Хмельницький національний університет
Факультет інформаційних технологій
Кафедра кібербезпеки

КВАЛІФІКАЦІЙНА РОБОТА МАГІСТРА

Метод виявлення вдосконалених атак на корпоративні мережі
Назва теми

Галузь знань 12 – Інформаційні технології

Спеціальність 123 – Комп'ютерна інженерія

КРМКІ.016047.20.01.02 ПЗ

Виконав: студент 2 курсу, група КІ1м-20-1 Підпис Вишневий В.В.

Керівник доц., к. т. н, доцент кафедри Кб Підпис Тітова В.Ю.

Нормоконтролер старший викладач кафедри Кб Підпис Мостовий С.В.

До захисту допускаю:
Зав. кафедри Кб, к.т.н., доц Підпис Кльоц Ю.П.

____ _____ 2021 р.

Хмельницький, 2021

ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

Факультет Інформаційних Технологій

Кафедра Кібербезпеки

Освітній рівень магістр

Галузь знань 12 Інформаційні технології

Спеціальність 123 Комп'ютерна інженерія

Освітня програма Програмування та захист комп`ютерних систем і мереж

ЗАТВЕРДЖУЮ

Зав. кафедри __Ю.П. Кльоц___

_____

_

" __ " _____2021 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

Вишневий Віктор Володимирович

Прізвище, ім'я, по батькові студента

1. Тема проекту (роботи) Метод виявлення вдосконалених атак на корпоративні мережі

Керівник роботи Тітова Вєра Юріївна

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

кандидат технічних наук, доцент

Затверджена наказом № 118 ректора університету додаток №23 від 01.09.2021

2. Строк подання студентом проєкту (роботи) на кафедру 20.11.2021

3. Вихідні дані до проекту (роботи) Удосконалена модель захисту корпоративних мереж від ускладнених атак.

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

Вступ. Дослідження відомих методів захисту від вдосконалених атак. Створення нового підходу виявлення аномалій. Комплексні підходи для виявлення та пом'якшення наслідків корпоративних атак.Оцінка методу захисту основаного на даних. Висновки.

5. Перелік графічного матеріалу (із зазначенням обов'язкових креслень) Тема, мета магістерської, наукова новизна, практична значення, публікації. Дослідження методів захисту від вдосконалених атак. Дослідження поведінки зловмисників. Проектування системи виявлення Hopper. Порівняння продуктивності Hopper з детектором структурно-аномальних входів. Висновки.

6. Консультанти розділів кваліфікаційної роботи

Розділ	Прізвище, ініціали і посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Нормоконтроль	Мостовий С.В. Старший викладач кафедри Кб		

7. Дата видачі завдання «1» 05 2021р.

КАЛЕНДАРНИЙ ПЛАН

№з/п	Назва етапів (розділів) кваліфікаційної роботи	Термін виконання етапів роботи	Примітка
1	Вибір напряму дослідження і узгодження тематики КРМ з керівником	1.09.2021	
2	Ознайомлення з предметною областю; формулювання мети і задач дослідження; визначення об'єкта і предмета дослідження	2.09.2021	
3	Робота над розділом 1 – аналіз відомих моделей, методів за темою; постановка задачі	5.09.2021	
4	Робота над розділом 2 – розробка моделей і методів для вирішення поставленої задачі	24.09.2021	
5	Робота над науковою публікацією	1.10.2021	
6	Робота над розділом 3 – розробка алгоритмів і технологій, їх аналіз	13.10.2021	
7	Робота над розділом 4 – апробація запропонованих рішень	28.10.2021	
8	Узгодження отриманих; оформлення пояснювальної записки згідно вимог	5.11.2021	
9	Попередній захист роботи	16.11.2021	
10	Захист роботи на засіданні ЕК	6.12.2021	

Студент _____ Підпис _____ Ініціали, прізвище

Керівник проекту (роботи) _____ Підпис _____ Ініціали, прізвище

АНОТАЦІЯ

Тема кваліфікаційної роботи: Метод виявлення вдосконалених атак на корпоративні мережі

Автор роботи: Вишневий Віктор Володимирович

Керівник роботи: к.т.н., доц. Тітова Вєра Юріївна

Загальний обсяг роботи: 83 сторінок, 15 рисунків, 8 таблиць, 2 додатків, 104 посилань, 2 діаграми.

Ключові слова: цільовий фішинг, бічний фішинг, бічне переміщення, виявлення вторгнень.

Метою кваліфікаційної роботи є створення нового методу виявлення вдосконалених атак для перешкодження викрадення облікових даних.

Дана кваліфікаційна робота присвячена для удосконалення методу виявлення вдосконалених атак на основі використання підходу основаного на даних.

03.12.201
Дата

Підпис студента

ANNOTATION

Theme of qualification work: Method of detection of improved attacks on corporate networks

Author of the work: Vyshnevy Viktor Volodymyrovych

Mentor: Ph.D., Assoc. Titova Vera Yuriyivna

Total volume of work: 83 pages, 18 figures, 8 tables, 2 appendices, 104 links, 3 diagrams.

Keywords: target phishing, lateral phishing, lateral movement, intrusion detection.

The purpose of the qualification work is to create a new method of detecting advanced attacks to prevent theft of credentials.

This qualification work is devoted to improving the method of detecting advanced attacks based on the use of a data-based approach.

03.12.2021

Date

Student's signature

ЗМІСТ

ВСТУП

Актуальність роботи. На сьогоднішній день цифрові атаки на підприємства залишаються невирішеною та потужною загрозою для суспільства. Протягом останнє десятиліття організації, що охоплюють сектори економіки і уряди по всьому світу, стали жертвами великої кількості атак [1, 2, 3, 4, 5]. У багатьох цих атаках зловмисники успішно викрадали конфіденційні дані про мільйони людей, починаючи від медичних карт і закінчуючи звітами про проходження державної служби безпеки, з метою отримання фінансової вигоди і шпигунства [6, 7, 8]. Тривожно і те, що ці атаки продовжують розвиватися і розширюватися, що видно на прикладі хвилі атак в стилі «ransomware», які монетизують свою компрометацію, вимагаючи гроші у користувачів і організацій [9]. Ці нові атаки не тільки завдають фінансовий збиток жертвам, але і часто завдають фізичної шкоди важливим інфраструктурам, які вони виводять з ладу в ході атаки [9, 10, 11]; дійсно, пацієнти лікарень вмирали в результаті збоїв в роботі, викликаних атаки «ransomware» [12]. Не обмежуючись випадковим ущербом, ряд атак, що здійснюються державами, свідомо і успішно намагались нанести відчутної шкоди, такі як руйнівний напад у 2014 році на Sony Pictures з боку Північної Кореї та російські кібератаки, спрямовані на електромережу України [4, 13]. У сукупності мільярди доларів фінансових втрат, конкретна шкода для людей та геополітична стабільність, і зростаюча кількість успішних атак ілюструють, що підприємства борються, щоб захиститися від сучасних атак.

На високому рівні корпоративні атаки включають два ключові етапи: дії, які зловмисник здійснює, щоб отримати доступ до мережі організації, та наступні дії, які вони виконують в межах внутрішнього середовища підприємства для досягнення кінцевої мети. Традиційні засоби захисту, які розгорнуті підприємствами, зосереджені на першому аспекті: зміцненні периметра організації для запобігання отримання зловмисником внутрішнього доступу. Ці засоби захисту, починаючи від мережевих брандмауерів, антивірусного програмного забезпечення, захищених веб-

браузерів і автоматичних оновлень, ефективно скорочують доступну поверхню атаки організації і майже усунули колись всюди сущі загрози, такі як масово розповсюджувані хробаки та завантаження за принципом «drive-by».

