість зменшення зусиль та уникнути обчислення поверхні атаки для тих ресурсів, які не сприяють кінцевому результату.

3.2 Розрахунок ймовірності мережевої атаки

Для розроблення моделі поверхні мережевої атаки необхідно розглянути концепцію поверхні атаки до рівня мережі за два кроки: механізм перетворення поверхні атаки програмного додатка на його ймовірність атаки, а саме розрахунок ймовірності атаки на основі зіставлення між поверхнею атаки та загальною системою оцінки вразливостей (CVSS), та розгляд взаємозв’язку між ресурсами та перетворення їх у загальну ймовірність атаки, та об’єднання ймовірності атаки різних мережевих ресурсів в єдиний показник поверхні мережевої атаки.

3.2.1 Розрахунок ймовірності атаки на основі CVSS

З метою розрахунку імовірності атаки на основі CVSS, необхідним є

дослідження проблеми операції додавання, яка використовується в поверхні атаки, несумісна з причинно-наслідковими зв’язками між досліджуваними мережевими ресурсами.

Головна ідея полягає в тому, щоб перетворити поверхню атаки кожного ресурсу програмного забезпечення в ймовірність атаки (відносну ймовірність того, що програмне забезпечення містить принаймні одну уразливість нульового дня, яку можна використовувати), яку потім можна об’єднати для різних ресурсів на основі їх причинно-наслідкових зв’язків.

Оскільки поверхня атаки надає вказівку як на серйозність (представлена ваговими показниками, тобто права доступу та привілеї), так і ймовірність (представлена підрахунками, тобто загальною кількістю методів, каналів і ненадійних елементів даних) потенційних вразливостей, перетворення буде здійснюватися у два кроки наступним чином.

Спочатку для кожної групи методів досліджуємо зіставлення між поверхнею атаки та загальною системою оцінки вразливостей (CVSS) [73], щоб перетворити права доступу та привілеї поверхні атаки на базову оцінку CVSS.

По-друге, на рівні програмного забезпечення об'єднуємо базові оцінки різних груп методів в єдину ймовірність атаки для всього програмного забезпечення.

Метод перетворення на рівні групи.

Спочатку коротко розглянемо поняття поверхні атаки та CVSS. Як показано в першому стовпці таблиці 3.3, CVSS визначає шість базових показників у двох групах: група доступності, що включає вектор доступу (AV), складність доступу (AC) та автентифікацію (Au), і група впливу, включаючи вплив на конфіденційність (C), вплив цілісності (I) і вплив доступності (A) (можливі значення кожного показника та відповідні їм числові оцінки також наведені в таблиці) [73].

Другий стовпець таблиці 3 показує різні права доступу та привілеї та їх числові

Важливо врахувати, що ймовірність атак тут призначена лише як відносна метрика для порівняння між різними програмними програмами, а не фактична

ймовірність атак, яку на практиці отримати, як правило, неможливо.

Оскільки і група доступності CVSS, і права доступу до поверхні атаки є передумовами для використання вразливості, їх усі значення можна зіставити разом.

Аналогічно, група впливу CVSS і привілеї поверхні атаки представляють наступні умови використання вразливості, і, отже, відображаються разом.

Точне відображення для цих двох демонів IMAP показано в останньому стовпці таблиці 3.1.

Кожен вектор CVSS відображає відповідне право доступу або привілею, показані в тому ж рядку в другій колонці.

Таблиця 3.1 – Відображення поверхні атаки на базові показники CVSS для Courier IMAP Server і Cryus IMAP Server

CVSS (Base Metrie Group)	Методи покриття атак			Вектори
AV:(L:0.395.A:0.646,N: 1 .01	Права доступу	anoymous	1	|AV:N.AC:L,Au:N)
AC:|H:0.35,M:0.61,L:0.71)		unauthenticated	1	|AV:N,AC:L,Au:N]
Au:[M:0.45,S:0.56,N:0.704]		authenticated	3	|AV:N,AC:M.Au:S)
		admin	4	(AV:A.AC:H.Au:M)
C:[N:0.0,P:0.275,C:0.66]	привілегії	authenticated	3	[C:P,I:P,A:C]
I:[N:0.0,P:0.275,C:0.66]		cyrus	4	[C:C.I:C.A:C|
A:(N:0.0,P:0.275,C:0.66)		root	5	[C:C,I:C.A:C]

Зіставлення встановлюється на основі розуміння програмного забезпечення, включаючи його канали та ненадійні елементи даних (тому повторний розрахунок при конвертації базових балів у ймовірність атаки не потрібний).

Наприклад, у третьому рядку право доступу з автентифікацією відображається на мережу IT-інфраструктури для вектора доступу (тобто, AV:N), оскільки сокет UNIX у програмному забезпеченні має локальне автентифіковане право доступу, що означає, що зловмисники можуть отримати локальний

автентифікований доступ прямо через мережу.

Крім того, призначаємо складність доступу до середовища (тобто AC:M), оскільки автентифіковане право доступу відповідає опису складності доступу до середовища.

Нарешті, призначаємо автентифікацію одиночній (тобто Au:S), оскільки для доступу потрібен один автентифікований сеанс у програмному забезпеченні IT-інфраструктури.

Аналогічно, у п’ятому рядку автентифікована привілей відображається на частковий вплив на конфіденційність, частковий вплив на цілісність та повний вплив на доступність (тобто C:P, I:P, A:C), оскільки автентифікований привілей передбачає доступ до 13 файлів у програмному забезпеченні дозволяє змінювати деякі системні файли або дані, а також може зробити систему непридатною для використання, видаливши важливі файли.

Важливо врахувати, що оскільки це зіставлення ґрунтується на розумінні прав доступу, привілеїв та програмного забезпечення, різні адміністратори можуть в кінцевому підсумку призначати зіставлення різними та непорівнянними способами. Однак, оскільки показники є відносними і призначені для порівняння різних конфігурацій однієї мережі, результати будуть значущими, якщо зіставлення є узгодженим у різних конфігураціях.

Як показано в таблиці 3.1, зіставляємо всі методи цих двох програм з відповідними базовими метриками CVSS, а потім обчислюємо загальний базовий бал за формулою CVSS [73], як показано в таблиці 3.2. Методи розділені на групи (перший стовпець) відповідно до подібних привілеїв (другий стовпець) і прав доступу (третій стовпець). У четвертому та п’ятому стовпцях показано загальну кількість точок входу та виходу в кожній групі. Наступні два стовпці показують відображений вектор CVSS та розрахований базовий бал для кожної групи.

Перетворення програмного рівня. Тепер, коли розраховано базову оцінку для кожної групи методів, можемо перетворити поверхню атаки на ймовірність атаки, що представляє відносну ймовірність використання програмного забезпечення

через принаймні одну вразливість нульового дня.

Припустимо, що на поверхні атаки є всього h груп методів.

Таблиця 3.2 – Групи методів та їх базові оцінки для Courier IMAP Server і Cryus IMAP Server

Метод	доступ	Права	DEP	DExp	Вектор	Основа	ймовірність
Courier							
Ml	root	unauthenticatcd	28	17	| AV: 1.0.AC:0.71 .Au :0.704,C:0.66.I:0.66. A :0.66)	10	0.000315
M2	root	authenticated	21	10	|AV:1.0.AC:0.61.Au:0.56.C:0.66.I:0.66,A:0.66]	8.5	0.000184
M3	authenticated	authenticated	113	28	1 AV: 1.0.AC:0.61 .Au:0.56.C:0.275.I:0.275.A:0.66)	7.5	0.000809
Cyrus							
Ml	cyrus	unauthenticatcd	16	17	| AV: 1.0.AC0.71 .Au :0.704.C:0.66.I:0.66.A:0.66]	10	0.000132
M2	cyrus	authenticated	12	21	|AV:1,0.AC:0.61.Au:0.56.C:0.66.I:0.66.A:0.66]	8.5	0.000112
M3	cyrus	admin	13	22	|AV:0.646,AC:0.35.Au:0.45.C:0.66.I:0.66,A:0.66]	6.3	0.0000882
M4	cyrus	anonymous	12	21	|AV:l.0.AC:0.7I.Au:0.7O4.C:().66.I:0.66.A:0.661	10	0.000132

Нехай b_i та s_i $(1 \le i \le h)$ позначають базову оцінку та кількість методів кожної групи відповідно.

Припустимо, що в середньому буде існувати одна вразливість нульового дня для кожні n методів, і ймовірність виявлення такої вразливості зловмисниками дорівнює r_0 (k і r_0 обидва призначені як нормалізуючі константи; див. нижче для додаткової інформації).

У рівнянні 3.1 базовий бал, поділений на його діапазон 10, дає ймовірність того, що вразливість у цій групі може бути використана; множення цього на r_0 дає ймовірність того, що метод може бути як відкритий, так і використаний; s_i/k показує кількість вразливостей із методів s_i в цій групі; отже, рівняння дає ймовірність r того, що програмне забезпечення містить принаймні одну уразливість нульового дня проти IT-інфраструктури, яку можна використовувати:

$$r = \prod_{i=1}^{h}(1 - r_0 \frac{b_i}{10})^{\frac{s_i}{k}} \quad (3.1)$$

Приклад 3.1. Припустимо k = 30 і r_0 = 0,08, можемо обчислити r для обох програм наступним чином.

Для Courier $r = 1-(1-0{,}08*10/10)^{45/30} *(1-0{,}08*8{,}5/10)^{31/30} *(1-0{,}08*7{,}5/10)^{141/30}$

= 0,384, і так само для Cryus, r = 0,273.

Важливо зауважити, що справжні значення параметрів k і r_0, безумовно, неможливо отримати на практиці, тому призначені лише як нормалізуючі константи, вибрані для забезпечення розумного значення для r.

Поки ці значення залишаються незмінними між різними програмними продуктами ІТ-інфраструктури, рівняння дасть відносне значення, достатнє для порівняння придатності для використання різного програмного забезпечення на основі як серйозності (представленої базовими балами b_i), так і кількості (представлених кількістю методів s_i).) потенційних вразливостей нульового дня проти ІТ-інфраструктури.

3.2.2 Розрахунок імовірності атаки на основі графа

Наступним кроком розроблено удосконаленого методу побудови резильєнтних ІТ-інфраструктур в умовах здійснення атак мережного типу є врахування взаємозв’язків між різними розмірами поверхні атаки, напр. канали та ненадійні елементи даних розглядаються лише як непрямі вхідні дані в процесі відображення, що відображає доступність та вплив методів відповідно.

Під час роботи фіксуємо зв’язки між різними ресурсами за допомогою моделі, яка представляє можливі атаки між ресурсами ІТ-інфраструктури, а потім агрегуємо загальну ймовірність атаки щодо критичної умови або активу, сформованого як умову цілі.

Далі здійснюється поєднання різних вимірів поверхні атаки, розглядаючи точку зору зловмисників, коли атака зазвичай вимагає каналів зв’язку (вимір каналу) для доступу до методів і викликає методи для маніпулювання ненадійними елементами даних для досягнення своєї мети.

Якщо ПЗ ІТ-інфраструктури ізольовано від КМ і не має пов’язаних з нею каналів, зловмисники не зможуть здійснити атаки на програмне забезпечення незалежно від кількості методів і ненадійних елементів даних.

Аналогічно, якщо ПЗ не має жодного методу, зловмисники не зможуть отримати доступ до ненадійних елементів даних. Це спостереження показує, що для завершення атаки зазвичай потрібна комбінація трьох вимірів. Перетворення поверхні атаки до ймовірності атаки на основі графа вимагає двох кроків.

1. Спочатку для кожної групи ресурсів у трьох вимірах поверхні атаки обчислюємо ймовірність для всієї групи ресурсів на основі рівняння 3.2.

2. По-друге, на рівні програмного забезпечення об'єднуємо ймовірності для кожної групи в єдину ймовірність атаки на основі байєсівських висновків наступним чином.

Метод перетворення на рівні групи. По-перше, поділяємо методи на групи на основі пари <права доступу, привілеї>, таким чином, що методи в одній групі вимагають ідентичних прав доступу та призводять до ідентичних привілеїв. Перший стовпець таблиці 3.2 містить назву групи для кожної групи методів, і просто використовуватимемо M1 в Courier, щоб посилатися на групу методів, обмежених правом неавтентифікованого доступу та призвести до прав root в Courier.

У групі M1 Courier зловмисникам потрібно використовувати лише один метод із 45, щоб отримати відповідний привілей; однак, не маючи знань про методи в програмному додатку, зловмисники можуть використовувати кілька методів в одній групі. Враховуючи це, визначаємо ймовірність атаки однієї групи методів як ймовірність скомпрометувати хоча б один метод із групи.

Припустимо, що маємо повністю s_i-методи в одній групі, і нехай b і r_0 позначають базову оцінку та ймовірність виявлення зловмисниками одного методу відповідно.

У рівнянні 3.2 базовий бал, поділений на його діапазон 10, дає ймовірність знайти метод у програмній програмі, який можна використовувати; помноження цього на r_0 дає ймовірність того, що метод можна як виявити, так і використати.

Як пояснювалося раніше, r_0 призначено лише як нормалізуючу константу:

$$r = 1 - (1 - r_0 \frac{b_i}{10})^{s_i} \quad (3.2)$$

Приклад 2. Для порівняння Courier і Cryus беремо r0 як відношення вибору одного методу на тисячу рядків вихідного коду.

Кількість рядків вихідного коду для Courier і Cryus становить 138,283 і 236,321 відповідно. Тому легко отримати r_0 = 0,00723 для Courier і r_0 = 0,00423 для Cryus. Можемо обчислити r для M2 для обох програмних додатків наступним чином. Для Courier $r = 1-(1-0{,}00723 * \frac{8{,}5}{10})^{31} = 0{,}174$, а також M2 в Cryus, r = 0,112.

Перетворення програмного рівня. Щоб змоделювати ймовірність атаки проти ПЗ IT-інфраструктури на основі поверхні атаки, спочатку потрібна модель взаємозв'язків між трьома вимірами поверхні атаки.

Граф поверхні атаки для набору поверхневих ресурсів атаки P_A від <M,C,D> і набір привілеїв Con, необхідне відношення $P_p \subseteq Con \times P_A$ і відношення $P_i \subseteq P_A \times Con$. Граф поверхні атаки H_A — це орієнтований граф $H_A(P_A \cup Con, P_p \cup P_i)$ ($P_A \cup Con$ — множина вершин, а $P_p \cup P_i$ — множина ребер). resource.P_p і resource.P_i позначають як запит відношень і мають на увазі цей ресурс відповідно.

1. Зловмисники можуть отримати доступ до ресурсу C, лише якщо C.P_p отримують зловмисники.
2. Ресурс M може бути викликаний C тільки тоді, коли C.$P_i \succ$ M.P_p.
3. До ресурсу D можна отримати доступ лише за умови M.$P_i \succ$ D.P_p .
4. На рисунку 3.2 зображено відповідний граф поверхні атаки на IT-інфраструктуру як для Courier, так і для Cryus (зверніть увагу, що малюнок може виглядати схожим на граф атаки, але тут він вказує на сукупність трьох вимірів всередині поверхні атаки).

Таблиця 3.2 – Канали IMAP Daemon та елементи недовірених даних

Канали	Дані
Type Access Rights	Group Type Access Rights
TCP remote unauthenticated SSL remote unauthenticated UNIX socket local authenticated	F1 file root F2 file authenticated F3 file world
Cyrus Channels	Untrusted Data Items
TCP remote unauthcnticatcd SSL remote unauthcnticatcd UNIX socket local authenticated	FI file root F2 file cyrus F3 file cyrus

Кожне квадратне поле на рисунку 3.2 представляє ресурс на поверхні атаки (наприклад, сокет TCP, SSL і UNIX, які є каналами на поверхні атаки, представлені як підключення для програмних додатків); краю вказують від попередніх умов до ресурсів (наприклад, <TCP з'єднання > і < віддалений неавтентифікований > до M1) або від ресурсів до пост-умов (наприклад, від M1 до <корінь>).

Зокрема, зловмисники можуть отримати прямий доступ до каналів, які моделюються як ресурси, пов'язані початковими умовами на графі поверхні атаки, оскільки початкові умови вважаються вже задоволеними.

Методи можуть бути викликані зловмисниками, лише якщо відповідні канали пов'язані з еквівалентним або вищим привілеєм.

Наприклад, зловмисники, які переходять із каналу UNIX-сокета, можуть отримати доступ до M1 (сокет UNIX має привілей локальної автентифікації, який вищий, ніж необхідне право доступу M1, неавтентифікований). Аналогічно, під час надсилання ненадійних елементів дати привілеї, отримані від методів, мають бути еквівалентними або вищими, ніж право доступу ненадійних елементів даних. У таблиці 5 для надсилання даних на F1 в Courier потрібен корінь, що означає, що M3 з автентифікацією не має достатнього права доступу для надсилання даних до F1.