На жаль, сучасні атаки відображають роботу наполегливих противників, які адаптувалися до даних засобів захисту, розширивши свої стратегії експлуатації. Протягом останніх років зловмисники перейшли від використання технічних атак, що використовують слабкі місця в коді і машинах, до атак з застосуванням соціальної інженерії, спрямованим на людей за допомогою витонченої брехні і обману. Наприклад, цільовий фішинг (spearphishing) який полягає в тому, що зловмисник майстерно складає оманливий електронний лист для одного або декількох співробітників організації. Ці зловмисні електронні листи зрештою виконують своє завдання, їх жертви виконують небезпечну дію, яка дозволяє зловмиснику закріпитися за стіною оборони, яку організації розміщують на своїх кордонах. Від цих типів атак з використанням соціальної інженерії як і раніше складно захиститися, оскільки вони покладаються виключно на людський обман і маніпуляції, що дозволяє зловмисникам обходити численні технічні засоби захисту, розроблені спільнотою.

За допомогою нового засобу компрометації, орієнтованому на людину, зловмисники регулярно отримують доступ до облікових даних і машин співробітника. У деяких випадках цей початковий доступ надає зловмисникам все, що їм потрібно для досягнення своєї мети. Однак у багатьох випадках початкова машина, яка зламана зловмисником, не дозволяє їм досягти своєї мети, тому вони в кінцевому випадку шукають доступ до більш конфіденційних даних на внутрішніх серверах або хочуть поширитися на якомога більшу кількість внутрішніх машин. Таким чином, в рамках своїх атак противники часто беруть участь в процесі, відомому як бічне переміщення: використовуючи законний доступ, наданий початковим користувачем (користувачами) і машиною (машинами), для компрометації, щоб перейти на інші облікові записи та машини в мережі підприємства, поки вони недосягнуть своєї мети [8, 14, 15]. На жаль, перешкоджання

бічному руху, цей більш просунутий етап сучасних атак, залишається недостатньо вивченою проблемою, оскільки дослідникам бракує доступу до достатньо багатих та реалістичних даних, і тому, що захист традиційно зосереджується на захист машин периметра та кінцевих точок мережі підприємства. В результаті, як тільки зловмисники зламують легітимну корпоративну машину, вони часто можуть використовувати доступ і можливості, властиві їх початкового плацдарму, для безперешкодного поширення по всьому внутрішньому середовищі організації в гонитві за конфіденційними даними або руйнівним хаосом.

У цій роботі представлений багатооб'єктивний шлях до зміни цих складних корпоративних атак. Розроблено практичний набір методів та ідей, заснованих на даних, які організації можуть використовувати для протидії атак на протязі всього їх життєвого циклу. По-перше, розглянуто нові методи, які дозволяють організувати виявлення та відбиття атаки на їх периметрі. У цій першій частині звернуто особливу увагу на метод виявлення атаки типу цільовий фішинг - основного методу, який зловмисники використовують для проникнення на периметр організації. Представлено концептуальну схему опису атаку підводного фішингу та новий алгоритм виявлення аномалій, який досягається за порядком більш високої продуктивності при порівнянні з традиційними методами машинного навчання. По-друге, досліджено нові парадигми та методи для виявлення активності зловмисників у внутрішньому середовищі підприємства. А саме, якщо припустити, що зловмисник може успішно прорватися через периметр захисту організації, чи можуть організації запобігти атаці до того, як вона успішно досягне своєї шкідливої мети? Відповіддю на це питання, представляється емпірично обґрунтовані алгоритми для виявлення і перешкоджання поширенню атаки зловмисника за межі початкової точки компрометації на інші машини або облікові записи в межах підприємства. В кінцевому рахунку, розроблені методи, моделі та ідеї які можуть значнопо ліпшити безпеку підприємства від складних атак, прицьому навантаження на службу безпеки організації буде мінімальним і практичним.

Отже, із вищесказаного випливає, що вибрана тема кваліфікаційного дослідження є актуальною.

Метою кваліфікаційно дослідження є створення нового методу виявлення вдосконалених атак для перешкоджання викрадення облікових даних.

Для досягнення цієї мети під час навчання потрібно вирішити наступні завдання:

1. Проаналізувати методи виявлення вдосконалених атак на корпоративні мережі і їх проблеми з перевагами та недоліками, щоб визначити найпоширеніший метод виявлення.

2. Розглянути найпопулярніші методи атак і способи зменшення помилкових спрацювань.

3. Запропонувати метод захисту для пом’якшення та виявлення атак в корпоративній мережі.

4. Розробити ментальні моделі, які покращать розуміння і здатнійсть виявляти фішингові атаки.

5. Розробити новий набір емпіричних результатів і алгоритмів виявлення.

6. Протестувати розроблену систему виявлення.

Об’єктом дослідження є система виявленння основана на даних для виявлення атак.

Предметом дослідження є алгоритми і методи виявлення фішингових атак.

Методи дослідження, що використовуються у даній роботі, засновані на методах та алгоритмах виявлення фішингових атак.

Наукова новизна: вдосконалено метод виявлення вторгнень у корпоративну мережу, представленно новий підхід до виявлення рідкісних, немаркованих атак у великих, складних наборах даних.

Практична цінність: розроблено систему для запобігання вдосконалених атак на підприємствах.

За темою кваліфікаційної роботи опубліковано 1 стаття у фаховому журналі.

1 ЗАПОБІГАННЯ АТАК НА ПЕРИМЕТРІ ПІДПРИЄМСТВА

У перші роки існування комп'ютерної безпеки слабка захищеність програмного забезпечення та відсутність базових мережевих фільтрів дозволяли зловмисникам компрометувати організації через прості уразливості. Сьогодні ряд практичних засобів захисту, таких як мережеві брандмауери і автоматичні оновлення, роблять усе щоб такі технічні експлойти ставали усе більш складними. У відповідь на це зловмисники еволюціонували і тепер використовують новий набір методів компрометації, відомих як соціальна інженерія. Замість того щоб використовувати технічні вразливості, атаки соціальної інженерії спрямовані на те, щоб обманом змусити жертву виконати небезпечні дії від імені зловмисника за допомогою хитромудро складеного і оманливого контенту.

1.1 Пов’язані роботи та передумови

Попередні роботи по боротьбі з фішинговими атаками поділяються на три основні підходи до захисту: посилення аутентифікації механізмів перевірки, навчання безпеки, поліпшення здатності користувача самостійно виявляти і уникати фішингових атак, методи виявлення і блокування фішингових атак.

Перевірка дійсності відправника електронної пошти

Щоб забезпечити більш надійну впевненість в аутентичності відправника електронного листа, за останні кілька десятиліть спільнота запропонувала набір з трьох механізмів перевірки достовірності електронної пошти Sender Policy Framework (SPF) який дозволяє організаціям перерахувати набір IP-адрес і/або імен хостів в DNS-метаданих організації, які визначають, які сервери можуть законно відправляти електронну пошту від імені організації і її користувачів [16]. Потім поштовий сервер одержувача може використовувати ці перевірочні метадані для перевірки того, чи надісланий лист нібито від організації з авторизованого сервера вихідної пошти. DKIM - Domain-Key Identified Mail [17] - розширює ідею SPF надаючи криптографічний підпис в заголовку листа, яку одержувач може використовувати,

щоб переконатися в тому, що лист дійсно прийшов з домену передбачуваного відправника. Щоб доповнити і поліпшити розгортання цих протоколів, консорціум партнерів по галузі розробив структуру DMARC (Domain-based Message Authentication, Reporting & Conformance) [18], яка детально описує політику для ефективного використання інформації SPF і DKIM електронної пошти, обробки різних побічних випадків і надання інформації про помилки та діагностиці домену-відправнику електронної пошти.

Хоча ці системи допомагають пом'якшити наслідки фішингових атак, вони страждають від ряду фундаментальних і реальних обмежень. По-перше, всі ці системи залежать від домену легітимного відправника електронної пошти, який налаштовує і належним чином підтримує записи SPF або DKIM; без цього активного співробітництва, одержувачі електронної пошти не можуть використовувати ці системи для перевірки того, чи дійсно лист прийшов від організації відправника. По-друге, багато фішингових атак використовують розрив між передбачуваним відправником листа і його (потенційно) перевіреним відправником, від якого ці системи не можуть захистити [19, 20]. Наприклад, зловмисник, який намагається видати себе за співробітника з сайту "example.com", може замість цього створити домен "exampIe.com". (Замінивши малу літеру "L" на прописну "I"). Потім зловмисник може налаштувати свій схожий домен з відповідними SPF і DKIM і використовувати цей оманливий домен для здійснення фішингових атак. Незважаючи на їх візуальну схожість, існуючі системи DMARC, SPF і DKIM помітять фішингові листи від "exampIe" як легітимні і авторизовані. Однак нічого не підозрюючи користувачможе не зрозуміти, що шкідливі листи, відправлені з адреси електронної пошти "exampIe.com", приходять від неаутентичного відправника, залишаючи користувача вразливим для атаки, незважаючи на те, що всі ці протоколи перевірки працюють правильно. Нарешті, недавня робота показала, що практична складність доставки і перевірки електронної пошти створює ряд реальних вразливостей, які зловмисники можуть використовувати для повного обходу цих протоколів перевірки

в ряді популярних постачальників поштових служб [21, 22].

Багатофакторна аутентифікація

Щоб зменшити шкоду від успішної фішинговою атаки, фахівці з безпеки розробили протоколи "багатофакторної аутентифікації" (БФА). При аутентифікації на основі БФА користувачам необхідно надати не тільки традиційну форму перевірки, таку як пароль, але і підтвердити свою особистість за допомогою додаткового (их) методу (ів), наприклад, одноразового генерованого коду, що відправляється на мобільний пристрій. Інтуїтивно зрозуміло, що цей підхід спрямований на пом'якшення наслідків фішингових атак, оскільки зловмисникові необхідно скомпрометувати два або більше джерел аутентифікації, щоб атака була успішною.

Хоча цей підхід забезпечує значний захист від фішингових атак, він довгий час страждав від проблем, пов'язаних з зручністю використання і розгортання в реальному світі, які зберігаються і сьогодні [23,24]. Наприклад, якщо користувач втрачає свій пристрій аутентифікації з другим фактором (наприклад, зламаний токен аутентифікації або вкрадений телефон), то він ризикує втратити доступ до будь-яких ресурсів і даних, захищених аутентификацией на основі БФА. Крім того, сучасні атаки пристосувалися обходити багато з найбільш популярних і широко доступних форм багатофакторної аутентифікації [25, 26, 27].

Виявлення фішингових веб-сайтів і URL-адрес

Кілька існуючих систем спрямовані на виявлення фішингових веб-сайтів і URL-адрес з метою додавання цих сайтів в широко поширені блокують списки, які запобігають завантаження користувачем шкідливого контенту. Garrera і ін. [28] розробили класифікатор логістичної регресії, який витягує з URL набір ознак, таких як такі як реєстраційна інформація домену, наявність певних ключових слів і ознаки домену, щоб визначити, чи вказує URL на фішингові сайти. Аналогічно, PhishNet [29] намагається ідентифікувати нові фішингові URL з баз шкідливих URL, використовуючи методи природної мови, щоб обчислити, чи має новий URL близьку лексикографічну схожість з будь-яким відомим фішингових URL. Хоча ці два

підходи спираються виключно на властивості URL фішингових сайту, в інших роботах також розглядається питання про включення вмісту веб-сторінки в рішення про класифікацію [30, 31, 32]. Ці системи використовують вміст веб-сторінки, наприклад, текст з HTML DOM або зображень, що відображаються на сайті, для вилучення набору ознак які їх класифікують, використовуючи в поєднанні з ознаками, заснованими на URL-адресах, для класифікації веб-сайту як фішингового чи ні. У новіших працях також пропонуються методи, що дозволяють організаціям виявляти, коли фішингові веб-сайти видають себе за них, аналізуючи, які зовнішні домени отримують ресурси або перенаправляють користувачів на їх сайти [33].

Хоча попередні роботи забезпечують певний захист від фішингових атак і вже отримали практичне застосування в великомасштабних системах, таких як Google SafeBrowsing, вони захищають в основному від масових фішингових атак, які повторно використовують інфраструктуру атаки на безліч різних жертв або видають себе за кілька популярних брендів. Проти більш складних атак, таких як цільові фішингові листи, спрямовані на конкретну організацію, ці засоби захисту навряд чи будуть успішними. Крім того, багато з цих підходів засновані на використанні глобально розгорнутих блок-листів шкідливих URL-адрес, які стикаються з низкою проблем щодо ухилення противника, таких як маскування веб-сайтів і швидка зміна інфраструктури та змісту фішингу [30,34, 35].

Виявлення фішингових листів

Оскільки багато корпоративних користувачів стикаються з фішинговими атаками через шкідливого електронного листа, в даній роботі, а також в деяких попередніх роботах, досліджувалися методи виявлення фішингових листів [36,37, 38,39,40,41,42]. На високому рівні попередні підходи будують поведінкові моделі для типового листа відправника шляхом побудови набору характеристик на основі метаданих, стильометрії і часу відправлення листа. На жаль, ці попередні підходи не можуть виявити різні фішингові атаки, такі як атаки, відправлені з підробленої зовнішньої адреси електронної пошти без історії.

Не менш важливо і те, що алгоритми виявлення, запропоновані в попередніх роботах, генерують занадто багато помилкових спрацьовувань для практичного використання, страждаючи від повторюваної проблеми, що зустрічається в даній роботі: оману базової ставки [43]. Оскільки складні атаки, такі як цільовий фішинг, відбуваються відносно рідко, а обсяг доброякісних подій (електронних листів) на підприємствах неймовірно великий, навіть низький коефіцієнт помилкових спрацьовувань або помилок буде генерувати занадто багато сигналів тривоги для практичного використання. Для Наприклад, IdentityMailer [42] досягає рівня помилкових спрацьовувань в діапазоні 1-10%. Хоча це досить низький діапазон але на реальному наборі даних, який ми використовували в розділі 3 і який містить понад 250 000 електронних листів в день, коефіцієнт помилкових спрацьовувань в 1% привів би до щоденного обсягу понад 2500 помилкових спрацьовувань.

Для зменшення цієї загрози останнім часом розробляються класифікатори на основі природної мови для виявлення таких типів фішингових листів [44]. В даній роботі не вивчається тип фішингу, заснований на транзакціях, замість цього зосередимося на атаках, спрямованих на отримання доступу до внутрішньої середовищі підприємства і його даними, які залишаються метою багатьох складних атак [2, 3, 5, 7, 8, 45, 46, 47, 48].