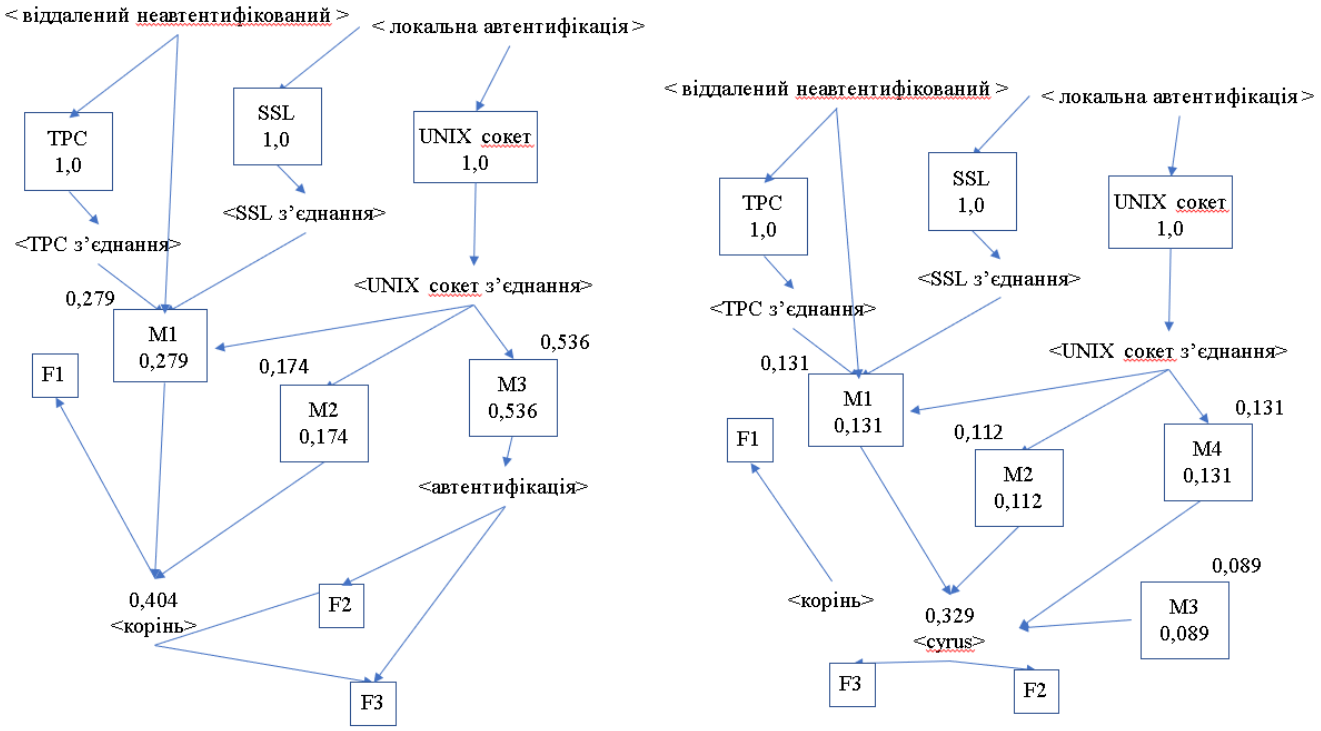


Рисунок 3.2 – Графи поверхні атаки для Courier (ліворуч) і Cryus (праворуч)

На графі поверхні атаки, оскільки шляхи атаки ведуть до критичних ресурсів або посилених привілеїв, ймовірність атаки програмного додатка можна представити як середнє зусилля атаки шляхом поєднання всіх шляхів атаки. Для цієї мети визначаємо загальну поверхню атаки як умовну ймовірність того, що зловмисник скомпрометує даний критичний актив у програмному додатку. Умову мети можна використовувати для моделювання будь-яких ресурсів або умов на графі поверхні атаки.

З умовою мети, наведеною на графі поверхні атаки, загальну ймовірність атаки можна розрахувати за допомогою байєсівського висновку, щоб об'єднати всі можливі шляхи атаки.

Наприклад, загальна ймовірність атаки проти IT-інфраструктури становить 0,404 (таблиця умовних ймовірностей показана збоку) за умови, що всі початкові умови задоволені, наприклад, віддалена автентифікація та локальна автентифікація.

Аналогічно, загальна ймовірність атаки для Cryus становить 0,329. Обидва методи обчислення ймовірності атаки (імовірність атаки на основі CVSS і на основі

графа) показують, про більшу ймовірність атаки порівняно з Cryus.

3.2.3 Агрегація ймовірностей атак всередині мережі

Наступник кроком удосконаленого методу побудови резильєнтних IT-інфраструктур в умовах здійснення атак мережного типу є здійснення процедури об'єднання поверхні атаки всіх ресурсів мережі IT-інфраструктури в єдине значення поверхні атаки мережі.

З цією метою було застосовано два способи для агрегування поверхні атаки ресурсів в мережі: підхід на основі найкоротшого шляху та підхід на основі байєсівської мережі (BN), які відображають найгірший сценарій (тобто з щодо зловмисників, які йдуть за найпростішим шляхом атаки) та сценарієм середнього випадку, відповідно.

Щоб проілюструвати цю ідею, на рисунку 3.3 показано частковий граф ресурсів для прикладу, а пунктирну лінію наразі можна проігнорувати і знадобиться пізніше (зверніть увагу, що як граф ресурсів, продемонстрований тут, так і граф поверхні атаки, який синтаксично еквівалентний графу атак, але граф ресурсів моделює атаки нульового дня на рівні мережі, тоді як граф поверхні атаки моделює відомі вразливості на рівні програмного забезпечення).

Зокрема, кожна пара у відкритому тексті є умовою, пов'язаною з безпекою, наприклад, з'єднання <джерело, призначення> або привілей <привілей, хост>, і кожна трійка всередині поля є вторгненням мережного типуом нульового дня <ресурс, джерело, призначення>.

Імовірність у кожній коробці є ймовірністю атаки відповідного ресурсу.

Приклад 3. На рисунку 3.3 для підходу на основі найкоротшого шляху можемо обчислити ймовірність атаки для найкоротшого шляху, позначеного пунктирною лінією, <IPCop,0,F> → <Courier,0,2>→ <FirewallBuilder, 2,4>, ймовірність може бути розрахована як r = 0,48 ∗ 0,384∗0,04 = 0,0074.

Приклад 4. Для підходу на основі BN можемо просто розглядати рисунок 18

як байєсівську мережу, причому ймовірність атаки кожного ресурсу розглядається як умовна ймовірність того, що відповідний ресурс може бути використаний за умови, що всі його передумови задоволені, а потім виконати байєсівський висновок, щоб обчислити загальну ймовірність атаки [140]. У прикладі можемо обчислити ймовірність досягнення зловмисниками <user,4> як $r_{goal} = 0{,}016$.

З наведених вище прикладів стає зрозуміло, що моделі ймовірностей атак на поверхні можуть допомогти подолати ключове обмеження існуючої метрики безпеки n-нульового дня (яка також використовує підхід на основі найкоротшого шляху), тобто він не може розрізняти різні ресурси на основі їх відносної ймовірності атаки. Більш формально, наступне формально визначає поняття поверхні мережевої атаки.

Поверхня мережевої атаки для мережі з набором ресурсів P ймовірність атаки r(p), як визначено в рівнянні 1 або рівнянні 2, для кожного $p \in P$, графа ресурсів H і заданої умови $c_h \in H$:

1. Нехай AP позначає сукупність усіх шляхів атаки в H, що закінчуються на c_h, і для кожного $ap \in AP$ нехай P(ap) позначає набір ресурсів, залучених до ap і позначає $p(ap) = \prod_{p \in P(ap)} r(p)$. Називаємо $\max(\{r(ap) : ap \in AP\})$ (де max(.) повертає максимальне значення набору) найгіршою поверхнею мережевої атаки c_h.

2. Нехай B = (H' ,θ) — BN, де H' — H, анотований ймовірностями атаки, а θ — набір параметрів BN (BN більш точно визначено в [140], а деталі тут опущені), і нехай C_1 — набір умов без батьків у H', називаємо $r = R(c_h \mid \forall_{c \in C1} c = \text{True})$ середньою поверхнею мережевої атаки щодо випадків c_h.

Важливо зазначити, що вони потенційно можуть бути перетворені в інші форми для різних інтерпретацій. Наприклад, враховуючи поверхню атаки мережі r як ймовірність, можемо легко перетворити p на еквівалентну кількість методів s із заданим базовим балом b, інвертуючи рівняння 1 як: $s = k \log_{1-r0} (1 - r)$. Тому можемо оцінити мережу як єдину програмну систему з поверхнею атаки, що складається з s методів з базовою оцінкою b (яку також можна зіставити з правами доступу та привілеями, якщо необхідно). Крім того, можемо перетворити p назад в

еквівалентну кількість вразливостей нульового дня як $\log_{0,08}r$, що є простим підрахунком. на основі метрики, корисної для інтерпретації та порівняння (будемо використовувати цей метод у алгоритмах та моделюванні).

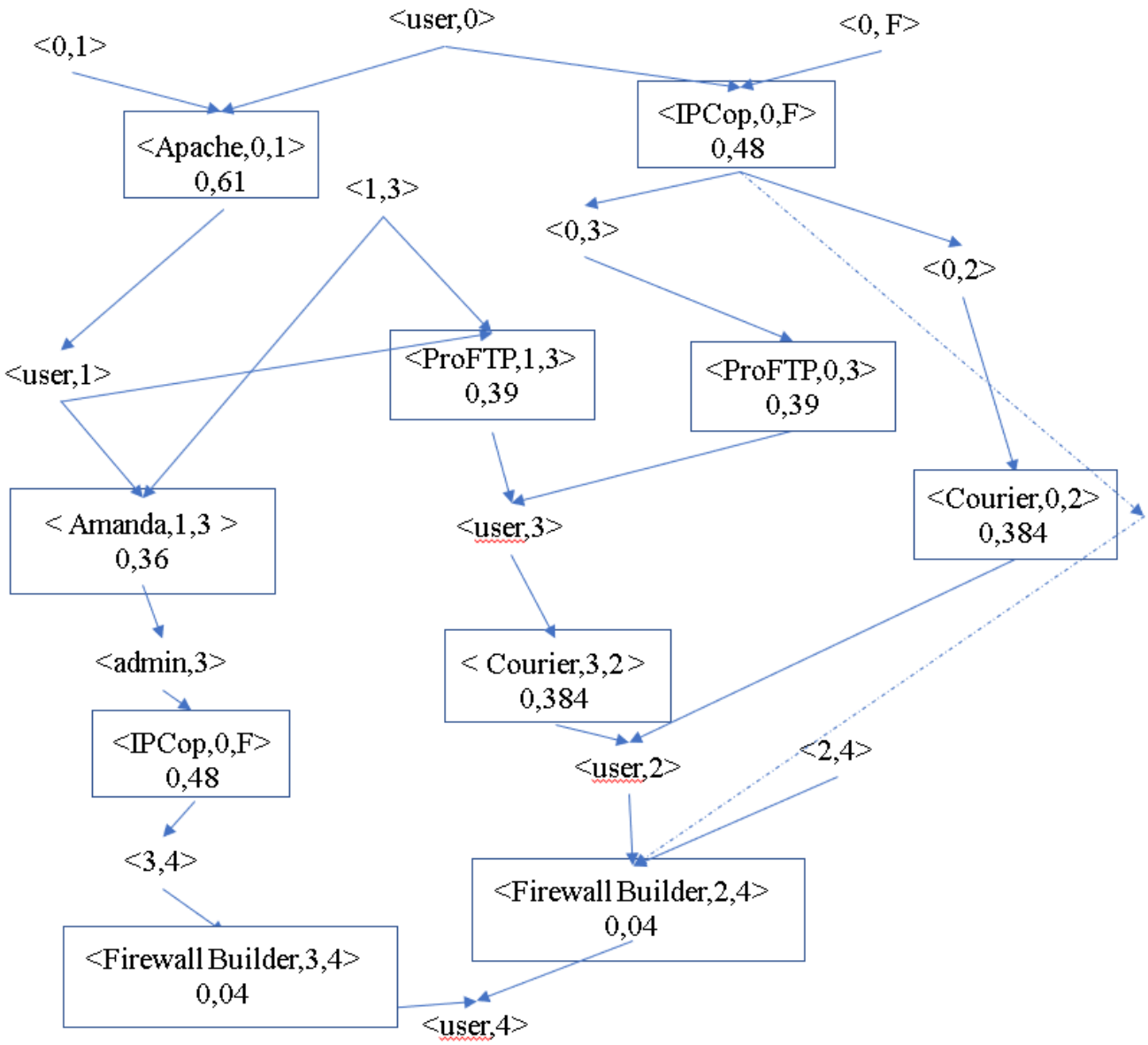


Рисунок 3.3 – Граф мережевих ресурсів із ймовірністю атаки для мережі

3.3 Евристичні алгоритми для обчислення поверхні мережевої атаки на IT-інфраструктуру

З метою обчислення поверхні мережевої атаки необхідним є застосування

евристичних алгоритмів. Для мережі з набором ресурсів P і припустимо, що справжнє значення поверхні мережевої атаки дорівнює r_{true}, а обчислене значення — r_{cal} (припускаємо, що всі значення засновані на підрахунку), хотіли б мінімізувати помилку $\frac{|r_{true}-r_{cal}|}{r_{true}}$ при розрахунку поверхні атаки не більше ніж на заданий відсоток ресурсів .

Наприклад, кількість рядків програмного забезпечення, згаданого в прикладі на рисунку 3.1: Nginx, IPCop, Apache, MySql, ядро Linux і Google Chrome.

Необхідно зауважити, що, хоча вищезазначене може здатися стандартною проблемою оптимізації, насправді це не так, оскільки цільова функція $\frac{|r_{true}-r_{cal}|}{r_{true}}$ містить невідоме значення r_{true}, обчислення якого означало б обчислення поверхні атаки для всіх ресурсів і суперечило б самій меті зниження вартості.

3.4 Застосування евристичних алгоритмів випадкового вибору та вибору частоти

Основне спостереження полягає в тому, що, оскільки можемо розрахувати лише певний відсоток ресурсів у рамках даного бюджету, помилка визначає порядок обчислення серед усіх ресурсів. Спочатку розглядаються кілька простих евристик для вибору ресурсів у правильному порядку, наприклад, шляхом дослідження структурних властивостей графа ресурсів.

Зосередимося на найгіршій поверхні мережевої атаки.

Випадковий вибір. Найочевиднішим рішенням, ймовірно, є просто вибір ресурсів абсолютно випадковим способом, а саме евристичним методом випадкового вибору. Хоча алгоритм випадкового вибору, ймовірно, далекий від оптимального, він дає базову лінію для порівняння з іншими евристичними алгоритмами, які запропонуємо. Наприклад, на рисунку 3.3, якщо бюджет передбачає розрахунок поверхні атаки щонайбільше двох ресурсів, то серед $\binom{6}{2}$= 15 можливих варіантів найгіршим результатом є r = 0,46 з коефіцієнтом помилок 0,76,

тоді як найкращим результатом є r = 1,73 з частотою помилок 0,109. Очевидно, що ця евристика може привести до рішення, яке далеке від оптимального.

Вибір частоти. Ідея цієї евристики полягає в тому, що, оскільки один і той самий ресурс може з'являтися на кількох хостах всередині мережі, обчислення поверхні атаки для ресурсів, які найчастіше зустрічаються, надасть найбільшу інформацію з однаковою вартістю. Наприклад, на рисунку 3.3 бачимо, що IPCop, Firewall Builder, Courier і ProFTP з'являються двічі серед 10 атак мережного типу. Отже, якщо бюджет дорівнює двом, то підрахунок будь-яких двох з них відкриє 4/10 атак мережного типу (найкращий результат — r = 1,73 з коефіцієнтом помилок 0,109, вибравши обчислення Firewall Builder і Courier. І найгірший результат дорівнює r = 0,60 з частотою помилок 0,69, вибравши обчислення IPCop і ProFTP).

Топологічний порядок. Ідея полягає в тому, що, оскільки вузли, ближчі до першого та останнього вузлів графа ресурсів (у сенсі топологічного сортування), мають тенденцію спільно використовувати більше шляхів атаки (наприклад, останні два атаки мережного типу спільні для всіх шляхи на рисунку 3.3), це може допомогти вибрати ресурси на основі топологічного порядку серед атак мережного типу. Розглядаємо як топологічний порядок, так і зворотний топологічний порядок евристики, які вибирають ресурси в тому ж і протилежному порядку як топологічне сортування, відповідно. Наприклад, на рисунку 3.3, припустимо, що бюджет дорівнює двом, евристика топологічного порядку може вибрати Apache і IPCop (результат буде r = 0,60 з частотою помилок 0,69), тоді як у зворотному топологічному порядку можна вибрати Firewall Builder і Courier (результат буде бути r = 1,73 з частотою помилок 0,109).

Найкоротший шлях. Ця евристика розпочинає обчислення з ресурсів на шляху з найменшою кількістю атак мережного типу (наприклад, шлях, зображений пунктирною лінією на рисунку 3.3), який, хоча і не завжди правильний шлях з точки зору кінцевого результату, може служити гарна відправна точка. Наприклад, на рисунку 3.3, якщо бюджет дорівнює двом, то евристика найкоротшого шляху вибере Courier та Firewall Builder на шляху пунктирної лінії (результат r = 1,73 з

частотою помилок 0,109). У конкретному прикладі цей шлях є правильним шляхом для розрахунку кінцевого результату, тому більший бюджет потенційно дасть більш точний результат.

Наведені вище евристики можуть не дати хороших результатів, якщо кожна з них використовується окремо, але їх поєднання призводить до алгоритмів з хорошою продуктивністю. Далі представлені два таких алгоритми.

Евристичний алгоритм Mpath-Topo. Цей алгоритм поєднує наведений вище топологічний порядок і евристики найкоротшого шляху наступним чином. Спочатку застосовуємо евристику найкоротшого шляху, щоб вибрати M (цілочисельний параметр) найкоротших шляхів, які ранжуються на основі кількості унікальних атак мережного типу, як початкові точки.

Оскільки на кожному такому шляху немає порядку між ресурсами, потім застосовуємо евристичний топологічний порядок для сортування всіх шляхів, а також тих, які не знаходяться на таких шляхах.

Послідовність ресурсів R-набору, ресурсний набір MS з найменшої кількості атак мережного типу у H з M шляхів (найменша кількість шляхів вторгнення мережного типу можна знайти за допомогою евристичного алгоритму), а T набір ресурсів незалежно від ресурсів у MS з топологічним порядком. Основний цикл представляє процес вибору ресурсу.

Обмежені бюджетом K, ресурси спочатку вибираються з набору MS і продовжують вибирати з набору T.

Остаточна послідовність ресурсів представлена у наборі R.

Евристичний алгоритм Keynode. Цей евристичний алгоритм ґрунтується на ідеї, що ресурс є більш важливим для визначення кінцевого поверхневого значення p мережної атаки, якщо зміна його значення може призвести до значних змін, наприклад, зміна оптимального шляху (шляху, обраного для розрахунку кінцевої результат), або зміна поточного обчисленого результату п.

Потім об’єднуємо цю евристику з евристикою топологічного порядку, щоб сформувати алгоритм, показуємо лише зміну p, яку можна замінити зміною

оптимального шляху.