1.2 Виявлення атак, пов’язаних з цільовим фішингом облікових данних

За останні кілька років численні гучні зломи показали зростаючу поширеність і міць атак цільового фішингу. Використовуючи ці атаки, противники успішно зламали широкий спектр урядових систем [2], відомих компаній [45], а останнім часом – політичних діячів та організації [5].

Атаки цільового фішингу мають кілька форм. Історично склалося так, що деякі з найбільш відомих атак включають в себе електронний лист, який намагається обдурити одержувача і змусити його завантажити шкідливе вкладення. Однак, жодна з успішних атак з використанням цільового фішингу не було шкідливого вкладення. Замість цього, переважна форма цільового фішингу, з якою стикалися, - це цільовий фішинг облікових даних, коли шкідливий електронний лист переконує одержувача натиснути на посилання, а потім ввести свої облікові дані на що відкрилася веб-сторінці. Цільовий фішинг облікових даних має неймовірно низький бар'єр входу: атакуючому для успішної атаки досить розмістити веб-сайт і створити оманливого електронного листа. Більш того, при широким поширенням віддалених робочих столів, VPN-додатків і хмарних провайдерів електронної пошти, вкрадені облікові дані часто надають зловмисникам багату інформацію і можливості.

У цьому розділі представлений новий підхід до виявлення атак цільового фішингу в корпоративних умовах. Виявлення таких атак виявляється вельми складним через проблеми з базовим рівнем. Наприклад, наша корпоративна база даних містить 370 мільйонів електронних листів, але менше 10 відомих випадків являють собою цільовий фішинг. Отже, багато природних методів зазнають невдачі через занадто високого коефіцієнта помилкових спрацьовувань: навіть при коефіцієнті помилкових спрацьовувань в 0,1% це призведе до 370 000 фіктивних тривог.

Щоб подолати ці труднощі, вносимо два ключових вклади. По-перше, представляємо аналіз характеристик, які є основоположними для атак цільового фішингу, ми сформулювали нову специфікацію атаки і відповідний набір

характеристик, які спрямовані на різні етапи успішної атаки. По-друге, представляємо просту пряму оцінку аномалій, нову методику виявлення аномалій, яка не вимагає маркованого навчання, яка не вимагає маркованих навчальних даних і працює в непараметричному режимі. Дана методика дозволяє користувачеві легко включити знання про свою проблему в оцінки аномалій, які пряма оцінка аномалій привласнює подіям. Таким чином, в даних умовах пряма оцінка аномалій може досягти на порядок більш високої продуктивності, ніж стандартні методи виявлення аномалій, які використовують ті ж характеристики. Об'єднавши ці дві ідеї разом, представляємо розробку детектора реального часу для атак з використанням облікових даних.

1.3 Таксономія атак і набори даних

При атаці типу цільовий фішинг противник відправляє електронного листа, покликаного обманом змусити одержувача виконати небезпечне діяння. У той час як звичайні фішингові листи в основному спрямовані на отримання грошей шляхом обману довільного користувача [31, 49], то атаки цільового фішингу спрямовані на користувачів, що володіють певними привілеями. які мають будь-якого привілейованим доступом або можливостями, яких домагається противник. Така вибірковість і мотивація відрізняють цільовий фішинг від звичайних фішингових атак.

Атаки цільового фішингу використовують широкий спектр обманних стратегій і контенту. Щоб краще зрозуміти цей складний кризовий проблемне простір, представляємо таксономию, яка характеризує фішингові атаки за двома параметрами: приманка і експлойт, який використовується в атаці. Ці вимірювання відповідають двом ключовим етапам успішної атаки. Протягом усієї цієї глави ми називаємо атакуючого Меллорі, а жертву - Алісою.

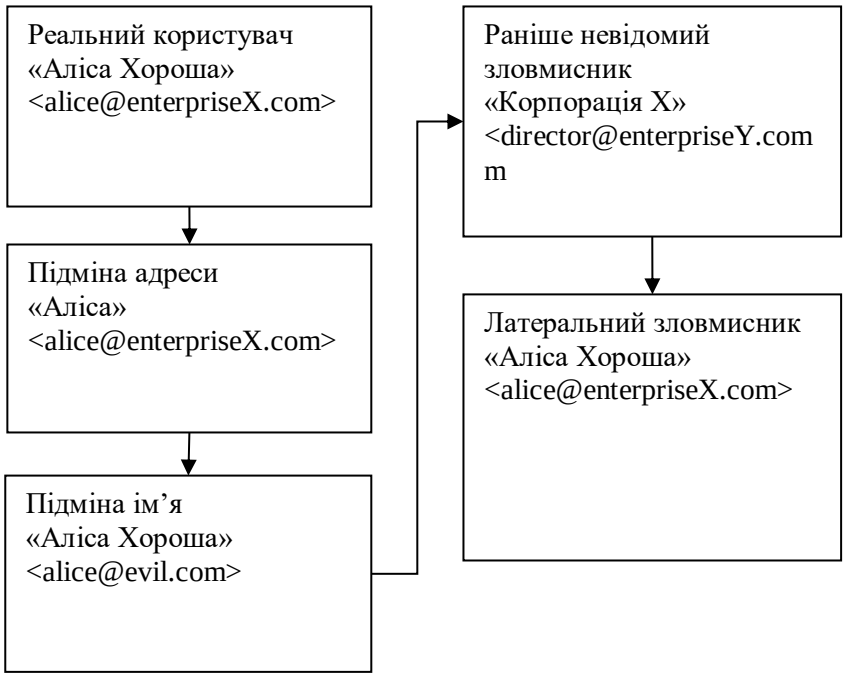


Рисунок 1.1 - Приклади чотирьох різних моделей пародіювання.

Заманювання

Атаки цільового фішингу вимагають від Меллорі надати в своєму листі почуття довіри та авторитету, щоб переконати Алісу виконати дію. Зловмисники зазвичай домагаються цього, відправляючи лист під ім'ям довіреної або авторитетної особи, та включивши в лист переконливий контент.

Модель імперсонації: Цільовий фішинг має на увазі видачу себе за іншу людину, як для того. щоб викликати довіру у одержувача, а також щоб мінімізувати ризик атрибуції і покарання. Існує існує кілька типів імперсонації (показано на рисунку 1.1):

1. Людина, підробляє адрес, готує свій лист для атаки таким чином, щоб в заголовку «Від» було вказано адресу електронної пошти довіреної особи. Зловмисник може також підробити ім'я в заголовку «Від», таким чином, що заголовок «Від» зловмисника в точності збігається з типовим заголовком «Від» справжнього користувача. DKIM і DMARC [18] блокують цю модель самозванства, дозволяючи доменам підписувати свої заголовки листів, що відправляються з криптографічним підписом, який приймають сервери та можуть перевірити за допомогою ключа перевірки на основі DNS. В останні роки ці протоколи отримують все більш широке поширення. широке поширення, причому багато великих постачальників послуг електронної пошти, такі як Gmail, впроваджують їх у відповідь на зростання числа фішингових атак [50].

2. Зловмисник підробляє ім'я в заголовку "Від" свого електронного листа, щоб воно точно збігалося з ім'ям існуючого довіреної особи (наприклад, Alice Good в "Alice Good <alice@evil.com>"). Однак в цій моделі імперсонації зловмисник не підробляє адресу своєї електронної пошти. Натомість, робить акцент на те, що одержувач побачить тільки ім'я відправника, або на те, що поштовий клієнт покаже тільки ім'я відправника. Якщо не підмінене поле «Від» електронної пошти, ця модель імперсонації обходить такі протоколи, як DKIM і DMARC.

3. Раніше невідомий зловмисник вибирає ім'я та адресу електронної пошти, щоб

помістити їх в поле "Від" фішингової пошти, які одержувач може сприйняти як ті що заслуговують на довіру або схожі на особистість реального користувача. Однак ні ім'я, ні адреса електронної пошти не збігаються з ім'ям справжнього користувача або адресу електронної пошти. Наприклад, Меллорі може вибрати підроблене ім'я Enterprise XIT Staff та адресу електронної пошти <helpdesk@enterpriseX.org> при атаці на організацію з законним доменом enterpriseX.com.

4. Латеральний зловмисник посилає свій список лист зі зламаного, але легітимного аккаунта користувача. З точки зору одержувача, метадані в фішинговому електронному листі, відправленому латерального зловмисника, не відрізняються від легітимного листи.

Набори даних

В даній роботі ми використовуємо журнали SMTP, журнали NIDS і журнали LDAP з LBNL. З міркувань конфіденційності, перш ніж надати нам доступ до даних, співробітники LBNL анонімізувати всі дані використовуючи процедуру, описану в кожному підрозділі нижче. Крім того, наші анонімізовані набори даних не містять вмісту електронних листів або веб-сторінок. У таблиці 1.1 представлена відповідна інформація цих наборів даних, а в таблиці 1.2 - розмір і тимчасові рамки наших даних.

Таблиця 1.1 - Схема кожного запису в наших джерелах даних.

Джерело даних	Поля та інформація для кожного запису журналу
Журнали SMTP	Мітка часу Від (заголовок відображається одержувачам як відправник електронної пошти) RCPT TO (список усіх одержувачів; з діалогового вікна SMTP)
Журнали NIDS	Відвідану URL -адресу Ідентифікатор журналу SMTP для найранішої електронної пошти з цією URL -адресою

	Найперше цю URL -адресу відвідували у трафіку HTTP # попередні HTTP -відвідування цієї URL -адреси # попередніх відвідувань HTTP до будь -якої URL -адреси з цим іменем хосту Ім'я хоста (повністю кваліфікований домен цієї URL -адреси) Найдавніший час відвідування будь -якої URL -адреси з такою назвою хосту
Журнали LDAP	Електронна адреса працівника Час поточного входу Час наступного входу, якщо такий є # всього входу цього співробітника # співробітників, які ввійшли в систему з міста поточного входу # попереднього входу цього співробітника з міста поточного входу

Журнали SMTP

Журнали SMTP містять анонімізовані заголовки SMTP для всіх вхідних і вихідних електронних листів протягом з 1 березня 2013 року по 14 січня 2017 року. Ці журнали містять інформацію про всі електронні листи, надіслані співробітникам організації і від них, в цілому 372 530 595 електронних листів. У другому рядку таблиці 1.1 показана відповідна інформація по заголовках, яку використовуємо для кожного електронного листа з цих журналів.

Дані були анонімізовані шляхом застосування хеша з ключем до кожного чутливого поля. Розглянемо заголовок наприклад, Аліса Хороша <ali@company.com>. "Ім'я" заголовка - це ім'я людини. (Аліса Хороша в нашому прикладі); коли ім'я людини відсутня, ми розглядаємо адреса електронної пошти як "ім'я" заголовка. Адреса "заголовка - це адреса електронної пошти: <ali@company.com>. Кожне ім'я і

кожну адресу електронної пошти хешируют окремо.

Журнали NIDS

LBNL має розподілений мережевий монітор (Zeek), який реєструє всі HTTP GET і POST запити, які перетинають його кордони. Кожен запис в журналі реєструє інформацію про запит, включаючи повний URL.

Крім того, NIDS веде звід всіх URL, що зустрічаються в тілах вхідних і вихідних електронних листів в LBNL. Щоразу, коли будь-який URL, вбудований в електронний лист, відвідується як адресата HTTP-запиту, NIDS записує інформацію про запит, включаючи URL, який був відвіданий, і запис в журналі SMTP для електронного листа, що містить отриманий URL. NIDS запам'ятовує URL-адреси протягом як мінімум одного місяця після отримання електронного листа; всі відвідування URL-адреси за протоколом HTTP зіставляються з найранішим електронним листом, що містить отриману URL-адресу.

Отримані анонімізовані журнали всіх HTTP-запитів з хеш-функцією, яка застосовується до кожної URL-адреси та його певного домену. Крім того, отримані анонімізовані журнали, що ідентифікують кожен електронний лист. URL якого був натиснутий, і анонімізована інформація про лист і URL, які показано в третьому рядку таблиці 1.1.

Журнали LDAP

LBNL використовує корпоративну пошту Gmail для управління електронною поштою своїх працівників. Щоразу, коли співробітник успішно входить в систему, Gmail записує корпоративну адресу електронної пошти користувача, час, коли стався вхід, і IP-адреса, з якого користувач пройшов аутентифікацію. З цих журналів LDAP отримали анонімізувати інформацію про сеанси входу, в яких (1) IP-адреса входу ніколи не використовувався (2) у користувача було більше 25 попередніх входів в систему, і (3) IP-адреса входу не належав мережі LBNL. В останньому рядку таблиці 1.1 показані анонімізовані дані кожного запису журналу LDAP, з тим же хешем з ключем, застосованим до таких полів, як адреса електронної пошти, задіяна при вході

в систему.

Таблиця 1.2 - Розмір даних за трьома джерелами журналів.

Проміжок часу	1 березня 2013 р. - 14 січня 2017 р
Всього листів Унікальні імена відправників (імена у «Від») Унікальні адреси відправника (адреси електронної пошти у «Від»)	372 530 595 3,415,471 4 791 624
Електронні листи з натиснутою URL -адресою Унікальні імена відправників (імена у «Від») Унікальні адреси відправника (адреси електронної пошти у «Від») Усього # натискань на вбудовані URL -адреси Унікальні URL -адреси Унікальні імена хостів	2 032 921 246 505 227,869 30,011,810 4 014 412 220 932
Вхід з нової IP -адреси # географічно розташованих міст серед усіх нових IP -адрес # електронних листів, надісланих під час сеансів звідки ввійшов співробітник нової IP -адреси	219 027 7 937 2 225 050

1.4 Виклик: Різноманітність доброякісної поведінки

У цьому підрозділі ми визначили декілька проблем, які роблять виявлення цільового фішингу особливо складним завданням. Зокрема, при роботі з реальним обсягом в мільйони листів на тиждень, різноманітність доброякісної поведінки призводить до неприпустимої кількості помилкових спрацьовувань для детекторів, які просто шукають аномальні значення заголовків.

Завдання 1: Відправники з обмеженою історією.

Природною стратегією виявлення є порівняння заголовків поточного аналізованого листа з усіма історичними заголовками листів від передбачуваного відправника. Наприклад, розглянемо спуфера імен, який намагається підробити одного з членів команди Аліси, надіславши електронного листа з заголовком Від Аліси<alice@evil.com>. Детектор на основі аномалій може визначити цю атаку, порівнявши адреса «Від» (<alice@evil.com>) листи з усіма адресами «Від» в попередніх листах з ім'ям Аліси.