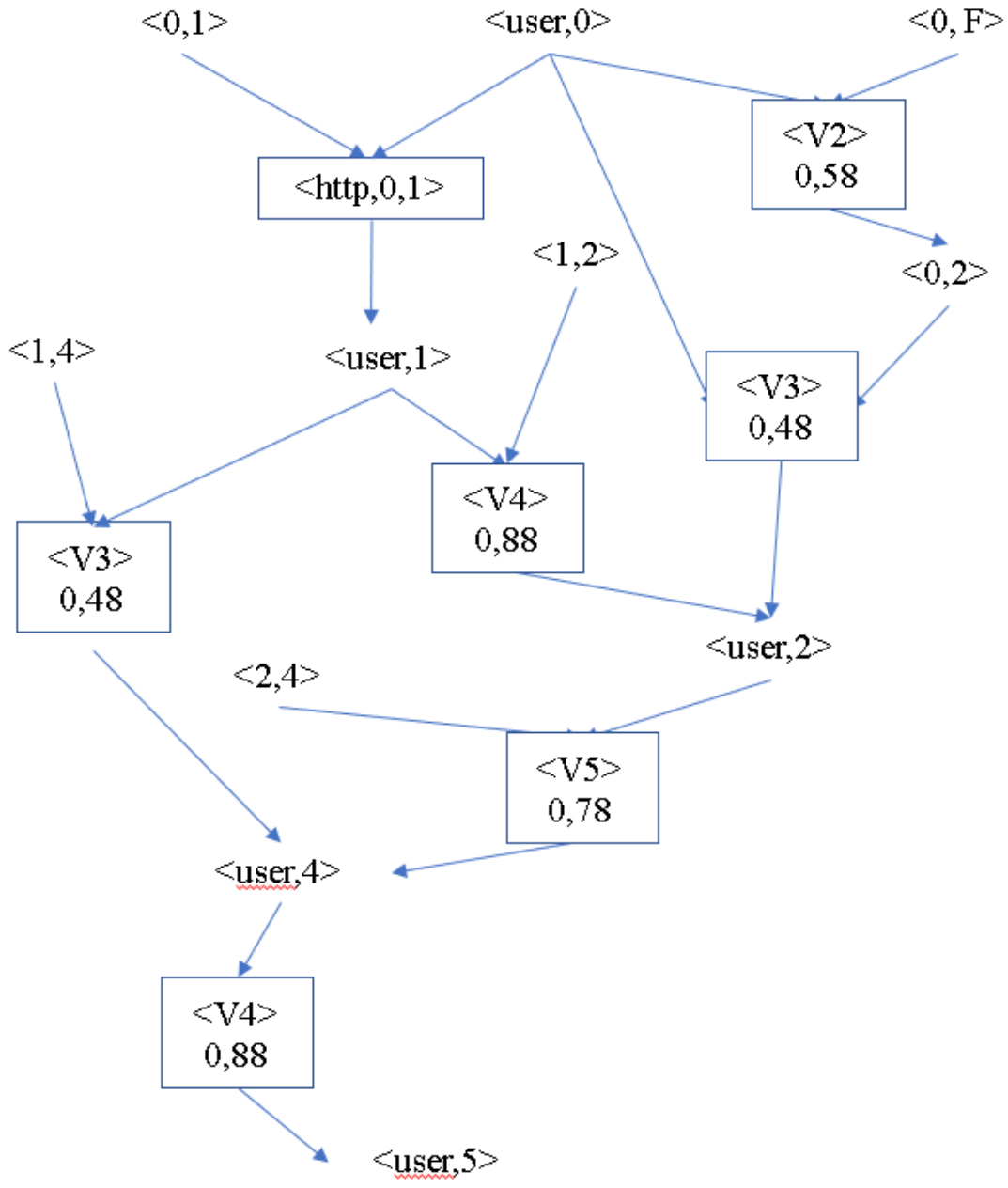


Рисунок 3.4 – Приклад застосування евристичних алгоритмів Mpath-Topo та Keynode

Приклад 3.6. Тут вибираємо $r_0 = 0,08$ і $r_1 = 1$. На рисунку 3.4 спочатку обчислимо $r = 5,12*10^{-4}$. Потім знову обчислюємо r, призначаючи r_1 кожному ресурсу. Наприклад, якщо V1 змінено з r_0 на r_1, маємо $r = 0,0064$, тож V1 є ключовим

вузлом. Аналогічно можемо отримати послідовність ключових вузлів як KN = {V1,V4,V3,V5}. Якщо бюджет K = 2, то будуть обрані V1 і V4 і результат r = 0,51 з частотою помилок 0,04.

Якщо застосувати всі евристичні алгоритми до рисунку 3.4, зможемо отримати коефіцієнт помилок з відповідними алгоритмами: Вибір частоти (0,33), Топологічний порядок (0,27), Зворотний топологічний Одер (0,33), Найкоротший шлях (0,29), Mpath-Topo (0,04), Keynode (0,04). Очевидно, що алгоритми mpath-topo і keynode значно точніші, ніж інші алгоритми.

3.5 Висновок

В даному розділі представлено удосконалений метод побудови резильєнтних IT-інфраструктур в умовах здійснення атак мережного типу.

Метод ґрунтується на розрахунках ймовірності мережевої атаки на IT-інфраструктуру, ймовірності атаки на основі CVSS, імовірності атаки на основі графа, і, на відміну від відомих , застосовує агрегацію ймовірностей атак всередині мережі.

Також мето залучає евристичні алгоритми для обчислення поверхні мережевої атаки на IT-інфраструктуру, зокрема застосування евристичних алгоритмів випадкового вибору та вибору частоти.

4 ПРОГРАМНО-АПАРАТНИЙ ЗАСОБИ ЗАБЕЗПЕЧЕННЯ ПОБУДОВИ РЕЗИЛЬЄНТНИХ ІТ-ІНФРАСТРУКТУР В УМОВАХ ЗДІЙСНЕННЯ АТАК МЕРЕЖНОГО ТИПУ

4.1 Конфігурація комп'ютерної ІТ-інфраструктури, яка може зазнати атаки

З метою практичної апробації розробленого удосконаленого методу побудови резильєнтних ІТ-інфраструктур в умовах здійснення атак мережного типу було реалізовано відповідні програмно-апаратний засоби.

З цією метою було змодельовано комп'ютерну мережу ІТ-інфраструктури, що може зазнати атаки.

Незважаючи на відносно невеликий масштаб, конфігурація мережі імітує типову корпоративну мережу, наприклад, із DMZ, веб-сервером за брандмауером, доступним із загальнодоступного Інтернету, та приватною мережею керування, захищеною тим же брандмауером, але з глибокою перевіркою пакетів і оснащеною контролером домену. і сервер CITRIX, як показано на рисунку 4.1.

Розглянемо деталізований опис механізмів збору необхідної вхідної інформації для створення екземплярів кожної метрики, а зібрана інформація наведена в таблиці 4.1.

Метрика d_1. Інформація, яка збирається для створення екземпляра d_1, включає:

1. Хости: топологічна карта мережі чітко вказує, що це хости: Firewall, Web Server, $host_1$, Citrix і Domain Controller.

2. Ресурси: опис конфігурації мережі вказує, що брандмауер, який використовується в цій мережі, це Symantec Endpoint Protection, який розгортає два різних правила: для мережі DMZ з фільтрованим вихідним трафіком і для мережі керування з глибокою перевіркою вмісту. Використовуємо nmap для сканування внутрішньої мережі, щоб збирати інформацію про відкриття портів, програми, що працюють на хостах, інформацію про операційні системи на хостах тощо.

Наприклад, визначено, що загальнодоступний веб-сервер має відкриті порти 80 і 43, із SQLite та Appache, які працюють поверх OpenSuSE.

3. Подібність між ресурсами: використовуємо спрощений підхід, розглядаючи ресурси в цій мережі як ідентичні або різні, тому оцінка схожості дорівнює 1 або 0.

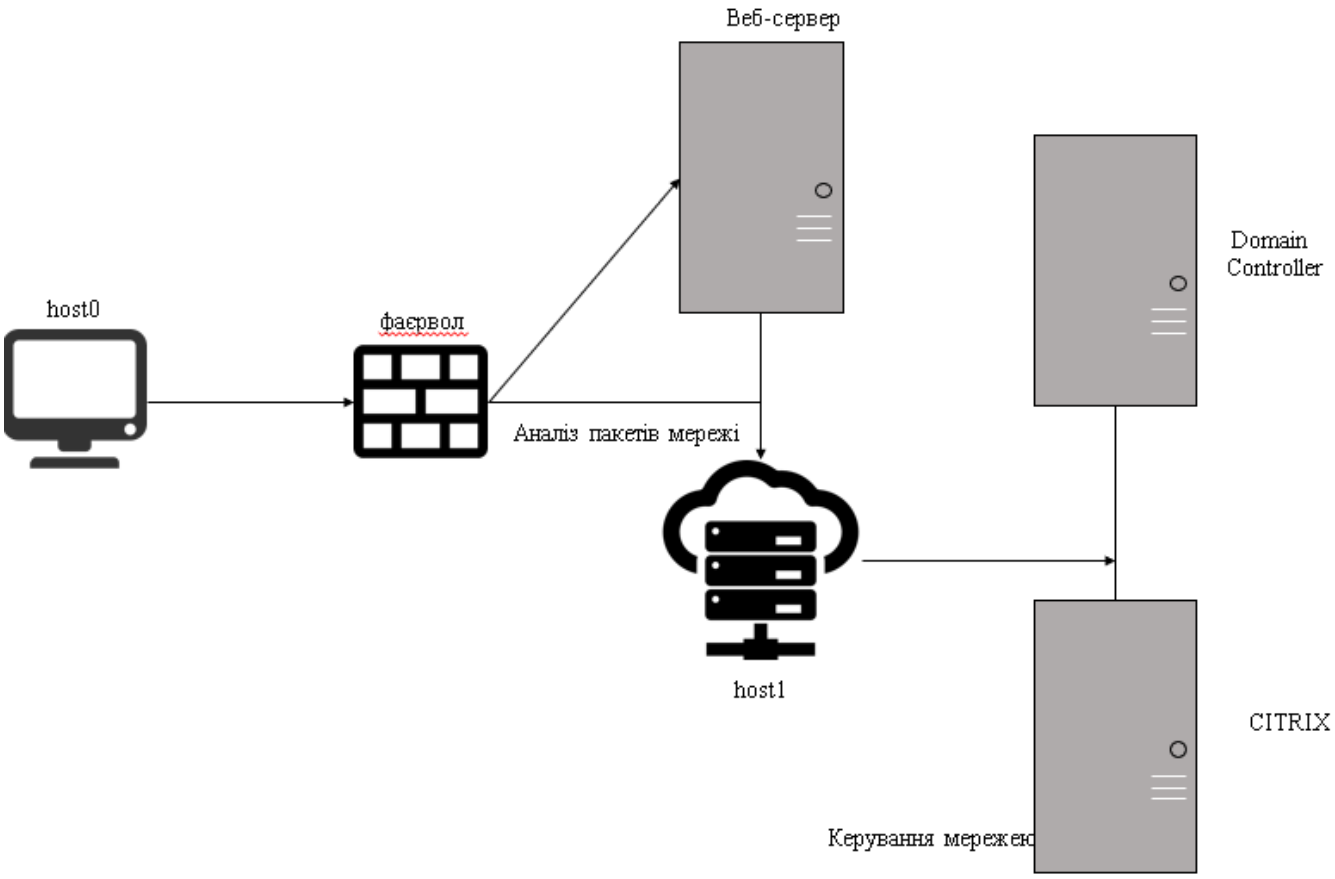


Рисунок 4.1 – Приклад архітектури досліджуваної комп’ютерної мережі IT-інфраструктури

Метрика d_2. Щоб створити екземпляр d_2, потрібно зібрати наступне, на додаток до того, що вже зібрано для d_1:

1. Підключення: топологічна карта мережі чітко забезпечує зв'язок між хостами.

2. Умови безпеки: вивчаємо програми та наявні в них уразливості, щоб зібрати відповідні умови безпеки, пов'язані з попередніми та наступними умовами

атаки. Наприклад, SQLiteManager працює з правами користувача на веб-сервері, що вказує на наступну умову привілеїв користувача на хості, тоді як OpenSuSE має права root як післяумову через потенційну вразливість підвищення привілеїв.

3. Критичні активи: у цій мережі вважаємо контролер домену критичним активом через його особливу роль (фактичні системні адміністратори будуть у кращому становищі для визначення своїх критичних активів).

Метрика d_3. Щоб створити екземпляр d_3, потрібно зібрати наступне, на додаток до того, що вже зібрано для d_2,

4. Граничні ймовірності спільних типів ресурсів та умовні ймовірності того, що ресурси можуть бути скомпрометовані, якщо всі передумови задоволені: призначаємо 0,08 як номінальне значення для обох ймовірностей, які, безсумнівно, можна уточнити, якщо адміністраторам буде доступна додаткова інформація.

Таблиця 4.1 – Зібрана інформація щодо IT-інфраструктури

Хости	Підключення	Порти	Ресурси	Умови безпеки
фаєрвол	Фаєрвол, Веб-сервер	-	Вихідний трафік відфільтровано. Правила глибокої перевірки вмісту	-
Веб-сервер	Фаєрвол, хост	80,43	Http Services, SQLite, OpenSuSE	користувач, root
хост	Фаєрвол, Веб-сервер, domain controller, citrix	80,3389	Передача файлів, RDP сервіс, Windows 10,	domain user, local administrator
Citrix	domain controller, хост	80,3389	Http сервіси, Citrix Xen App, RDP сервіс	користувач, local administrator
Domain Controller	citrix, хост	3389	RDP сервіс	користувач, domain administrator

4.2 Експериментальні дослідження програмно-апаратної реалізації методу

Розглянемо три запропоновані показники, застосовуючи їх до різних випадків використання за допомогою моделювання. Усі результати моделювання збираються за допомогою КС, оснащеної ЦП 3,0 ГГц і 16 ГБ оперативної пам'яті в середовищі Python під OpenSuSE.

Байєсівська мережна метрика реалізована за допомогою OpenBayes [33]. Щоб створити велику кількість графів ресурсів для моделювання, спочатку створюємо невелику кількість початкових графів на основі реальних мереж, а потім створюємо більші графи з цих початкових графів, вводячи нові хости та призначаючи ресурси випадковим, але реалістичним способом (наприклад, змінюємо кількість попередніх умов кожного вторгнення мережного типу в невеликому діапазоні, оскільки реальні атаки мережного типу зазвичай мають невелику кількість передумов).

В дослідженні було застосовано три показники різноманітності мережі до різних випадків використання для оцінювання трьох показників за допомогою числових результатів і вивчити ці результати разом зі статистично очікуваними результатами, представленими різними сценаріями атаки.

Перші два моделювання порівнюють результати всіх трьох показників, щоб вивчити їх різні тенденції в міру збільшення розмірів графів і збільшення різноманітності. Перш за все, щоб перетворити байєсівську мережну метрику d_3 у порівнянну шкалу двох інших, використовуємо $\frac{\log 0.08(r')}{\log 0.08(r)}$ (тобто коефіцієнт, заснований на еквівалентній кількості атак мережного типу нульового дня) замість d_3. У лівій частині малюнка 8 точки розсіювання, позначені червоним кольором X, є окремими значеннями d_2. Сині точки, позначені Y, є значеннями d_3 (перетворені, як зазначено вище). Також показано їх середні значення та середнє значення ефективної метрики на основі багатства d_1. Правий малюнок показує середнє значення трьох показників у збільшенні кількості окремих ресурсів для графів ресурсів фіксованого розміру.

Результати: обидві симуляції показують, що, хоча всі три показники мають подібну тенденцію (на рисунку ліворуч, різноманітність зменшиться на більших графах, оскільки буде більше дубльованих ресурсів) і фіксують той самий ефект збільшення різноманітності (на рисунку праворуч), байєсівська мережна метрика d_3 якимось чином відображає проміжний результат між двома іншими крайнощами (d_1 можна розглядати як середнє значення для всіх ресурсів, тоді як d_2 залежить лише від найкоротшого шляху). Ці результати показують, що застосування всіх трьох показників може дати послідовні результати і мотивує порівнювати їх за допомогою подальшого моделювання.

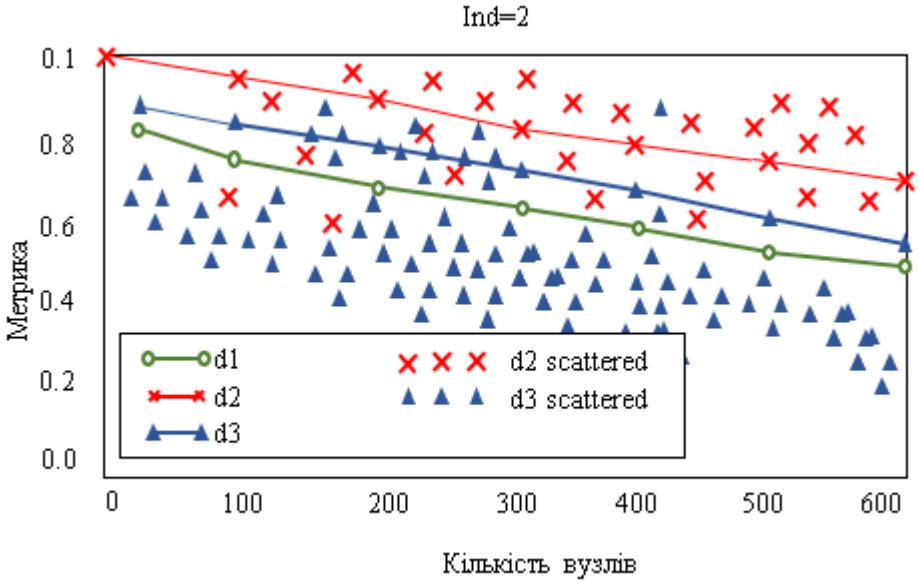
Перший варіант використання розглядає атак мережного типу, що характеризуються таким чином. По-перше, кожен атака мережного типу може використовувати лише невелику кількість вразливостей. Під час реалізації випадковим чином вибираємо від одного до трьох типів ресурсів як можливості кожного атаки мережного типу. По-друге, мета атаки мережного типу — інфікувати якомога більше хазяїв, не потребує конкретних цілей. Хоча деякі атаки мережного типу або боти дійсно можуть мати ціль, їм, як правило, все одно необхідно спочатку скомпрометувати велику кількість машин, перш ніж ціль буде досягнута (наприклад, Stuxnet [29]).

На рисунку 4.3 вісь X – це відношення кількості типів ресурсів до кількості екземплярів ресурсів, що приблизно відображає рівень різноманітності з точки зору багатства (можна помітити, що d_1 близький до прямої лінії). Вісь Y показує результати трьох показників, а також співвідношення хостів, які не заражені імітованими атаками мережного типу. Чотири рядки представляють три показники (позначені d_1, d_2 і d_3) і відношення хостів, не заражених імітованими атаками мережного типу (позначені S_1).