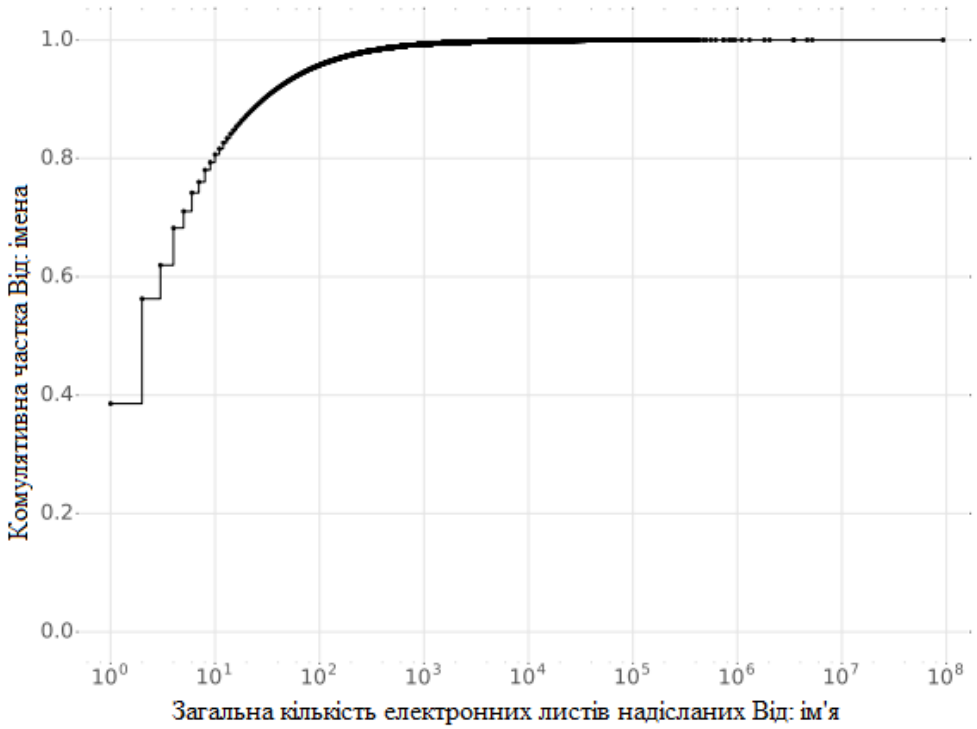


Рисунок 1.2 - Розподіл кількості відправлених листів на ім'я «Від».

Однак такий підхід не дозволить виявити іншу атаку цільового фішингу, в якій ні ім'я ні адреса в заголовку «Від» ніколи не зустрічалися раніше: Аліса <alice@evil.com> або HR-команда <hr.enterpriseX@gmail.com>. У цій раніше невидимої ситуації з атакуючим, немає ніякої історії, щоб визначити, чи є адреса «Від» аномальним.

Щоб усунути цю прогалину, детектор може відзначати всі електронні листи з новим або раніше невідомим ім'ям «Від». На жаль, на практиці такий підхід призводить до переважної числа попереджень, оскільки мільйони імен «Від» зустрічаються лише в декількох листах. На рисунку 1.2 показано розподіл кількості листів на ім'я «Від» в нашому наборі даних. Зокрема, ми виявили, що понад 60% імен «Від» відправили три або менше листів, а більше 40% імен відправили рівно один лист. Таким чином, навіть якщо організація запустила детектор для ретроспективного попередження про кожному листі з ім'ям «Від» який ніколи не був помічений раніше і не став в результаті активним і залученим відправником, це дало б більше 1,1 мільйона попереджень: відсоток помилкових спрацьовувань менше 1% на нашому наборі даних з майже 370 мільйонів листів, але все ж на порядки більше, ніж практичний обсяг попереджень.

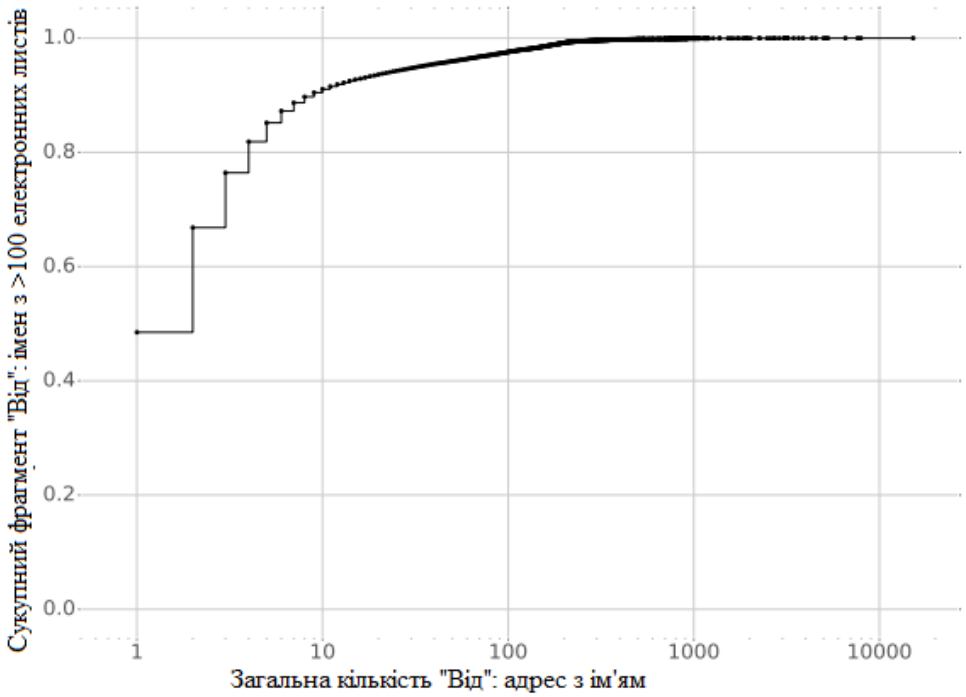


Рисунок 1.3 - Розподіл загальної кількості адрес «Від» на ім'я «Від» за всіма електронними листами, відправленим від цього імені.

Незважаючи на те, що частина цих листів з новими іменами «Від» може бути викликана спамом, співробітники служби безпеки LBNL досліджували випадкову вибірку цих листів і виявили спектр доброякісного поведінки: запрошення на входи / конференції, повідомлення про управління списками розсилки, реклама пробних версій програмного забезпечення і електронні листи служби підтримки. Таким чином, детектор, який використовує тільки традиційний підхід пошуку аномалій в значеннях заголовків, стикається з обмеженим діапазоном аномального, але доброякісної поведінки.

Завдання 2: Зміна значень заголовків

Навіть якщо ми відмовимося від виявлення атак, що виходять від раніше невидимих імен «Від» або адрес, детектор, заснований на аномаліях в заголовках, все одно зіткнемось з ще одним спектром різноманітної, доброякісної поведінки. А саме, значення заголовків для відправника часто змінюються за різними доброякісним причин. Щоб проілюструвати це, ми розглянемо всі імена «Від», які зустрічаються принаймні в 100 електронних листах і оцінимо частоту, з якою ці імена

використовують нову електронну адресу «Від» при відправці електронної пошти.

На рисунку 1.3 показано кумулятивний розподіл загальної кількості адрес електронної пошти «Від» на кожне від імені. З цього графіка видно, що навіть імена «Від» зі значною історією (ті, хто надіслали понад 100 листів) відправляють листи зі значним розкидом в значеннях заголовків: 52% цих імен відправляють листи з більш ніж однієї адреси електронної пошти «Від». Було виявилено, що 1,347,744 електронних листів містять нову адресу електронної пошти «Від», який ніколи не використовувався ні в одному з попередніх листів імені «Від». Генерування попередження для кожного з цих листів значно перевищить нашу мету - 10 попереджень в день.