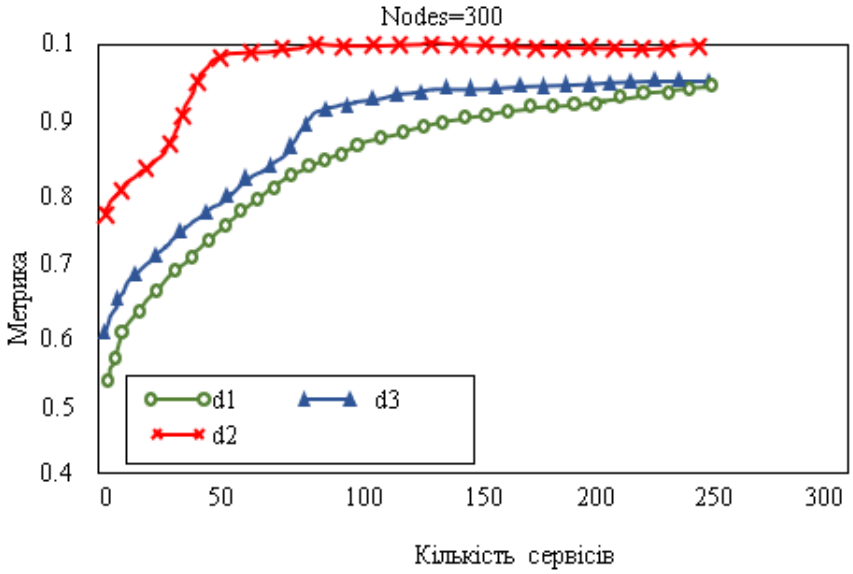
Рисунок 4.2 (a) і (b) відповідають різному відсотку атак мережного типу першого рівня (атак мережного типу, які мають лише початкові умови як передумови) серед усіх атак мережного типу, що приблизно показує, наскільки добре захищена мережа (наприклад, 50% означає більш вразливу мережу, ніж 10%,

оскільки спочатку зловмисники можуть досягти половини або в п’ять разів більше атак мережного типу). Для кожної конфігурації повторюємо 500 разів, щоб отримати середній результат моделювання атак мережного типу.

Результати: При моделюванні можемо зробити наступні спостереження. По-перше, усі три показники все ще демонструють подібні тенденції та зв’язки, які обговорювалося вище.



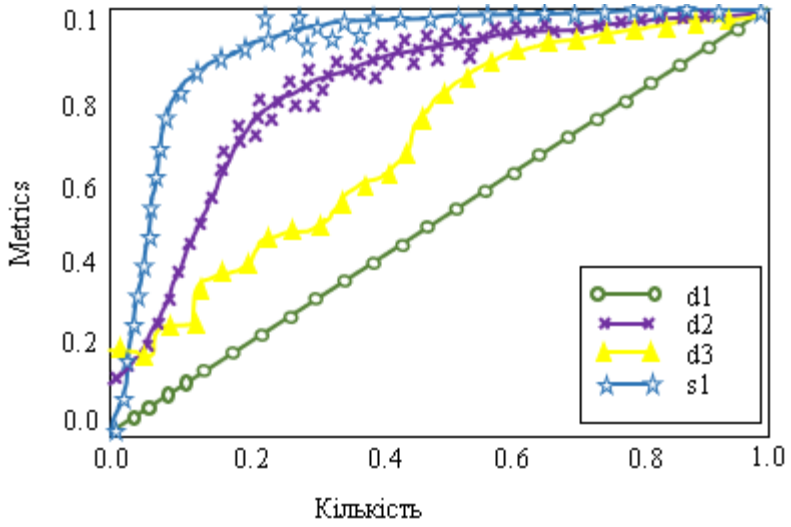
а)



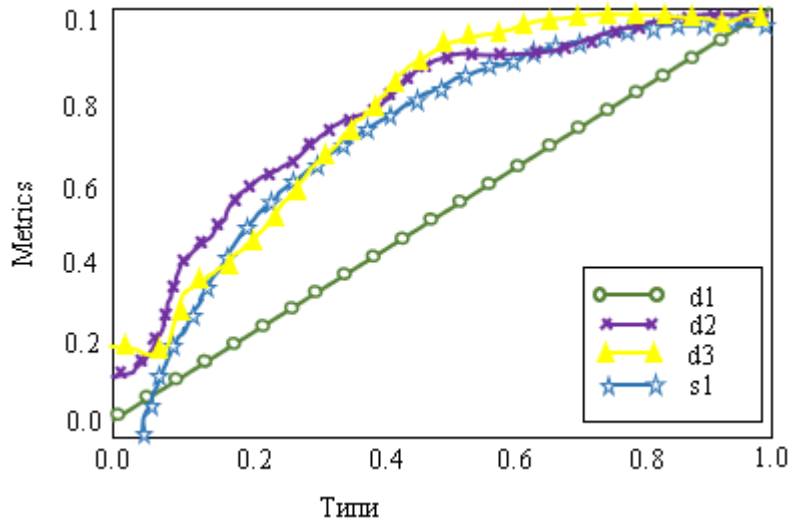
б)

Рисунок 4.2 – Порівняння показників (а) та ефекту збільшення різноманітності (б)

Рисунок 4.3 (а) показує, що, коли мережа краще захищена (з початково доступними лише 10% атак мережного типу), ефект збільшення різноманітності на змодельованих атак мережного типу показує більш тісний зв'язок з показником d_2, ніж дві інші, обидва з яких показують, що збільшення різноманітності може значно збільшити відсоток неінфікованих атаками мережного типу хостів. Для порівняння, малюнок 9 (б) показує менш багатообіцяючий результат, де як три показники, так і відсоток неінфікованих хостів мають тенденцію до подібної тенденції.



а)



б)

Рисунок 4.3: Поширення атаки мережного типу (10% початково успішних атак мережного типу (а), 50% початково успішних атак мережного типу (б))

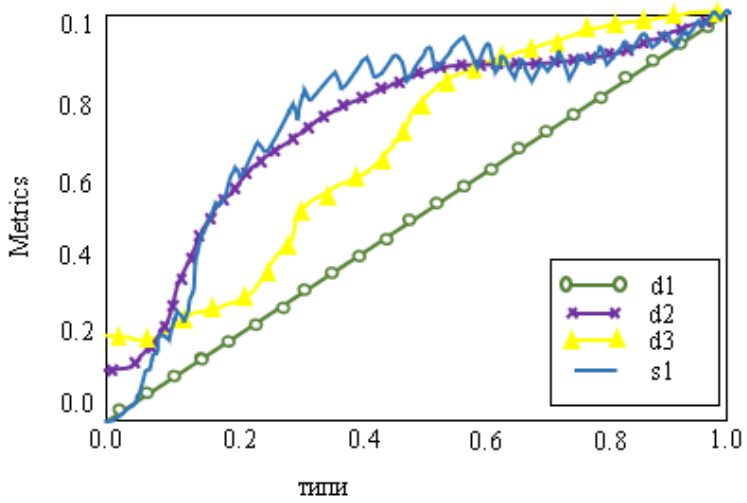
В добре захищених мережах багато хостів не можуть бути досягнуті, поки атаки мережного типу не інфікують інші сусідні хости, тому збільшення різноманітності може ефективніше пом’якшити поширення атак мережного типу. У менш захищених мережах, де половина атак мережного типу може бути досягнута спочатку, вплив різноманітності на атак мережного типу майже пропорційний багатству ресурсів (d_1), і всі три показники, як правило, дають подібні результати.

Другий варіант використання – цілеспрямовані атаки. Моделюємо зловмисників з різними можливостями відповідає гамма-розподілу [72]. Аналогічно, також повторюємо кожен експеримент 500 разів і досліджуємо два різних випадки, що відповідають різному відсотку атак мережного типу першого рівня. На рисунку 4.4, S_2 є результатом імітованого нападника, що означає відсоток нападників, які не можуть досягти випадково вибраної мети.

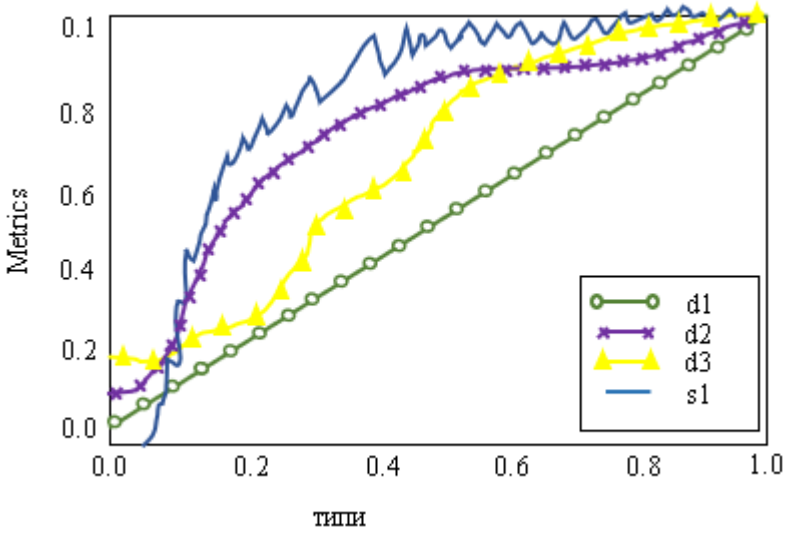
З результатів можна спостерігати результати, що й у змодельованих атак мережного типу. Зокрема, збільшення різноманітності може більш ефективно пом’якшити шкоду, завдану імітованими зловмисниками для добре захищених мереж (ліворуч), ніж для менш захищених мереж (малюнок 10 (a)). Крім того, на лівому рисунку результати змодельованих зловмисників ближчі до результатів d_2, ніж за двома іншими показниками, тоді як на рисунку 4.4 (b) ближчі як до d_2, так і до d_3. Крім того, порівнявши результати на рисунку 10 (цільова атака) з результатами на рисунку 4.3 (атака мережного типу), можна побачити, що той самий рівень різноманітності може ефективніше пом’якшити вплив атаки мережного типу, ніж для імітованих зловмисників. Це можна пояснити тим, що атака мережного типу має набагато менші можливості (набір ресурсів, які він може скомпрометувати), ніж імітований зловмисник.

Наступний набір симуляцій досліджує зв’язки між d_2 і d_3 більш детально. Обидва ці два показники враховують причинно-наслідкові зв’язки між ресурсами, але зосереджені на дещо інших перспективах (найменші та середні зусилля, необхідні для компрометації мережі). У той же час ці два показники тісно пов’язані. У попередніх моделюваннях вже бачимо, що значення d_3 майже завжди менші за d_2.

Ці дві метрики також можуть сходитися до подібних значень у певних випадках (наприклад, у крайньому випадку лише одного шляху на графі ресурсів, d_2 і d_3 дадуть ідентичне значення). Щоб побачити, як ці дві метрики пов'язані один з одним, моделюємо графи ресурсів різних розмірів і форм, а середні результати (d_3/d_2) 500 симуляцій показано на рисунку 4.5.



а)

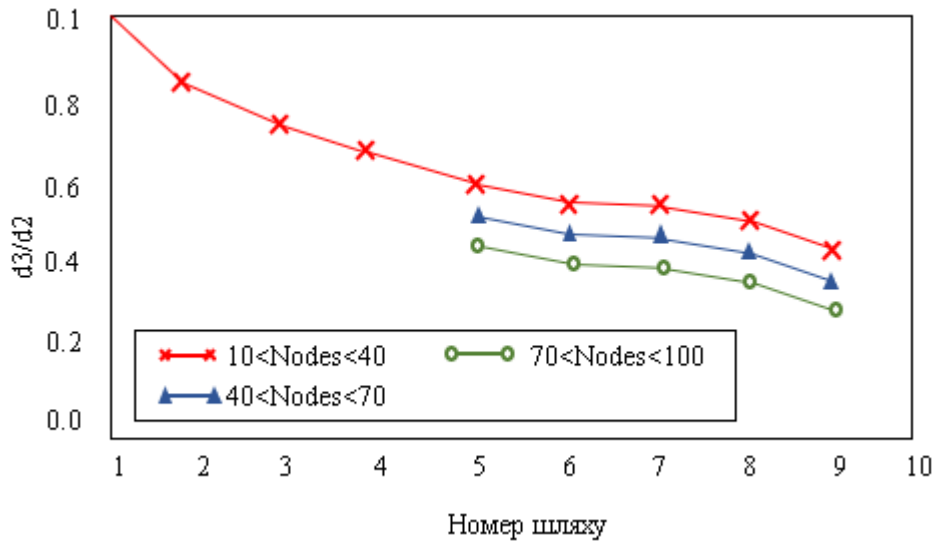


б)

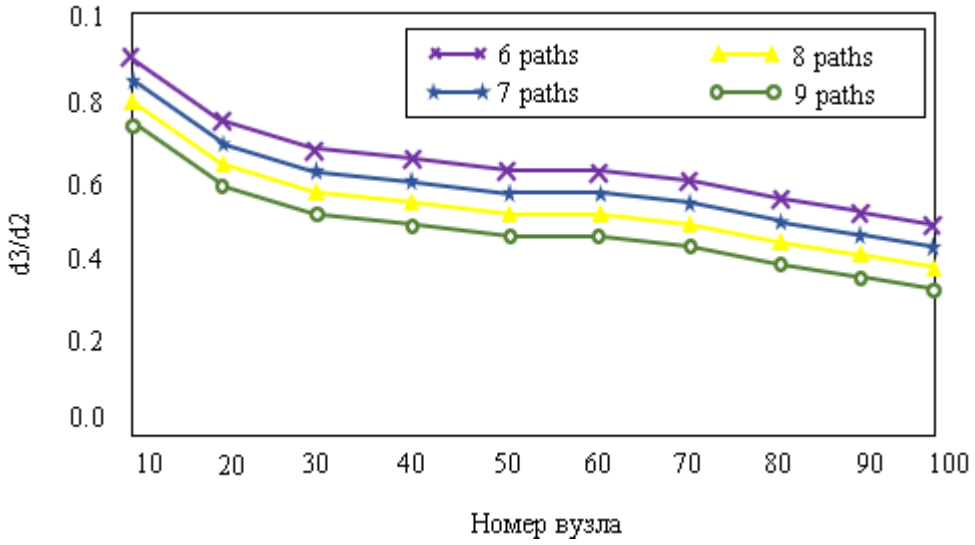
Рисунок 4.4: Цільова атака (0% початково успішних вразливостей (а), 50% початково успішних вразливостей (б))

Результати: рисунок 4.5 (а) показує, що за фіксованих розмірів графів

ресурсів різниця між d_2 і d_3 збільшується (відношення 1 означає, що показники мають ідентичні значення) з кількістю шляхів у графах ресурсів. Це очікується, оскільки d_2 представляє лише різноманіття з точки зору одного конкретного шляху, тоді як d_3 враховує всі шляхи, тому зі збільшенням кількості шляхів співвідношення стає меншим (означаючи більші відмінності). На рисунку 4.5 (b) показано, що відмінності збільшуються (відношення зменшується) зі збільшенням розмірів графа (усі графи ресурсів мають від 6 до 9 шляхів).



а)



б)

Рисунок 4.5 – d_3/d_2 у кількості шляхів (a) і вузлів (b)

Це можна пояснити тим, що різниця між різними шляхами буде ставати більш

значною зі збільшенням розміру графів і фіксованою кількістю шляхів, а отже, різниця між двома метриками повільно збільшується (відношення зменшується). Обидва моделювання означають, що, хоча вибір метрики переважно залежить від бажаної перспективи, обидві метрики слід враховувати особливо для великих і менш захищених мереж (це означає, що існує більше шляхів атаки), оскільки їх значення можуть значно відрізнятися.

Зараз вивчаємо третій варіант використання, Moving Target Defense (MTD). Підхід MTD намагається досягти кращої безпеки, змінюючи в часі конфігурації мереж, у яких різноманітність відіграє вирішальну роль. Мета тут полягає в тому, щоб вивчити вплив різної різноманітності на ефективність MTD, а також на оцінку метрики при застосуванні до MTD. Наскільки відомо, це одна з перших зусиль щодо вивчення MTD за допомогою моделювання (інша подібна робота Чжуанга та ін. [141] також використовує результати моделювання, але мета полягає в тому, щоб оцінити пропонований показник різноманітності мережі в додатках MTD, що відрізняється від їх вивчення).

Моделювання базується на наступних припущеннях:

1. Припускаємо, що зловмисники спочатку не мають повної інформації про мережу, але можуть поступово дізнаватися про такі деталі, коли конфігурація змінюється з часом. Зокрема, зловмисники можуть дізнатися про зміну конфігурації (наприклад, через невдачу своїх атак, або спостерігаючи особливі особливості конфігурації).

2. Припускаємо, що можливості зловмисників, тобто набори ресурсів слідують гамма-розподілу. І кожен ресурс має вікно атаки, лише ті, що досягли тривалості вікна атаки, можуть бути скомпрометовані зловмисниками, які мають відповідні можливості. Більше того, наявність можливостей не означає, що зловмисник може негайно скомпрометувати ресурс, оскільки йому все одно може знадобитися певний час, щоб фактично здійснити атаку на певні екземпляри ресурсу. Тому в моделюванні призначаємо кожному ресурсу вікно атаки, і лише той

ресурс, тривалість якого в конфігурації є довшою, ніж відповідне вікно атаки, може бути скомпрометовано зловмисниками, які мають такі можливості.

3. Припускаємо, що підхід MTD використовує конфігурації мережі, що динамічно змінюються. Але використовуємо фіксовану частоту змін у моделюванні. Зокрема, топологія мережі залишається незмінною, але ресурси кожного хоста можуть змінюватися. Крім того, припускаємо фіксовану частоту змін у моделюванні (наприклад, на рисунку 12, лівий малюнок показує, що частота зміни конфігурації становить 1 конфігурацію на день). Залишаємо інші способи зміни конфігурації мережі, наприклад, використання різної частоти, як майбутню роботу.

На рисунку 4.6 (a) показано середній рівень успіху зловмисників після 40 днів впливу мережі за кількість днів до зміни конфігурації (наприклад, 5 днів означає, що конфігурація змінюється одна на п’ять днів). На рисунку 4.6 (b) показано коефіцієнт успіху атаки (ліва вісь Y представляє рівень успіху атак мережного типу, а права – цілеспрямованих атак) у кількості днів з моменту виявлення мережі, з частотою змін, установленою в одній конфігурації. зміна за день.