Така велика кількість нових адрес електронної пошти на одне ім'я «Від» зумовлена великою кількістю різних джерел: робочі і особисті адреси електронної пошти користувача, популярні людські імена, де кожну адресу електронної пошти являє окрема людина в реальному житті (наприклад, кілька людей з одним ім’ям), опитування професійних співтовариств і специфічні для конкретної функції адреси електронної пошти (наприклад, Foo <noreply@foo.com>, Foo <help@foo.com>, Foo donate@foo.com>). Хоча може виникнути спокуса використовувати репутацію домену або схожість домену між новою адресою «Від» і попередніми адресами «Від», для відсіювання помилкових спрацьовувань, це не вдається в ряді випадків.

З огляду на поширеність електронних листів з аномальними, але доброякісними значеннями заголовків, практичний детектор явно повинен використовувати додаткові сигнали, крім значень заголовків електронної пошти. У деяких попередніх наукових роботах робилися спроби використовувати стилометричні ознаки з тіла листа для виявлення атак типу цільовий фішинг [42]; однак, як уже говорилося раніше, ці системи мають коефіцієнт помилкових спрацьовувань 1% і вище, що призведе до мільйонів помилкових спрацьовувань. У наступному розділі представляємо новий підхід, який використовує інший набір сигналів, заснований на глибинній природі атак цільового фішингу.

1.5 Опис детектора

На високому рівні детектор складається з трьох етапів, показаних на рисунку 1.4 і описаних нижче: етап вилучення ознак, етап нічної оцінки і етап генерації попереджень в реальному часі. Концептуально робота представляє дві ключові ідеї, які дозволяють нашому детектору виявляти широкий спектр атак, при цьому практичний рівень помилкових спрацьовувань в 200 разів нижче. По-перше, детектор витягує два набори ознак, основаних на репутації, які незалежно націлені на два ключових етапів атаки цільового фішингу, визначених у нашій таксономії атак. По-друге, вводимо новий метод виявлення аномалій без контролю, який дозволяє нашому детектору автоматично ранжувати набір немаркованих подій і вибирати найбільш підозрілі події для розгляду командою безпеки. Спочатку обговоримо кожен з цих елементів, а потім покажемо, як об'єднати їх для створення детектора в режимі реального часу.

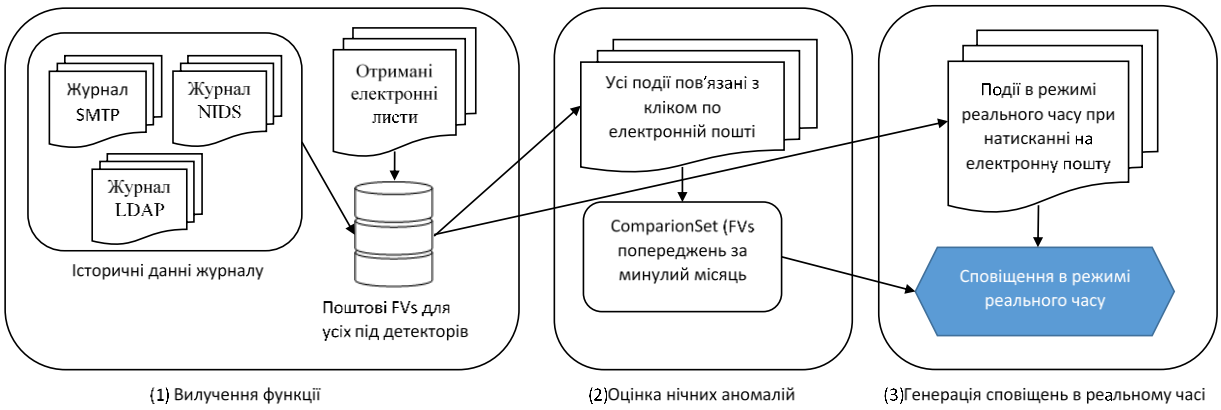


Рисунок 1.4 - Огляд детектора реального часу, який використовує результати нічного пакетного завдання для аналізу в реальному часі векторах ознак (FVs – вектори ознак).

Обмеження стандартних методів виявлення

Після того як наш детектор витягнув характеристики для кожної події, пов'язані з кліком по електронній пошті, йому необхідно вирішити, які з них повинні

викликати сигнал тривоги для команди безпеки. Спочатку обговоримо три природних, але в кінцевому рахунку неефективних підходу до визначення того, які події слід повідомляти. Та представимо нову техніку, DAS, яку детектор використовує для подолання обмежень цих канонічних підходів.

Ручні порогові значення: Найпростіший підхід до класифікації полягає в ручному виборі порога для кожної ознаки і видачі попередження, якщо всі значення ознаки нижче порога. Можна використовувати знання домену про кожну ознаку, щоб вгадати поріг для кожного вимірювання ознаки: наприклад, при шпигунських атаках будуть використовуватися URL-адреси, домену яких має менше п'яти відвідувань або був вперше відвіданий менше п'яти днів назад. На жаль, цей підхід за своєю суттю є довільним, оскільки ми не знаємо істинного розподілу значень ознак для атак з використанням цільового фішингу. Таким чином, цей спеціальний підхід може легко пропустити атаки і не дає критеріїв відбору, які можна було б використовувати на різних підприємствах.

Контрольоване навчання: Велика кількість літератури з виявлення атак, починаючи з класифікації спаму і закінчуючи попередніми роботами по боротьбі зі цільовим фішингом, в значній мірі спирається на алгоритми машинного навчання з контролем. Однак в нашому випадку ці методи мають серйозні недоліки.

Для точної класифікації нових подій методи контрольованого навчання вимагають маркованого навчального набору даних, який відображає діапазон можливих значень шкідливих і доброякісних ознак. На жаль, в нашому контексті нам не вистачає досить великого набору помічених атак для навчання функціональної моделі контрольованого навчання; атаки цільового фішингу відбуваються з низькою частотою, і їх вкрай складно виявити, коли вони відбуваються.

Крім того, в нашій ситуації спостерігається крайня незбалансованість класів: через брак даних про дані атаки в навчальному наборі буде набагато більше доброякісних примірників, ніж шкідливих. Методи супервізії часто вимагають збалансованого набору даних; класифікатори, навчені на сильно незбалансованих

даних, часто вчаться завжди передбачати клас більшості (тобто класифікувати всі як доброякісні), патологічно перебудовуються під випадкові характеристики класу меншини або створюють занадто слабкі межі прийняття рішення і генерують непомірно велика кількість помилкових спрацьовувань [51]. Хоча співтовариство машинного навчання вивчило ряд методів вирішення проблеми незбалансованих навчальних даних [51, 52], таких як недобір перепредставленного класу або синтетична генерація зразків для недопредставлених класів, ці методи не підходять для дисбалансу порядку мільйон до одного.

Стандартне виявлення аномалій: В якості альтернативи детектор може використовувати неконтрольовані або напівнеконтрольовані методи виявлення аномалій. Хоча існує ряд таких методів, включаючи методи оцінки щільності, такі як моделі гауссових сумішей [53] і кластеризація або методи на основі відстаней, такі як «k-nearest-neighbor» (kNN) [54], ці класичні методи страждають від трьох обмежень.