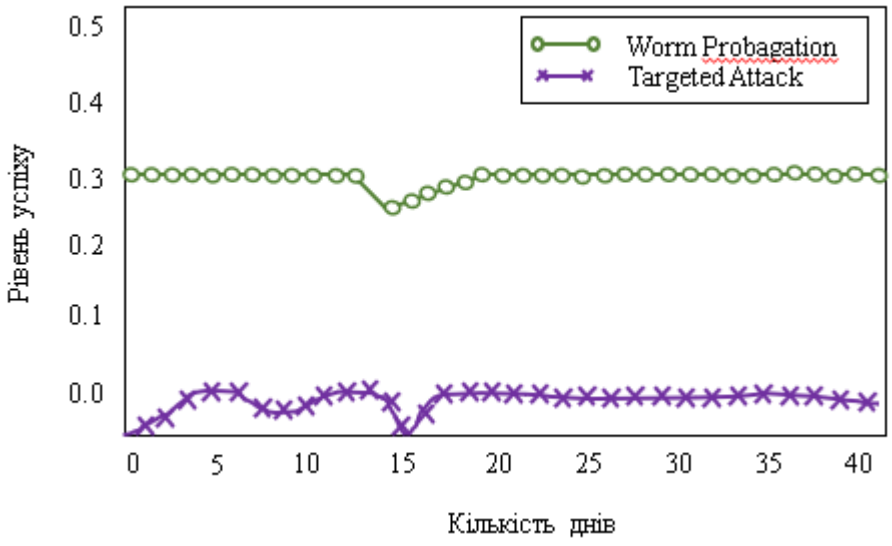
Результати: на рисунку 4.6 (a) бачимо, що більш часта зміна конфігурації мережі не обов’язково означає кращу безпеку (нижчий коефіцієнт успіху), оскільки надто швидкі чи надто повільні зміни можуть збільшити ризик ресурс і, отже, збільшує шанси зловмисника скомпрометувати цей ресурс.

Фактично, видно чітку тенденцію щодо успішності, оскільки кількість днів для зміни збільшується. Невеликі падіння в обох рядках вказують на те, що найнижчі показники успіху співпадають із середнім розміром вікон атаки (розглянувши випадок припускаємо, що для зкомпрометації ресурсу потрібно 10 або 15 днів), що насправді може не мати значення, оскільки різні ресурси можуть мати різні вікна атаки.

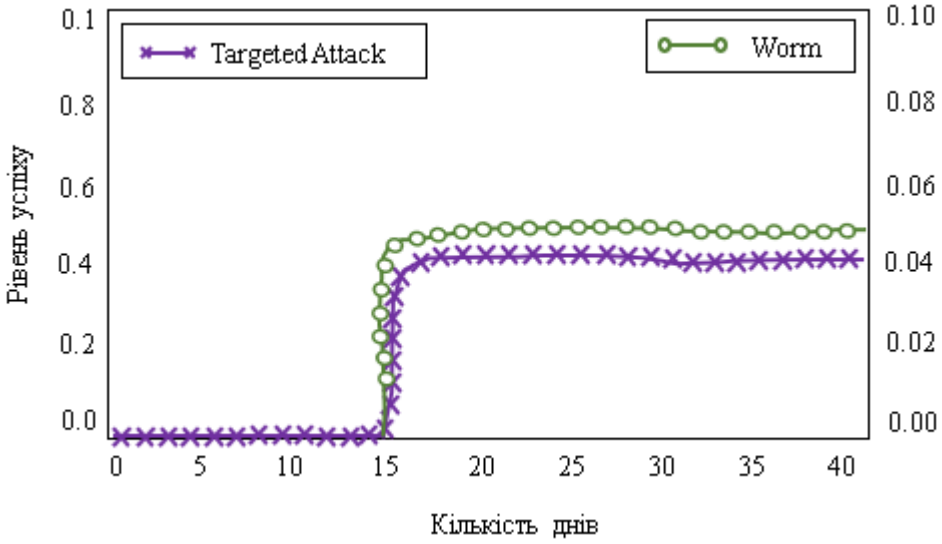
На рисунку 4.6 (b) бачимо, що до 15 днів існує нульовий рівень успіху як для поширення атаки мережного типу, так і для цільової атаки, через припущені вікна атаки від 10 до 15 днів.

Через 15 днів відбувається стрибок в обох рядках, а це означає, що завдяки

накопиченим зусиллям і атаки мережного типу, і зловмисники зможуть скомпрометувати все більше і більше ресурсів, і показники успіху майже не змінюються приблизно через 20 днів, оскільки тепер залежать більше про здатність атак мережного типу (зловмисників).



а)



б)

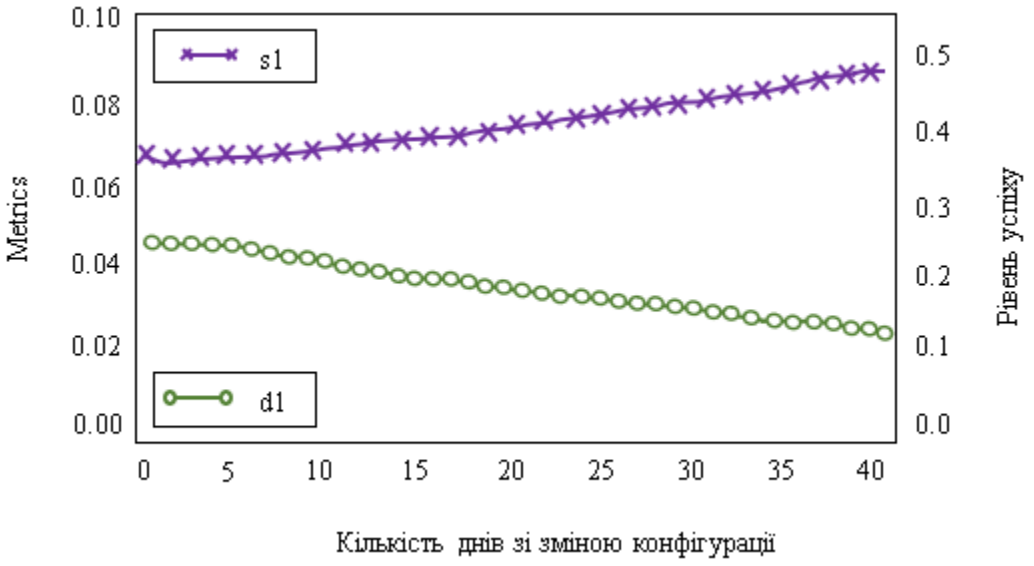
Рисунок 4.6 – Швидкість успішних атак у частоті змін (a) і в часі (b)

На рисунку 4.7 застосовуємо лише метрику d_1 до MTD, оскільки всі три показники відображатимуть подібні тенденції, як описано вище. На рисунку 4.7 (a)

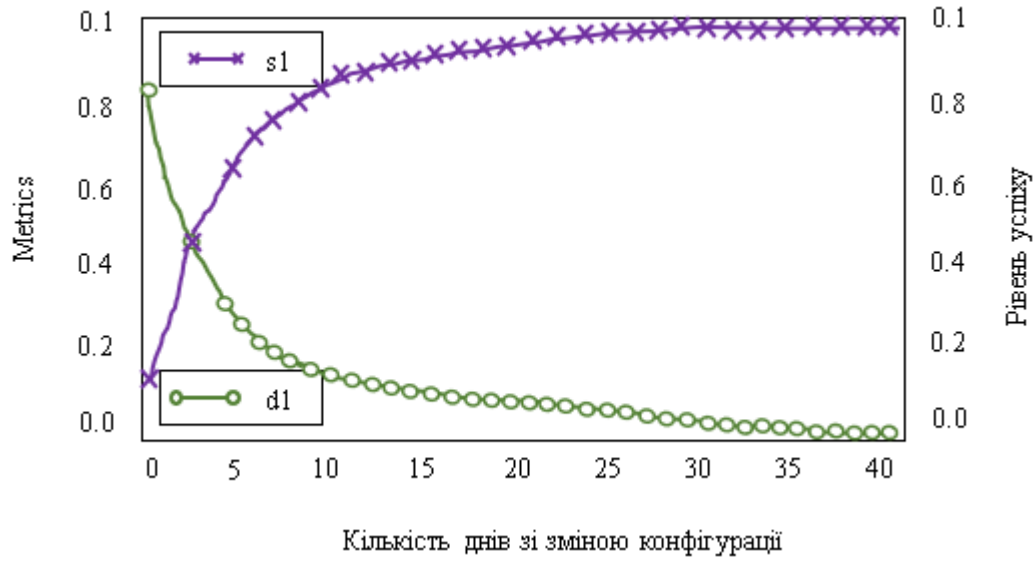
бачимо, що якщо набір типів ресурсів залишається відносно невеликим (наприклад, $\frac{\#\text{ ресурсів}}{\#\text{днів}} < 20\%$), то метрика d_1 залишається майже рівною, коли частота конфігурації зміни відносно високі, а потім різко падають, коли частота нижча (приблизно одна зміна на 20 днів). Це вказує на те, що при обмеженій кількості типів ресурсів вища частота змін конфігурації не забезпечує значного підвищення безпеки, якщо тільки частота не надто низька. Рисунок 4.7 (a) також показує, що коли кількість днів на зміну збільшується понад 20, різноманітність падає, а рівень успіху зростає, що означає, що метрика різноманітності ефективно фіксує ефективність MTD. На рисунку 4.7 (b) показані подібні результати для більшого набору ресурсів ($\frac{\#\text{ ресурсів}}{\#\text{днів}} > 80\%$).

Успішний курс і d_1 показує відповідні (зворотні) тенденції. Однак тепер з достатньо великим набором ресурсів на вибір, менш часті зміни конфігурацій означають меншу різноманітність і, отже, меншу безпеку.

В останньому моделюванні вивчаємо взаємозв’язок між вартістю та безпекою в MTD, як показано на рисунку 4.8. Для поширення атаки мережного типу призначаємо грошові значення всім хостам, причому критичні активи (умови цілі) мають вищі значення, тоді як для цільової атаки, призначаємо значення лише критичним активам. Червона та зелена пунктирні лінії у верхній частині малюнка показують загальне значення для кожного сценарію. Кожного разу, коли атаки мережного типу або зловмисники компрометують ресурс, його призначене значення вважається втраченим. Така втрачена вартість є першою частиною загальної вартості. Інша частина – це витрати на зміну конфігурацій у MTD (наприклад, адміністративні витрати на придбання нового програмного забезпечення або витрати на продуктивність затримки запиту клієнта).



а)



б)

Рисунок 4.7 – d1 в частоті змін, у частині «Менше типів ресурсів» (ліворуч) і «Більше типів джерел ресурсів» (праворуч)

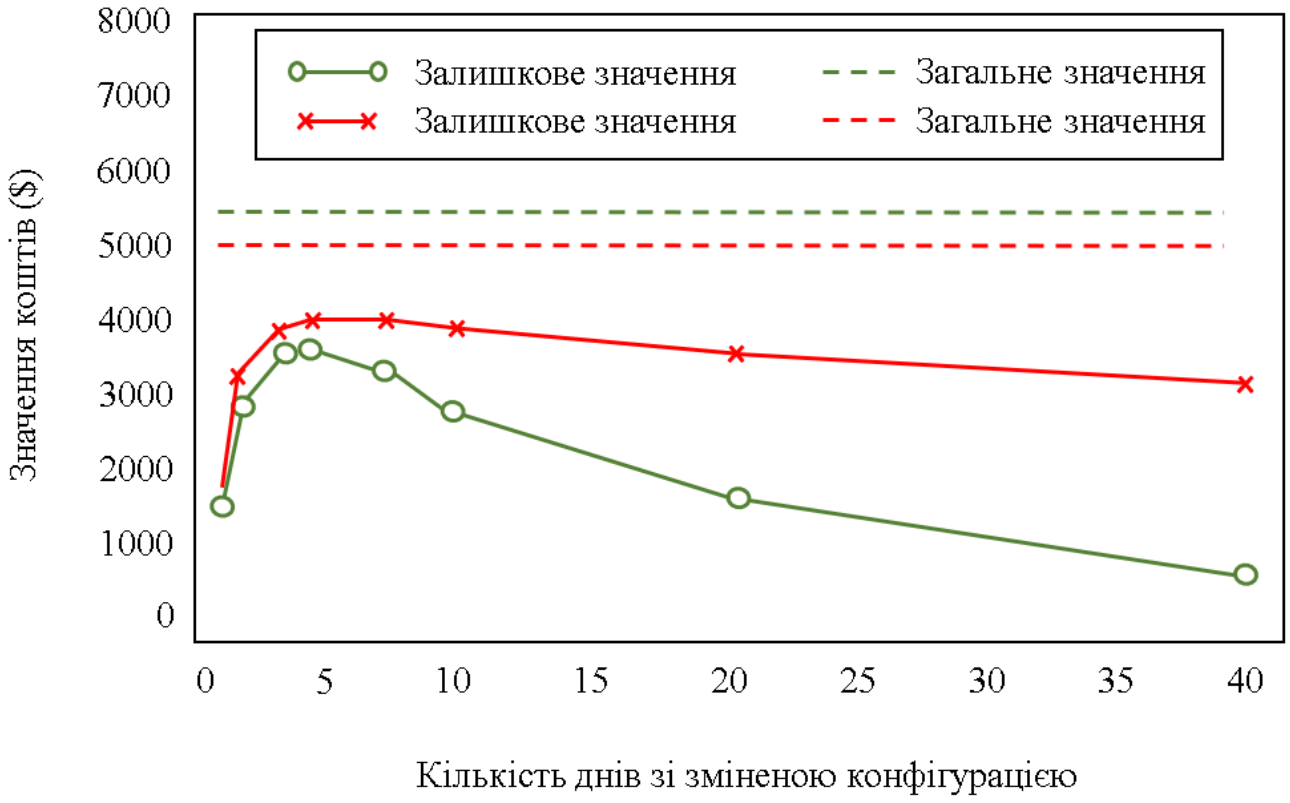


Рисунок 4.8 – Загальна вартість у частоті змін

На рисунку 4.8 зображена загальна вартість (втрачена вартість через скомпрометовані ресурси плюс вартість змін конфігурації) у кількості днів для зміни конфігурації. Результати показують, що загальна вартість спочатку зменшиться, а потім зросте. Оптимальним налаштуванням частоти є приблизно одна зміна конфігурації на 5 днів, що найкраще врівноважує вартість зміни нової конфігурації з втраченими значеннями скомпрометованих ресурсів. Зауважте, що, згідно з обговореннями вище, оптимальна частота також буде залежати від кількості доступних ресурсів.

4.3 Аналіз отриманих результатів практичного застосування рішення

Подібність між ресурсами є важливою інформацією для метричних моделей. Хоча такий підхід може бути прийнятним у багатьох випадках, його, безумовно, потрібно уточнити, враховуючи той факт, що незначні відмінності зазвичай існують між різними версіями одного ПЗ ІТ-інфраструктур, тоді як різне ПЗ мати загальний код або бібліотеки. Вимірювання таких відмінностей або подібності між ресурсами ІТ-інфраструктур призведе до більш точних результатів для показників різноманітності.

4.3.1 Апробація запропонованих рішень

Щоб продемонструвати необхідність вивчення подібності ПЗ ІТ-інфраструктур, було проведено тематичне дослідження різних версій Chrome і показуємо, як такі відмінності, якщо врахувати, можуть вплинути на результати метрики різноманітності. Зокрема, досліджуємо 7 послідовних версій браузера Chrome. Версії позначені номерами індексів від 0 до 10, від останньої до найстарішої, на рисунку 15. Використовуємо останню версію 0 як базову для порівняння.

Можна очікувати, що між цими версіями з такими невеликими відмінностями в номерах версій і часу публікації не буде великої різниці. Однак дослідження показує, що насправді це не так. Взявши для прикладу версії 0 і 1, хоча загальна кількість вихідних файлів досить подібна (75136 і 73596 відповідно), є відмінності в 9338 файлах (близько 12%) між двома версіями. Крім того, такі відмінності включають 767 987 вставок і 190 943 видалення, що становить близько 13% рядків (загальна кількість рядків у цих двох версіях становить 14 750 264 і 15 330 677 відповідно). Ці цифри чітко вказують на значну різницю між двома версіями, незважаючи на те, що час їх публікації розривається лише в кілька хвилин.

На рисунку А.1 Додатку А два рядки показують кількість відмінностей у

файлах і модифікаціях відповідно для десяти версій. Очевидно, що існує подібна тенденція на цих двох різних рівнях відмінностей, з меншими відмінностями між версіями в одній гілці (наприклад, 42) і більш значними відмінностями між різними гілками (числа можуть досягати майже 10 000 на рівні файлу і 1 000 000 на модифікації рівень). Хоча номери версій можуть надавати корисну інформацію в конкретному випадку, така інформація про зв’язки між кількома версіями (наприклад, гілки розробки) не завжди може бути надана постачальником ПЗ IT-інфраструктур.

4.4 Висновки

В розділі представлено практичну реалізацію програмно-апаратних засобів удосконаленого методу побудови резильєнтних IT-інфраструктур в умовах здійснення атак мережного типу. Також в розділі подано конфігурацію мережі IT-інфраструктур з тестуванням на проникнення, здійснено експериментальні дослідження програмно-апаратної реалізації методу, зокрема, аналіз отриманих результатів практичного застосування рішення. В результаті отримано висновки, що застосування методу забезпечує резильєнтне функціонування IT-інфраструктур в умовах здійснення атак мережного типу.

ВИСНОВКИ

В першому розділі розглянуто поняття резильєнтного функціонування та оцінка резильєнтності іт-інфраструктур в умовах здійснення атак мережного типу, досліджено відомі методи оцінки резильєнтності мережі в умовах атак. Проаналізовано та досліджено показники резильєнтності мережі в умовах здійснення атак, зокрема показник різноманітності мережі, метрику покриття атаки мережі. В розділі досліджено моделі вразливостей, показники безпеки, показник різноманітності мережі, метрику поверхні мережевих атак, а також модель виявлення вразливостей. В розділі зроблено висновки щодо необхідності розроблення удосконалений метод побудови резильєнтних ІТ-інфраструктур в умовах здійснення атак мережного типу.

В другому розділі здійснено моделювання показників безпеки для оцінки резильєнтності іт-інфраструктур в умовах здійснення атак мережного типу. Зокрема, виконано формальне моделювання різноманітності мережі, подано особливості застосування моделі. В розділі розглянуто показник мережевого різноманіття, натхненного біорізноманіттям та показник різноманітності мережі на основі найменших зусиль. В розділі також представлено модель різноманітність на основі найменших зусиль для атаки. Розглянута імовірнісна різноманітність мережі та імовірнісна модель мережевого різноманіття. Надано рекомендації щодо створення моделей мережевого різноманіття.

В третьому розділі представлено удосконалений метод побудови резильєнтних ІТ-інфраструктур в умовах здійснення атак мережного типу. Метод ґрунтується на розрахунках ймовірності мережевої атаки на ІТ-інфраструктуру, ймовірності атаки на основі CVSS, імовірності атаки на основі графа, і, на відміну від відомих , застосовує агрегацію ймовірностей атак всередині мережі. Також метод залучає евристичні алгоритми для обчислення поверхні мережевої атаки на ІТ-інфраструктуру, зокрема застосування евристичних алгоритмів випадкового вибору та вибору частоти.

В четвертому розділі представлено практичну реалізацію програмно-апаратних засобів удосконаленого методу побудови резильєнтних ІТ-інфраструктур в умовах здійснення атак мережного типу. Також в розділі подано конфігурацію мережі ІТ-інфраструктур з тестуванням на проникнення, здійснено експериментальні дослідження програмно-апаратної реалізації методу, зокрема, аналіз отриманих результатів практичного застосування рішення. В результаті отримано висновки, що застосування методу забезпечує резильєнтне функціонування ІТ-інфраструктур в умовах здійснення атак мережного типу.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ

1. Заграй А.О., Лисенко С.М. Метод побудови резильєнтних ІТ-інфраструктур в умовах здійснення атак мережного типу. *Збірник наукових праць XIII Всеукраїнської науково-практичної конференції «Актуальні проблеми комп'ютерних наук АП*КН-2021». Хмельницький, 2021. с. 15-16.

2. Лисенко С. М., Харченко В.С., Бобровнікова К.Ю., Щука Р. Резильєнтність комп'ютерних систем в умовах кіберзагроз: Онтологія та таксономії. *Радіоелектронні і комп'ютерні системи*. 2020. №1. С. 17-28.

3. S. Hopkins, E. Kalaimannan and C. S. John, Foundations for Research in Cyber-Physical System Cyber Resilience using State Estimation, *2020 SoutheastCon*, 2020, pp. 1-2, doi: 10.1109/SoutheastCon44009.2020. 9249745.

4. N. Jacobs, S. Hossain-McKenzie and E. Vugrin, Measurement and Analysis of Cyber Resilience for Control Systems: An Illustrative Example, *2018 Resilience Week (RWS)*, 2018, pp. 38-46, doi: 10.1109/RWEEK.2018.8473549.

5. C. Onwubiko, Focusing on the Recovery Aspects of Cyber Resilience, 2020 International Conference on Cyber Situational Awareness, *Data Analytics and Assessment (CyberSA)*, 2020, pp. 1-13.

6. S. Hopkins, E. Kalaimannan and C. S. John, Cyber Resilience using State Estimation Updates Based on Cyber Attack Matrix Classification, *2020 IEEE Kansas Power and Energy Conference (KPEC)*, 2020, pp. 1-6, doi: 10.1109/KPEC47870.2020.9167652.

7. Лисенко С. М. Моделі кібератак мережного та хостового типу. *Вимірювальна та обчислювальна техніка в технологічних процесах*. 2019. №2. С. 58-63.

8. Лисенко С. М. Метод забезпечення резильєнтності комп'ютерних систем в умовах кібер-загроз на основі самоадаптивності. *Радіоелектронні і комп'ютерні системи.* 2019. №4. С. 4–16.

9. A. Warzyński and G. Kołaczek, Intrusion detection systems vulnerability on adversarial examples, *2018 Innovations in Intelligent Systems and Applications (INISTA)*, 2018, pp. 1-4, doi: 10.1109/INISTA.2018.8466271.

10. Z. S. Malek, B. Trivedi, A. Shah, User behavior Pattern -Signature based Intrusion Detection, *2020 Fourth World Conference on Smart Trends in Systems, Security and Sustainability (WorldS4)*, 2020, pp. 549-552.

11. Lysenko, S., Bobrovnikova, K., Savenko, O., & Shchuka, R. Technique for Cyberattacks Detection Based on DNS Traffic Analysis, *CEUR-WS*, Vol 2732. ISSN: 1613–0073. 2020. pp. 171-182

12. Bobrovnikova, Kira, Sergii Lysenko, Piotr Gaj, Dmytro Denysiuk "Technique for IoT Cyberattacks Detection Based on DNS Traffic Analysis.*CEUR-WS.*Vol.2623. ISSN: 1613–00732020, pp. 208-218.

13. Sergii Lysenko, Kira Bobrovnikova, Peter Popov, Viacheslav Kharchenko, Dmytro Medzatyi. Spyware Detection Technique Based on Reinforcement Learning. *CEUR-WS.*Vol. 2623 ISSN: 1613–0073 (2020), pp. 307-316

14. Lysenko, S., Bobrovnikova, K., Matiukh, S., Hurman, I., Savenko, O.Detection of the botnets' low-rate DDoS attacks based on self-similarity. *International Journal of Electrical and Computer Engineering*, ISSN 2088-8708, 2020, 10(4), pp. 3651-3659.

15. Savenko, O., Sachenko, A., Lysenko, S., Markowsky, G., Vasylkiv, N. Botnet detection approach based on the distributed systems. *International Journal of Computing*, ISSN 1727-6209, 2020, 19(2), pp. 190-198.

16. N. Jacobs, S. Hossain-McKenzie and E. Vugrin, Measurement and Analysis of Cyber Resilience for Control Systems: An Illustrative Example, *Resilience Week (RWS)*, 2018, pp. 38-46, doi: 10.1109/RWEEK.2018.8473549.

17. Z. Abbadi. Security metrics what can we measure? Nova Chapter meeting presentation on security metrics, viewed. *Open Web Application Security Project (OWASP).* 2016. Volume 2.

18. S. Hopkins, E. Kalaimannan and C. S. John, Foundations for Research in

Cyber-Physical System Cyber Resilience using State Estimation, *SoutheastCon*, 2020, pp. 1-2, doi: 10.1109/SoutheastCon44009.2020.9249745.

19. H. Abdi and L. J. Williams. Principal component analysis. Wiley interdisciplinary reviews: computational statistics. 2014. Vol. 2(4). pp. 433–459.

20. N. Ahmad, U. A. Mokhtar, W. Fariza Paizi Fauzi, Z. A. Othman, Y. Hakim Yeop and S. N. Huda Sheikh Abdullah, Cyber Security Situational Awareness among Parents, *Cyber Resilience Conference (CRC)*, 2018, pp. 1-3, doi: 10.1109/CR.2018.8626830.

21. M. Albanese, S. Jajodia, and S. Noel. A time-efficient approach to cost-effective network hardening using attack graphs. *Dependable Systems and Networks (DSN'12)*. 2017. pp. 1–12.

22. C. Onwubiko, "Focusing on the Recovery Aspects of Cyber Resilience, *International Conference on Cyber Situational Awareness, Data Analytics and Assessment (CyberSA)*, 2020, pp. 1-13, doi: 10.1109/CyberSA49311.2020.9139685.

23. O. H. Alhazmi and Y. K. Malaiya. Prediction capabilities of vulnerability discovery models. *Reliability and Maintainability Symposium (RAMS'06)*. 2016. Pages 86–91.

24. S. Hopkins, E. Kalaimannan and C. S. John, Cyber Resilience using State Estimation Updates Based on Cyber Attack Matrix Classification, *IEEE Kansas Power and Energy Conference (KPEC)*, 2020, pp. 1-6, doi: 10.1109/KPEC47870.2020.9167652.

25. S. Ahmed-Zaid, S. M. Loo, A. Valdepena-Delgado and T. Beam, Cyber-Physical Security Assessment and Resilience of a Microgrid Testbed, *2021 Resilience Week (RWS)*, 2021, pp. 1-3, doi: 10.1109/RWS52686.2021.9611806.

26. D. Irene Christine and M. Thinyane, Comparative Analysis of Cyber Resilience Strategy in Asia-Pacific Countries, 2020 IEEE Intl Conf on Dependable, Autonomic and Secure Computing, Intl Conf on Pervasive Intelligence and Computing, Intl Conf on Cloud and Big Data Computing, *Intl Conf on Cyber Science and Technology Congress (DASC/PiCom/CBDCom/CyberSciTech*), 2020, pp. 71-78, doi: 10.1109/DASC-PICom-CBDCom-CyberSciTech49142.2020.00027.

27. Clark and S. Zonouz, Cyber-Physical Resilience: Definition and Assessment Metric, *IEEE Transactions on Smart Grid*, vol. 10, no. 2, pp. 1671-1684, March 2019, doi: 10.1109/TSG.2017.2776279.

28. M. Segovia, J. Rubio-Hernan, A. R. Cavalli and J. Garcia-Alfaro, Cyber-Resilience Evaluation of Cyber-Physical Systems, *2020 IEEE 19th International Symposium on Network Computing and Applications (NCA)*, 2020, pp. 1-8, doi: 10.1109/NCA51143.2020.9306741.

29. H. Heck, O. Kieselmann and A. Wacker, Evaluating Connection Resilience for Self-Organizing Cyber-Physical Systems, *2016 IEEE 10th International Conference on Self-Adaptive and Self-Organizing Systems (SASO)*, 2016, pp. 140-141, doi: 10.1109/SASO.2016.20.

30. D. Galinec and W. Steingartner, Combining cybersecurity and cyber defense to achieve cyber resilience, *2017 IEEE 14th International Scientific Conference on Informatics*, 2017, pp. 87-93, doi: 10.1109/INFORMATICS.2017.8327227.

31. M. S. Abdullah, A. Zainal, M. A. Maarof and M. Nizam Kassim, Cyber-Attack Features for Detecting Cyber Threat Incidents from Online News, *2018 Cyber Resilience Conference (CRC)*, 2018, pp. 1-4, doi: 10.1109/CR.2018.8626866.

32. K. Ligo, A. Kott and I. Linkov, How to Measure Cyber-Resilience of a System With Autonomous Agents: Approaches and Challenges, *IEEE Engineering Management Review*, vol. 49, no. 2, pp. 89-97, 1 Secondquarter,june 2021, doi: 10.1109/EMR.2021.3074288.

33. K. Hasan, S. Shetty, A. Hassanzadeh and S. Ullah, Towards Optimal Cyber Defense Remediation in Cyber Physical Systems by Balancing Operational Resilience and Strategic Risk, *MILCOM 2019 - 2019 IEEE Military Communications Conference (MILCOM)*, 2019, pp. 1-8, doi: 10.1109/MILCOM47813.2019.9021076.

34. J. Rajamäki, Industry-university collaboration on IoT cyber security education: *Academic course: Resilience of Internet of Things and cyber-physical systems*, *2018 IEEE Global Engineering Education Conference (EDUCON)*, 2018, pp. 1969-1977, doi: 10.1109/EDUCON.2018.8363477.

35. C. Rieger, C. Kolias, J. Ulrich and T. R. McJunkin, A Cyber Resilient Design for Control Systems, *2020 Resilience Week (RWS)*, 2020, pp. 18-25, doi: 10.1109/RWS50334.2020.9241300.

36. Sood and K. Moidu, Protection of Healthcare Information: Adding Cyber Resilience and Recovery, *2018 International Conference on Computational Science and Computational Intelligence (CSCI)*, 2018, pp. 132-134, doi: 10.1109/CSCI46756.2018.00033.

37. Sood and K. Moidu, Protection of Healthcare Information: Adding Cyber Resilience and Recovery, *2018 International Conference on Computational Science and Computational Intelligence (CSCI)*, 2018, pp. 132-134, doi: 10.1109/CSCI46756.2018.00033.

38. S. Alrabaee, P. Shirani, L. Wang, and M. Debbabi. SIGMA: A semantic integrated graph matching approach for identifying reused functions in binary code. *Digital Investigation*. 2015. Vol. 12. pp. 61–71.

39. P. Ammann, D. Wijesekera, and S. Kaushik. Scalable, graph-based network vul- nerability analysis. *Computer and Communications Security (CCS'02)*. 2015. Pages 217–224.

40. A. Avizienis and L. Chen. On the implementation of n-version programming for software fault tolerance during execution. *Computer Software and Applications Conference (COMPSAC'77).* 1977. Volume 77. Pages 149–155.

41. D. Balzarotti, M. Monga, and S. Sicari. Assessing the risk of using vulnerable components. *Quality of Protection (QoP'06)*. 2016. Pages 65–77.

42. A. Bartel, J. Klein, Y. Le Traon, and M. Monperrus. Automatically securing permission-based software by reducing the attack surface: An application to an- droid. *IEEE/ACM International Conference on Automated Software Engineering (ASE'12)*. 2016. pp. 274–277.

43. H. A. Basit and S. Jarzabek. Efficient token based clone detection with flexible tokenization. *European Software Engineering Conference and Symposium on the Foundations of Software Engineering.* 2017. pp. 513–516.

44. S. Bhatkar, D. DuVarney, and R. Sekar. Address obfuscation: An efficient approach to combat a broad range of memory error exploits. *USENIX Security Symposium*. 2013. Vol. 120.

45. S. Bhatkar and R. Sekar. Data space randomization. *Detection of Intrusions and Malware, and Vulnerability Assessment*. 2018. pp.1–22.

46. L. Breiman. Decision tree forest. *Machine Learning*. 2013. Vol. 45. pp. 5–32.

47. R. Brixtel, M. Fontaine, B. Lesner, C. Bazin, and R. Robbes. Language-independent clone detection applied to plagiarism detection. *Source Code Analysis and Ma- nipulation (SCAM'10)*. 2018. pages 77–86.

48. J. Caballero, T. Kampouris, D. Song, and J. Wang. Would diversity really increase the robustness of the routing infrastructure against software defects? *Network and Distributed System Security (NDSS'08).* 2008. page 40.

49. S. Checkoway, D. McCoy, B. Kantor, D. Anderson, H. Shacham, S. Savage, K. Koscher, A. Czeskis, F. Roesner, T. Kohno, et al. Comprehensive experimental analyses of automotive attack surfaces. *USENIX Security Symposium.* 2011.

50. H. Chen, D. Wagner, and D. Dean. Setuid demystified. *USENIX Security Sympo- sium*. 2012. pages 171–190.

51. B. Chun, P. Maniatis, and S. Shenker. Diverse replication for single-machine byzantine-fault tolerance. *USENIX Annual Technical Conference*. 2018. pages 287– 292.

52. B. Cox, D. Evans, A. Filipi, J. Rowanhill, W. Hu, J. Davidson, J. Knight, A. Nguyen- Tuong, and J. Hiser. N-variant systems: A secretless framework for security through diversity. *Defense Technical Information Center*. 2006.

53. CVE Details. CVE for Ubuntu 11.04. URL: http://www.cvedetails.com/ vulnerability- list/ vendor_id-4781/ product_id-20550/ version_id-104819/Canonical-Ubuntu-Linux-11.04.html (дата звернення: 25.04.2022).

54. M. Dacier. Towards quantitative evaluation of computer security. Ph.D. Thesis, Institut National Polytechnique de Toulouse. 1994.

55. A. Danial. Count lines of code (cloc). URL: https://github.com/AlDanial/

cloc. (дата звернення: 25.04.2022).

56. M. Davari and M. Zulkernine. Analysing vulnerability reproducibility for Firefox browser. *Privacy, Security and Trust (PST'16)*. 2016. pages 674–681.

57. M. Doyle and J. Walden. An empirical study of the evolution of php web application security. *Security Measurements and Metrics (Metrisec'11).* 2011. pages 11–20.

58. T. Dullien, E. Carrera, S.-M. Eppler, and S. Porst. Automated attacker correlation for malicious code. Technical report. Ruhr-University Bochum (Germany). 2010.

59. W. S. Evans, C. W. Fraser, and F. Ma. Clone detection via structural abstraction. *Software Quality Journal*. 2013. Vol. 17(4). pp. 309–330.

60. Falliere, L. O. Murchu, and E. Chien. W32.stuxnet dossier. Symantec Security Response, 2014.

61. D. Farmer and E. H. Spafford. The COPS security checker system. Technical Report 90-993, Purdue University. Computer Science Technical Reports. 1990.

62. M. Frigault and L. Wang. Measuring network security using bayesian network-based attack graphs. *Security, Trust, and Privacy for Software Applications (STPSA'08)*. 2018. pages 698–703.

63. M. Frigault, L. Wang, A. Singhal, S. Jajodia. Measuring network security using dynamic bayesian network. *Quality of Protection (QoP'08)*. 2008. pages 23–30.

64. D. Gao, M. Reiter, and D. Song. Behavioral distance measurement using hidden markov models. *Recent Advances in Intrusion Detection (RAID'06)*. 2016. pp. 19–40.

65. M. Garcia, A. Bessani, I. Gashi, N. Neves, and R. Obelheiro. OS diversity for intru- sion tolerance: Myth or reality? *Dependable Systems and Networks*. 2015. pages 383–394.

66. N. Gruschka and M. Jensen. Attack surfaces: A taxonomy for attacks on cloud services. *International Conference on Cloud Computing (CLOUD'10)*. 2010. pages

276–279.

67. G. Gu, P. Fogla, D. Dagon, W. Lee, and B. Skoric. Towards an information-theoretic framework for analyzing intrusion detection systems. *European Symposium on Research in Computer Security (ESORICS'06)*. 2006. pages 527–546.

68. M. A. Hall. Correlation-based Feature Selection for Machine Learning. PhD thesis, University of Waikato, Hamilton, New Zealand. Apr. 1999.

69. M. Hill. Diversity and evenness: a unifying notation and its consequences. *Ecology*. 1973. Vol. 54(2). pp. 427–432.

70. G. E. Hinton and S. T. Roweis. Stochastic neighbor embedding. *Advances in Neural Information Processing Systems*. 2003. pages 857–864.

71. H. Holm, M. Ekstedt, and D. Andersson. Empirical analysis of system-level vulner- ability metrics through actual attacks. *IEEE Transactions on Dependable and Secure Computing*. Nov. 2012. Vol. 9(6). pp. 825–837.

72. M. Howard, J. Pincus, and J. Wing. Measuring relative attack surfaces. *Computer Security in the 21st Century*. 2013. pages 109–137.

73. N. Idika and B. Bhargava. Extending attack graph-based security metrics and aggre- gating their application. *IEEE Transactions on Dependable and Secure Computing*. 2012. Vol. 9. pp. 75–85.

74. K. Ingols, M. Chu, R. Lippmann, S. Webster, and S. Boyer. Modeling modern net- work attacks and countermeasures using attack graphs. *Annual Computer Security Applications Conference (ACSAC'09)*. 2019. pp. 117–126.

75. S. Jajodia. Topological analysis of network attack vulnerability. *ACM Symposium on Information, Computer and Communications Security (ASIACCS'07)*. 2007. page 2.

76. S. Jajodia, A. Ghosh, V. S. Subrahmanian, V. Swarup, C. Wang, and X. Wang. Moving Target Defense II: Application of Game Theory and Adversarial Modeling. *Springer*. 2012.

77. S. Jajodia, A. Ghosh, V. Swarup, C. Wang, and X. Wang. Moving Target Defense: Creating Asymmetric Uncertainty for Cyber Threats. Springer. 1st edition, 2017.

78. J. Jang, D. Brumley, and S. Venkataraman. Bitshred: Fast, scalable malware triage. *Cylab, 22*. 2010.

79. A. Jaquith, Addison Wesley. Security Merics: Replacing Fear Uncertainity and Doubt, 2017.

80. W. Wu, R. Kang and Z. Li, Risk assessment method for cybersecurity of cyber-physical systems based on inter-dependency of vulnerabilities, *2015 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM)*, 2015, pp. 1618-1622, doi: 10.1109/IEEM.2015.7385921.

81. R. Zhou, M. Peng and X. Gao, Vulnerability Assessment of Power Cyber-Physical System Considering Nodes Load Capacity, *2021 6th International Conference on Intelligent Computing and Signal Processing (ICSP)*, 2021, pp. 1438-1441, doi: 10.1109/ICSP51882.2021.9408825.

82. C. Couch and T. Wei, Cyber Vulnerabilities in ATE Systems, 2019 IEEE AUTOTESTCON, 2019, pp. 1-4, doi: 10.1109/AUTOTESTCON43700.2019.8961067.

83. Y. Mishina, K. Takaragi and K. Umezawa, A Method of Threat Analysis for Cyber-Physical System using Vulnerability Databases, *2018 IEEE International Symposium on Technologies for Homeland Security (HST)*, 2018, pp. 1-7, doi: 10.1109/THS.2018.8574154.

84. S. Kim, Y. Won, I. -H. Park, Y. Eun and K. -J. Park, Cyber-Physical Vulnerability Analysis of Communication-Based Train Control, *IEEE Internet of Things Journal*, vol. 6, no. 4, pp. 6353-6362, Aug. 2019, doi: 10.1109/JIOT.2019.2919066.

85. V. Venkataramanan, A. Srivastava, A. Hahn and S. Zonouz, Enhancing Microgrid Resiliency Against Cyber Vulnerabilities, 2018 IEEE Industry Applications Society Annual Meeting (IAS), 2018, pp. 1-8, doi: 10.1109/IAS.2018.8544667.

86. Y. Kim, C. Son and S. Lee, A method of cyber security vulnerability test for the DPPS and PMAS test-bed, *2017 17th International Conference on Control, Automation and Systems (ICCAS)*, 2017, pp. 1749-1752, doi: 10.23919/ICCAS.2017.8204258.

87. S. Sunny, V. Pavithran and K. Achuthan, Synthesizing perception based on analysis of cyber attack environments, *2014 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, 2014, pp. 2027-2030, doi: 10.1109/ICACCI.2014.6968639.

88. F. N. Ugwoke, K. C. Okafor and V. C. Chijindu, Security QoS profiling against cyber terrorism in airport network systems, *2015 International Conference on Cyberspace (CYBER-Abuja),* 2015, pp. 241-251, doi: 10.1109/CYBER-Abuja.2015.7360516.

Додаток А

(обов’язковий)

Відмінності на рівнях файлів і модифікацій між різними версіями ПЗ ІТ-інфраструктур

Рисунок А.1 – Відмінності на рівнях файлів і модифікацій між різними версіями ПЗ ІТ-інфраструктур

Додаток Б

(обов’язковий)

Копія тез доповіді на Всеукраїнській науково-практичній конференції Актуальні Проблеми Комп’ютерних Наук (АПКН-2021)

УДК 004.7.056.5

Заграй А. О., Лисенко С. М.

Хмельницький національний університет

МЕТОД ПОБУДОВИ РЕЗИЛЬЄНТНИХ IT-ІНФРАСТРУКТУР В УМОВАХ ЗДІЙСНЕННЯ АТАК МЕРЕЖНОГО ТИПУ

Запропоновано метод побудови резильєнтних IT-інфраструктур в умовах здійснення атак мережного типу. В роботі здійснено формальне моделювання мережевої диверсифікації як метрики резильєнтності IT-інфраструктури. Метод передбачає здійснення обчислення метрики на основі ефективної кількості різних залучених ресурсів, а також залучення взаємодоповнювальних метрик на основі найменших та середніх зусиль здійснення атаки відповідно. Пропонований метод оперує множиною поведінок атак у мережі і виконує оцінку резильєнтності мережі на основі аналізу внутрішніх властивостей програмних додатків, а також оцінку показника безпеки для оцінки стійкості мереж до потенційних атак нульового дня.

The method of construction of resistant IT infrastructures in the conditions of network type attacks is offered. The formal modeling of network diversification as a metric of IT infrastructure resilience is carried out in the work. The method involves the calculation of metrics based on the effective number of different resources involved, as well as the involvement of complementary metrics based on the smallest and average effort to carry out the attack, respectively. The proposed method operates with a set of behaviors of attacks on the network and evaluates the resilience of the network based on the analysis of the internal properties of software applications, as well as the evaluation of security to assess the resilience of networks to potential zero-day attacks.

Сьогодні сучасні комп'ютерні мережі відіграють надзвичайно велику роль у багатьох IT-інфраструктурах: критичних, урядових та військових організаціях та підприємствах [1-3].

Захист таких критично важливих мереж означає більше, ніж просто виправлення відомих вразливостей та розгортання брандмауерів або IDS.

Відповідні показники необхідні для оцінки рівня безпеки мереж та забезпечення розширених рішень щодо безпеки, а саме рівня резильєнтності IT-інфраструктур – здатності передбачати мережні кібератаки, протистояти їм та мати здатність до відновлення після їх здійснення [1-5].

Однак, не враховуючи невідомих вразливостей нульового дня (англ. Zero-day / 0day), показників безпеки недостатньо, щоб чисельно відобразити справжній рівень безпеки мережі [4-7].

Дане дослідження присвячене а робота спрямована на розробку метод побудови резильєнтних IT-інфраструктур в умовах здійснення атак мережного типу шляхом впровадження нових метрик безпеки мережі в умовах здійснення атак.

Особливістю методу передбачається моделювання атак нульового дня та вивченні взаємозв'язку між функціями програмного забезпечення та вразливими місцями комп'ютерних мереж, які зазнають атак.

В дослідженні пропонується здійснити формальне моделювання мережевої диверсифікації як метрики резильєнтності IT-інфраструктури шляхом розробки та оцінки серії показників диверсифікації.

Метод передбачає здійснення обчислення метрики на основі ефективної кількості різних залучених ресурсів.

Також метод передбачає залучення взаємодоповнювальних метрик на основі найменших та середніх зусиль здійснення атаки відповідно.

Також пропонований метод оперує множиною поведінок атак у мережі і виконує оцінку резильєнтності мережі на основі аналізу внутрішніх властивостей програмних додатків, а також оцінку показника безпеки для оцінки стійкості мереж до потенційних атак нульового дня.

З цією метою було розробляємо моделі поведінок атаки між різними з урахування використання атаками ресурсів всередині мережі.

Висновки щодо присутності та здатності до резильєнтного функціонування мережі в умовах здійснення атак покладено на запропоновані евристичні алгоритми, що зменшує загальну обчислювальну складність роботи методу.

Для перевірки ефективності запропонованого методу було здійснено ряд експериментальних досліджень, результати яких показують здатність методу оцінки резильєнтності IT-інфраструктур в умовах атак, а також здійснення виявлення атак нульового дня на рівні понад 94% та хибних спрацювань 6%.

Результати роботи методу подано в таблиці 1.

Таблиця 1 – Результати роботи методу

Параметри	**Значення**
Кількість аналізованих атак нульового дня	3446
Кількість аналізованих вразливостей	23300
Поріг виявлення,%	25
Ефективність виявлення, %	94,5
хибні спрацювання, %	5,88

Перелік посилань

1. Лисенко С. М., Харченко В.С., Бобровнікова К.Ю., Щука Р. Резильєнтність комп'ютерних систем в умовах кіберзагроз: Онтологія та таксономії. Радіоелектронні і комп'ютерні системи. 2020. №1. С. 17-28.
2. S. Hopkins, E. Kalaimannan and C. S. John, Foundations for Research in Cyber-Physical System Cyber Resilience using State Estimation, 2020 SoutheastCon, 2020, pp. 1-2, doi: 10.1109/SoutheastCon44009.2020. 9249745.
3. N. Jacobs, S. Hossain-McKenzie and E. Vugrin, Measurement and Analysis of Cyber Resilience for Control Systems: An Illustrative Example, 2018 Resilience Week (RWS), 2018, pp. 38-46, doi: 10.1109/RWEEK.2018.8473549.
4. C. Onwubiko, Focusing on the Recovery Aspects of Cyber Resilience, 2020 International Conference on Cyber Situational Awareness, Data Analytics and Assessment (CyberSA), 2020, pp. 1-13.
5. S. Hopkins, E. Kalaimannan and C. S. John, Cyber Resilience using State Estimation Updates Based on Cyber Attack Matrix Classification, 2020 IEEE Kansas Power and Energy Conference (KPEC), 2020, pp. 1-6, doi: 10.1109/KPEC47870.2020.9167652.
6. Лисенко С. М. Моделі кібератак мережного та хостового типу. Вимірювальна та обчислювальна техніка в технологічних процесах. 2019. №2. С. 58-63.
7. Лисенко С. М. Метод забезпечення резильєнтності комп'ютерних систем в умовах кібер-загроз на основі самоадаптивності. Радіоелектронні і комп'ютерні системи. 2019. №4. С. 4–16.
8. A. Warzyński and G. Kołaczek, "Intrusion detection systems vulnerability on adversarial examples," 2018 Innovations in Intelligent Systems and Applications (INISTA), 2018, pp. 1-4, doi: 10.1109/INISTA.2018.8466271.
9. Z. S. Malek, B. Trivedi, A. Shah, User behavior Pattern -Signature based Intrusion Detection, 2020 Fourth World Conference on Smart Trends in Systems, Security and Sustainability (WorldS4), 2020, pp. 549-552.

Додаток В

(обов'язковий)

Презентація доповіді

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
Кафедра комп'ютерної інженерії та інформаційних систем

Заграй Альона

Метод побудови резильєнтних IT-інфраструктур в умовах здійснення атак мережного типу

Науковий керівник – д.т.н. проф. Лисенко С.М.

Хмельницький - 2022

Мета і задачі дослідження

Метою роботи є забезпечення резильєнтного функціонування IT-інфраструктур в умовах здійснення атак мережного типу.

Об'єкт дослідження – процес побудови резильєнтних IT-інфраструктур в умовах здійснення атак мережного типу.

Предмет дослідження – моделі, удосконалений метод та програмно-технічні засобами побудови резильєнтних IT-інфраструктур в умовах здійснення атак мережного типу.

Мета і задачі дослідження

Поставлена мета досягається розв'язанням таких основних **задач**:

1. Розглянути поняття резильєнтного функціонування та оцінка резильєнтності іт-інфраструктур в умовах здійснення атак мережного типу. Дослідити відомі методи оцінки резильєнтності мережі в умовах атак. Проаналізувати та дослідити показники резильєнтності мережі в умовах здійснення атак, зокрема показник різноманітності мережі, метрику покриття атаки мережі.
2. Здійснити моделювання показників безпеки для оцінки резильєнтності іт-інфраструктур в умовах здійснення атак мережного типу.
3. Виконати формальне моделювання різноманітності мережі.
4. Розробити удосконалений метод побудови резильєнтних ІТ-інфраструктур в умовах здійснення атак мережного типу.
5. Практично реалізувати програмно-апаратних засобів удосконаленого методу побудови резильєнтних ІТ-інфраструктур в умовах здійснення атак мережного типу з тестуванням на проникнення.

Наукова новизна отриманих результатів

1. Удосконалено метод удосконалений метод побудови резильєнтних ІТ-інфраструктур в умовах здійснення атак мережного типу, який на відміну від відомих враховує розрахунках ймовірності мережевої атаки на ІТ-інфраструктуру, ймовірності атаки на основі загальної системи оцінки вразливостей, і який забезпечують резильєнтне функціонування ІТ-інфраструктур в у мовах атак.

2. Набули подальшого розвитку програмно-технічні засоби побудови резильєнтних ІТ-інфраструктур в умовах здійснення атак мережного типу, які забезпечують резильєнтне функціонування ІТ-інфраструктур в у мовах атак.

Практичне значення отриманих результатів

Практична значимість отриманих результатів полягає у В результаті виконаного наукового дослідження було розроблено програмно-технічні засоби побудови резильєнтних ІТ-інфраструктур в умовах здійснення атак мережного типу, які забезпечують резильєнтне функціонування ІТ-інфраструктур в умовах атак.

Актуальність дослідження

Тривожна статистика кібербезпеки

- Факти кібербезпеки в 2022 році:

- потрібно півроку, щоб виявити злом даних,
- 43% усіх кібератак спрямовані на малий бізнес.
- 91% атак запускаються з фішингової електронної пошти.
- Підприємство стає жертвою атаки програм-вимагачів кожні 14 секунд .
- 38% шкідливих вкладень маскуються як файли того чи іншого типу Microsoft Office.
- компанії стикалися в середньому з 22 порушеннями безпеки .
- Очікується, що глобальна вартість онлайн-злочинності досягне 6 трильйонів доларів до 2023 року .

Удосконалений метод побудови резильєнтних IT-інфраструктур в умовах здійснення атак мережного типу

Основи методу

Метод ґрунтується на розрахунках:

- ❑ ймовірності мережевої атаки на IT-інфраструктуру, ймовірності атаки на основі CVSS,
- ❑ імовірності атаки на основі графа
- ❑ застосовує агрегацію ймовірностей атак всередині мережі.

Удосконалений метод побудови резильєнтних IT-інфраструктур в умовах здійснення атак мережного типу

Основи методу

Метод залучає евристичні алгоритми для обчислення поверхні мережевої атаки на IT-інфраструктуру, зокрема застосування евристичних алгоритмів випадкового вибору та вибору частоти

Удосконалений метод побудови резильєнтних IT-інфраструктур в умовах здійснення атак мережного типу

Розрахунок ймовірності атаки на основі CVSS

Нехай b_i та s_i ($1 \leq i \leq h$) позначають базову оцінку та кількість методів кожної групи відповідно.

Припустимо, що в середньому буде існувати одна вразливість нульового дня для кожні n методів, і ймовірність виявлення такої вразливості зловмисниками дорівнює r_0 (k і r_0 обидва призначені як нормалізуючі константи; див. нижче для додаткової інформації).

Удосконалений метод побудови резильєнтних IT-інфраструктур в умовах здійснення атак мережного типу

Розрахунок ймовірності атаки на основі CVSS

Базовий бал, поділений на його діапазон 10, дає ймовірність того, що вразливість у цій групі може бути використана; множення цього на r_0 дає ймовірність того, що метод може бути як відкритий, так і використаний; s_i/k показує кількість вразливостей із методів s_i в цій групі; отже, рівняння дає ймовірність r того, що програмне забезпечення містить принаймні одну уразливість нульового дня проти IT-інфраструктури, яку можна використовувати:

$$r = \prod_{i=1}^{h}(1 - r_0 \frac{b_i}{10})^{\frac{s_i}{k}} \quad (1)$$

Удосконалений метод побудови резильєнтних IT-інфраструктур в умовах здійснення атак мережного типу

Розрахунок імовірності атаки

Наступним кроком удосконаленого методу є врахування взаємозв'язків між різними розмірами поверхні атаки, напр. канали та ненадійні елементи даних розглядаються лише як непрямі вхідні дані в процесі відображення, що відображає доступність та вплив методів відповідно.

Під час роботи фіксуємо зв'язки між різними ресурсами за допомогою моделі, яка представляє можливі атаки між ресурсами IT-інфраструктури, а потім агрегуємо загальну ймовірність атаки щодо критичної умови або активу, сформованого як умову цілі.

Удосконалений метод побудови резильєнтних ІТ-інфраструктур в умовах здійснення атак мережного типу

Розрахунок імовірності атаки

Далі здійснюється поєднання різних вимірів поверхні атаки, розглядаючи точку зору зловмисників, коли атака зазвичай вимагає каналів зв'язку (вимір каналу) для доступу до методів і викликає методи для маніпулювання ненадійними елементами даних для досягнення своєї мети.

Якщо ПЗ ІТ-інфраструктури ізольовано від КМ і не має пов'язаних з нею каналів, зловмисники не зможуть здійснити атаки на програмне забезпечення незалежно від кількості методів і ненадійних елементів даних.

Аналогічно, якщо програмна програма не має жодного методу, зловмисники не зможуть отримати доступ до ненадійних елементів даних. Це спостереження показує, що для завершення атаки зазвичай потрібна комбінація трьох вимірів.

Удосконалений метод побудови резильєнтних ІТ-інфраструктур в умовах здійснення атак мережного типу

Розрахунок імовірності атаки

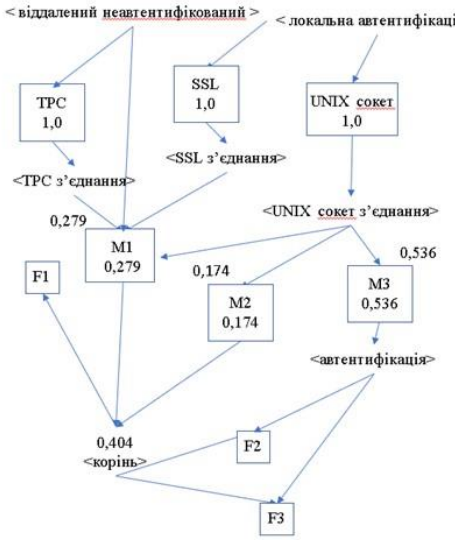
Базовий бал, поділений на його діапазон 10, дає ймовірність знайти метод у програмній програмі, який можна використовувати; помноження цього на r_0 дає ймовірність того, що метод можна як виявити, так і використати.

r_0 призначено лише як нормалізуючу константу:

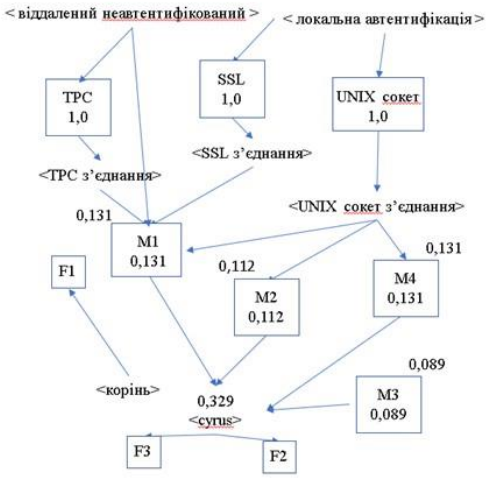
$$r = 1 - (1 - r_0 \frac{b_i}{10})^{s_i} \qquad (2)$$

Удосконалений метод побудови резильєнтних IT-інфраструктур в умовах здійснення атак мережного типу

Графи поверхні атаки



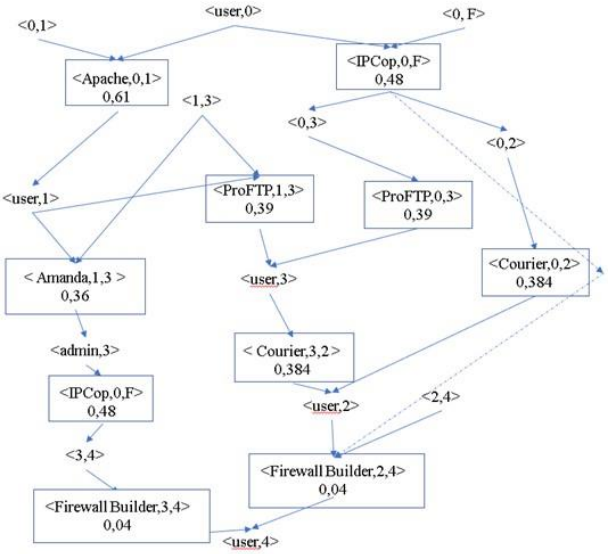
для Courier



для Cryus

Удосконалений метод побудови резильєнтних IT-інфраструктур в умовах здійснення атак мережного типу

Граф мережевих ресурсів із ймовірністю атаки для мережі



Удосконалений метод побудови резильєнтних ІТ-інфраструктур в умовах здійснення атак мережного типу

Застосування евристичних алгоритмів випадкового вибору та вибору частоти

Випадковий вибір. Найочевиднішим рішенням, ймовірно, є вибір ресурсів абсолютно випадковим способом, а саме евристичним методом випадкового вибору.

Удосконалений метод побудови резильєнтних ІТ-інфраструктур в умовах здійснення атак мережного типу

Застосування евристичних алгоритмів випадкового вибору та вибору частоти

Вибір частоти. Ідея цієї евристики полягає в тому, що, оскільки один і той самий ресурс може з'являтися на кількох хостах всередині мережі, обчислення поверхні атаки для ресурсів, які найчастіше зустрічаються, надасть найбільшу інформацію з однаковою вартістю.

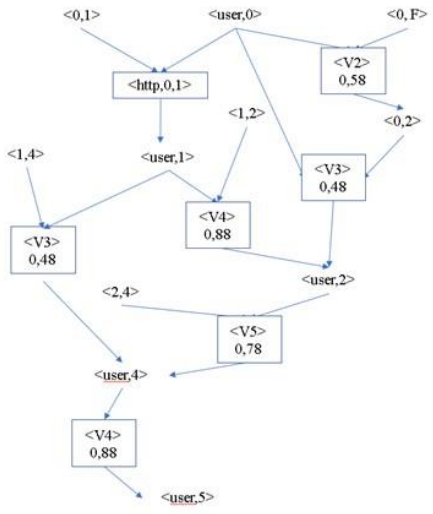
Удосконалений метод побудови резильєнтних IT-інфраструктур в умовах здійснення атак мережного типу

Застосування евристичних алгоритмів випадкового вибору та вибору частоти

Топологічний порядок. Ідея полягає в тому, що, оскільки вузли, ближчі до першого та останнього вузлів графа ресурсів (у сенсі топологічного сортування), мають тенденцію спільно використовувати більше шляхів атаки, це може допомогти вибрати ресурси на основі топологічного порядку серед атак мережного типу.

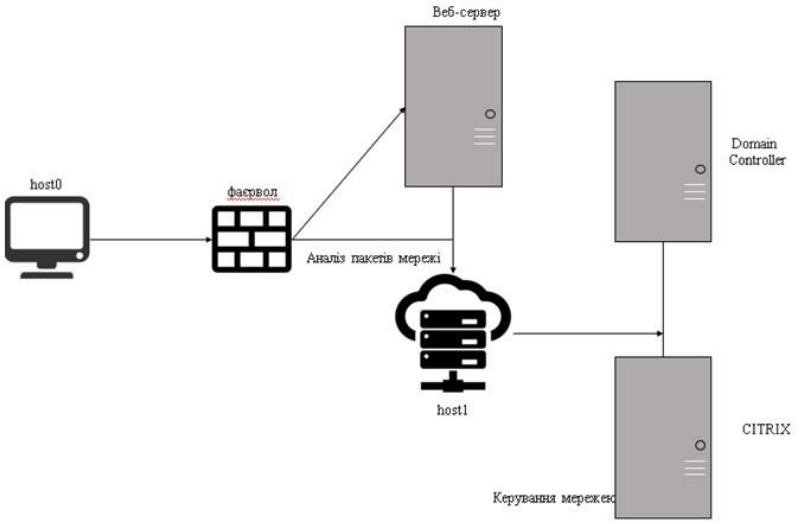
Удосконалений метод побудови резильєнтних IT-інфраструктур в умовах здійснення атак мережного типу

Застосування евристичних алгоритмів Mpath-Topo та Keynode



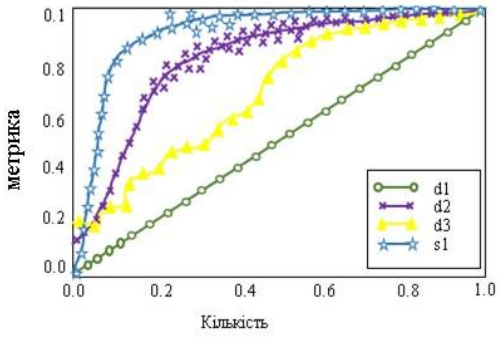
Програмно-апаратний засоби забезпечення побудови резильєнтних іт-інфраструктур в умовах здійснення атак мережного типу

Конфігурація комп'ютерної ІТ-інфраструктури

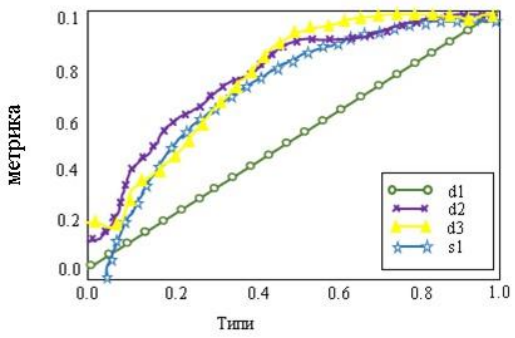


Програмно-апаратний засоби забезпечення побудови резильєнтних ІТ-інфраструктур в умовах здійснення атак мережного типу

Поширення атаки мережного типу



10% початково успішних атак мережного типу



50% початково успішних атак мережного типу

Публікації за матеріалами дипломної роботи

1. Заграй А.О., Лисенко С.М. Метод побудови резильєнтних ІТ-інфраструктур в умовах здійснення атак мережного типу. Збірник наукових праць XIII Всеукраїнської науково-практичної конференції «Актуальні проблеми комп'ютерних наук АПКН-2021». Хмельницький, 2021. с. 15-16.

Зв'язок роботи з науковими програмами, планами, темами

Дослідження, представлені у кваліфікаційній роботі, проводились в рамках держбюджетної НДР Хмельницького національного університету 1Б-2021 «Самоорганізована розподілена система виявлення зловмисного програмного забезпечення в комп'ютерних мережах» (ДР № 0121U109936) 2021-2022 рр.

Висновки

В першому розділі розглянуто поняття резильєнтного функціонування та оцінка резильєнтності іт-інфраструктур в умовах здійснення атак мережного типу, досліджено відомі методи оцінки резильєнтності мережі в умовах атак.

Проаналізовано та досліджено показники резильєнтності мережі в умовах здійснення атак, зокрема показник різноманітності мережі, метрику покриття атаки мережі.

В розділі досліджено моделі вразливостей, показники безпеки, показник різноманітності мережі, метрику поверхні мережевих атак, а також модель виявлення вразливостей. В розділі зроблено висновки щодо необхідності розроблення удосконалений метод побудови резильєнтних ІТ-інфраструктур в умовах здійснення атак мережного типу.

Висновки

В другому розділі здійснено моделювання показників безпеки для оцінки резильєнтності іт-інфраструктур в умовах здійснення атак мережного типу.

Зокрема, виконано формальне моделювання різноманітності мережі, подано особливості застосування моделі.

В розділі розглянуто показник мережевого різноманіття, натхненного біорізноманіттям та показник різноманітності мережі на основі найменших зусиль. В розділі також представлено модель різноманітність на основі найменших зусиль для атаки. Розглянута імовірнісна різноманітність мережі та імовірнісна модель мережевого різноманіття.

Надано рекомендації щодо створення моделей мережевого різноманіття.

Висновки

В третьому розділі представлено удосконалений метод побудови резильєнтних ІТ-інфраструктур в умовах здійснення атак мережного типу.

Метод ґрунтується на розрахунках ймовірності мережевої атаки на ІТ-інфраструктуру, ймовірності атаки на основі CVSS, імовірності атаки на основі графа, і, на відміну від відомих , застосовує агрегацію ймовірностей атак всередині мережі.

Також мето залучає евристичні алгоритми для обчислення поверхні мережевої атаки на ІТ-інфраструктуру, зокрема застосування евристичних алгоритмів випадкового вибору та вибору частоти.

Висновки

В четвертому розділі представлено практичну реалізацію програмно-апаратних засобів удосконаленого методу побудови резильєнтних ІТ-інфраструктур в умовах здійснення атак мережного типу.

Також в розділі подано конфігурацію мережі ІТ-інфраструктур з тестуванням на проникнення, здійснено експериментальні дослідження програмно-апаратної реалізації методу, зокрема, аналіз отриманих результатів практичного застосування рішення.

В результаті отримано висновки, що застосування методу забезпечує резильєнтне функціонування ІТ-інфраструктур в умовах здійснення атак мережного типу

UNICHECK

Ім'я користувача:
Кафедра КІ

ID перевірки:
1010993783

Дата перевірки:
29.04.2022 09:04:24 EEST

Тип перевірки:
Doc vs Internet + Library

Дата звіту:
29.04.2022 09:05:08 EEST

ID користувача:
100005591

Назва документа: Метод побудови резильєнтних IT-інфраструктур в умовах здійснення атак мережного типу

Кількість сторінок: 97 Кількість слів: 18633 Кількість символів: 137014 Розмір файлу: 875.28 KB ID файлу: 1010898913

1.26%
Схожість

Найбільша схожість: 1.02% з джерелом з Бібліотеки (ID файлу: 1010898795)

0.39% Джерела з Інтернету	43	Сторінка 99
1.12% Джерела з Бібліотеки	95	Сторінка 99

0% Цитат

Не знайдено жодних цитат

Не знайдено жодних посилань

0%
Вилучень

Немає вилучених джерел

Модифікації

Виявлено модифікації тексту. Детальна інформація доступна в онлайн-звіті.

Замінені символи	17

Anti-Plagiarism v-15.257

Максимальное совпадение с одним документом 1.0%

Словари проверки: en_US, ru_RU, ua_UA. **Ошибок в документах: 8%**

	Документ		Суммарное совпадение по Базе Данных	
ID: 103211 Название: Метод побудови резильєнтних IT-інфраструктур в умовах здійснення атак мережного типу Добавлено в БД: 2022-04-29 Авторы: Заграй А.О. Руководители: Лисенко С.М. Консультанты: Опоненты:	Символы	Лексемы	Символы	Лексемы
	130455	814	1173 (1%)	11 (1%)

Источник плагиата

ID	Описание	Наличие плагиата в документе	
		Символы	Лексемы

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

РЕЦЕНЗІЯ НА ДИПЛОМНУ РОБОТУ

Дипломник: Заграй Альона Олексіївна

Тема: Метод побудови резильєнтних ІТ-інфраструктур в умовах здійснення атак мережного типу

Спеціальність: 123 «Комп'ютерна інженерія»

Обсяг дипломної роботи:

Кількість листів креслень – ; кількість сторінок записки 89

1. Короткий зміст роботи та прийнятих рішень У роботі розроблено метод побудови резильєнтних ІТ-інфраструктур в умовах здійснення атак мережного типу

2. Висновок про відповідність роботи дипломному завданню Дипломна робота відповідає виданому завданню

3. Характеристика виконання кожного розділу, ступінь використання останніх досягнень науки і техніки і передових методів роботи: В першому розділі розглянуто поняття резильєнтного функціонування та оцінка резильєнтності іт-інфраструктур в умовах здійснення атак мережного типу, досліджено відомі методи оцінки резильєнтності мережі в умовах атак. В другому розділі здійснено моделювання показників безпеки для оцінки резильєнтності іт-інфраструктур в умовах здійснення атак мережного типу. В третьому розділі представлено удосконалений метод побудови резильєнтних ІТ-інфраструктур в умовах здійснення атак мережного типу. В четвертому розділі представлено практичну реалізацію програмно-апаратних засобів удосконаленого методу побудови резильєнтних ІТ-інфраструктур в умовах здійснення атак мережного типу.

4. Позитивні сторони роботи: В результаті виконаного наукового дослідження було розроблено програмно-технічний засіб забезпечення побудови резильєнтних ІТ-інфраструктур в умовах здійснення атак мережного типу

5. Негативні сторони роботи: Не в повній мірі здійснено аналіз методів побудови резильєнтних ІТ-інфраструктур в умовах здійснення атак мережного типу

6. Оцінка графічного оформлення та пояснювальної записки роботи: Матеріали кваліфікаційної роботи є структурованими у чіткій та логічній формі та відображають послідовність виконання поставлених задач

7. Відгук про роботу в цілому: Зміст представленої роботи в повній мірі розкриває обрану тему. Дослідження, проведені є аргументованими в повній мірі.

8. Інші зауваження: Немає

9. Оцінка дипломної роботи:

Розглянувши позитивні та негативні сторони представленої дипломної роботи вважаю, що робота заслуговує оцінки «добре» 4,25 (В)

Рецензент (прізвище, ім'я, по батькові, посада, місце роботи)) Джулій В.М. , к.т.н., доцент, кафедри кібербезпеки Хмельницького національного університету

" 27 " квітня 2022р.

Завідувачу кафедри КІІС
д-р.техн.наук, проф. Говорущенко Т. О.

Заграй Альона Олексіївна
ПІБ здобувача вищої освіти

ФІТ, 2 курсу, групи КІ2м-20-1

ЗАЯВА

З правилами чинного Положення «Про дотримання академічної доброчесності в Хмельницькому національному університеті» від 26.09.2020 (зі змінами від 26.11.2020), згідно з яким виявлення плагіату є підставою для відмови в допуску кваліфікаційної роботи до захисту та застосування заходів дисциплінарної та академічної відповідальності, ознайомлений (а). Про використання програмно-технічних засобів для перевірки кваліфікаційних робіт здобувачів вищої освіти на плагіатоповіщений (а) та надаю свою згоду на обробку та збереження університетом моєї роботи в інституційному репозитарії університету.

Також надаю університету право на передачу моєї роботи для обробки та збереження в базах даних програмно-технічних засобів (Unicheck та Anti-Plagiarism) та використання роботи для виявлення плагіату в інших роботах, які перевіряються програмно-технічними засобами та користувачами, що мають доступ до цих програмно-технічних засобів, виключно в обмежених цілях для виявлення плагіату в текстах робіт.

Робота для перевірки університетом надається в друкованому та електронному варіанті. Електронна версія моєї роботи збігається (ідентична) з друкованою.

27.04.2022
дата


підпис

РІШЕННЯ ЕКСПЕРНОЇ КОМІСІЇ

КАФЕДРИ КОМП'ЮТЕРНОЇ ІНЖЕНЕРІЇ ТА СИСТЕМНОГО ПРОГРАМУВАННЯ

ПРО ДОПУСК КВАЛІФІКАЦІЙНОЇ РОБОТИ ДО ЗАХИСТУ

Підтверджуємо ознайомлення з результатом звіту подібності щодо роботи, генерованого системою виявлення текстових збігів/ідентичності/схожості:

Назва: Метод побудови резильєнтних ІТ-інфраструктур в умовах здійснення атак мережного типу

Автор: Заграй Альона Олексіївна

Спеціальність: 123 – Комп'ютерна інженерія та програмування

Освітня програма: освітньо-наукова

Науковий керівник: Лисенко С.М., д.т.н, професор

Після аналізу звіту подібності зроблено такий висновок:

№	Висновок	Позначка про відповідність
1	Запозичення, виявлені в роботі, є законними і не є плагіатом. Робота приймається до захисту.	відповідає
2	Виявлені запозичення не є плагіатом, розміщені в розділах, які не описують безпосередньо авторське дослідження, але кількість цитат перевищує обсяг, виправданий поставленою метою роботи Робота приймається до захисту, але має бути відкоригована. Відкоригований варіант має бути поданий на кафедру за 2 дні до захисту, разом із заявою щодо самостійності виконання письмової роботи та ідентичності друкованої та електронної версії роботи	
3	Виявлені запозичення не є плагіатом, але частково розміщені в розділах, які описують безпосередньо авторське дослідження, а кількість цитат перевищує обсяг, виправданий поставленою метою роботи. В зв'язку з цим мета роботи та поставлені завдання не були досягнені. Робота може бути допущена до захисту (наступного року) після того як буде відкоригована та допрацьована і успішно пройде повторну перевірку на академічний плагіат.	
4	Робота містить навмисні текстові спотворення, передбачувані спроби укриття запозичень або інші прояви академічного плагіату. Робота містить фабрикацію або фальсифікацію даних. Робота не допускається до захисту.	

Підтвердження:

Запозичення, виявлені в роботі, є незначними, законними і не є плагіатом, оскільки:

1) окремі виявлені збіги є загальновживаними фразами або виразами, про що свідчить посилання системи на збіг з 93 джерелами з бібліотек та 43 джерелами з мережі Інтернет (найбільша схожість: 1.02%) ;
2) всі зафіксовані системою ознаки модифікації тексту відносяться до комбінування латинських символів зі україномовними скороченнями індексів в формулах, що не є модифікацією тексту.

Сумарний обсяг всіх запозичень, визначений системою виявлення збігів/ ідентичності/схожості, складає 1.26% і адресується до 10 першоджерел, що, з урахуванням наведених обґрунтувань, відповідає характеру наукового дослідження і свідчить на користь кваліфікаційної роботи.

Керівник роботи _____ С.М. Лисенко

Гарант ОП _____ О. С. Савенко

Завідувач кафедри КІСП _____ Т. О. Говорущенко