По-перше, в ряді умов безпеки скалярні характеристики часто мають спрямованість своїх значень; дійсно, всі наші характеристики володіють цією властивістю. Наприклад, чим менше відвідувань у домені, тим більш підозрілим він є; незвично мала кількість відвідувань є підставою для підозр, а незвично велика немає. Стандартні методи виявлення аномалій не включають поняття асиметрії або спрямованості в свої обчислення. Наприклад, методи виявлення аномалій на основі щільності такі як оцінка щільності ядра (KDE) та GMM, підганяють розподіл ймовірності до даних і попереджають про події з найменшою вірогідністю. Події, які мають статистично екстремальні, але доброякісні значення характеристик матимуть дуже низьку ймовірність виникнення, що призведе до великої кількості непотрібних попереджень.

По-друге, стандартні методи виявлення аномалій часто розглядають подію як аномальну, навіть якщо лише один або кілька ознак події є статистично аномальними. Однак в наших умовах ми очікуємо, що атаки будуть аномальними і підозрілими за всіма параметрами. Отже, в наших умовах класичні методи будуть генерувати багато

помилкових попереджень для подій, які є аномальними тільки в декількох вимірах.

По-третє, класичні методи є параметричними: вони або припускають, що дані походять з певного базового розподілу, або містять ряд параметрів, які повинні бути правильно задані для отримання прийнятної продуктивності. GMM припускає, що дані надходять в суміші гауссових розподілів, KDE має параметр пропускної здатності, який вимагає настройки розробником, а kNN вимагає від розробника вибору значення k (кількість найближчих сусідів / найбільш схожих подій, які повинні бути правильно встановлені розробником). Ці вимоги проблематичні для виявлення цілового фішінгу, оскільки ми не знаємо істинного розподілу атакуючих і доброякісних електронних листів (наприклад, дійсний розподіл може бути не гауссовским), і у нас немає надійного способу вибору параметрів.

Спрямована оцінка аномалій (DAS)

З огляду на обмеження традиційних методів виявлення, ми представляємо просту і загальну техніку для автоматичного вибору найбільш підозрілих подій з немаркованого набору даних. Дану техніку називаємо спрямованою оцінкою аномалій (Directed Anomaly Scoring, DAS). На високому рівні DAS ранжирує всі події, порівнюючи, наскільки підозрілим є кожна подія по відношенню до всіх інших. Після ранжирування всіх подій DAS просто вибирає N найбільш підозрілих подій, де N - бюджет команди безпеки на оповіщення.

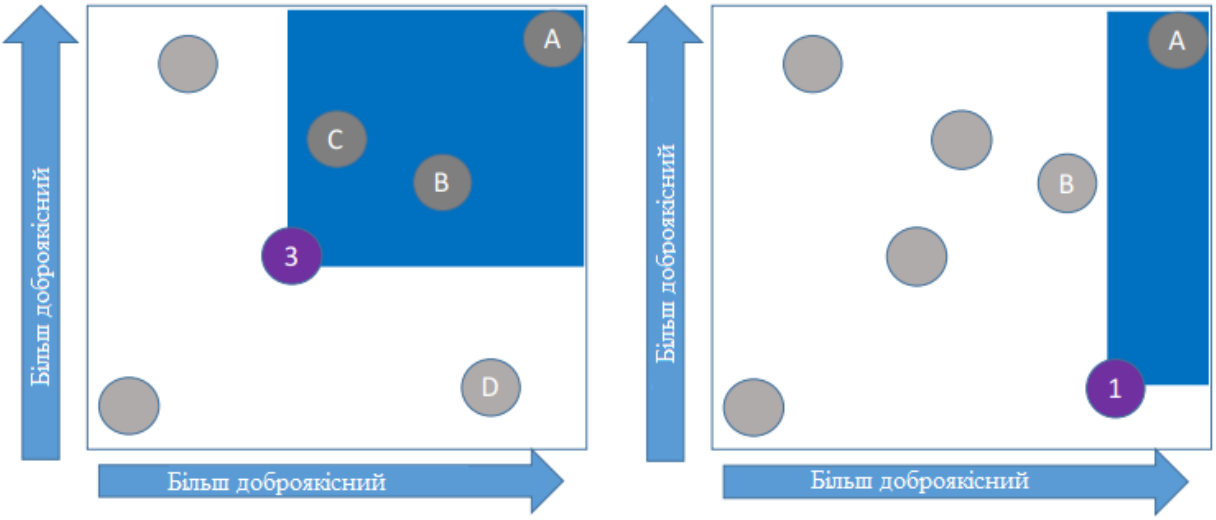


Рисунок 1.5 - Приклади оцінки DAS для подій в двовимірному просторі ознак.

Формально, ми ідентифікуємо кожна подія з його вектором ознак $E \epsilon R^d$. Вважаємо, що подія E менш підозріла ніж подія E′, записуємо $E \succcurlyeq E'$ і Ei $\leq E_i'$, для всіх i =1, 2,…, d. Тоді, оцінка події E - це кардинальність безлічі $\{E' : E \succcurlyeq E'\}$. (Для простоти припускаємо, що меншізначення ознаки є більш підозрілими в кожному вимірі; для вимірювань, де вірно зворотне,ми замінюємо компаратор ≤ на ≥. У додатку A.1 наведено компаратори, які ми використовуємо для кожної ознаки).

Алгоритм 1. Підрахунок і вибір попереджень в DAS

Оцінка (E, L):

1: для кожна події X в L зробити:

2: якщо E більш підозріло, ніж X в кожному вимірі:

3: збільшуємо оцінку E на одиницю.

AlertGen (L (список подій), N):

1: для кожної події E в L зробити:

2: Оцінка (E, L)

3: Відсортувати L за кількістю балів кожної події

4: повертає N подій з L з найбільшою кількістю балів

Алгоритм 1 показує процедуру підрахунку балів і генерації попереджень за

допомогою DAS. Конкретно, DAS спочатку присвоює бал аномалії для кожної події E, обчислюючи загальна кількість інших подій де вектор ознак E принаймні настільки ж підозрілий, як і інші події в кожному вимірі ознак. Таким чином, оцінка E підраховує, скільки подій, принаймні, настільки ж підозрілі, як і воно; події з більш високими оцінками є більш підозрілими, ніж події з більш низькими балами. На Рисунку 1.5 представлено декілька прикладів обчислення балів DAS. Після оцінки кожної події алгоритм просто сортує всі події і виводить N подій з найбільшою кількістю балів.

DAS добре підходить для цілого ряду проблем виявлення безпеки, де атаки можуть бути охарактеризовані комбінацією числових і логічних ознак, як в нашому прикладі з цільовим фішингом; досягає на порядки кращих результатів, ніж класичні методи виявлення аномалій, оскільки використовує знання про те, які області простору ознак є найбільш підозрілими; зокрема, вона долає всі три обмеження класичних методів.

1.6 Помилкові спрацьовування і тягар сповіщень

При щоденному бюджеті в 10 повідомлень в день наш детектор досяг середньої частоти помилкових спрацьовувань в 0,004%. Щоб оцінити фактичне щоденне навантаження, ми запустили детектор в реальному часі на сто випадково вибраних днів в нашому наборі даних і підрахували загальну кількість попереджень, які він генерував в кожен день, як показано на малюнку 1.6. З цієї гістограми видно, що, хоча наш детектор іноді генерує сплески, що перевищують наш цільовий бюджет, в переважній більшості днів (80%) він генерує 10 або менше попереджень в день; майже в 20% днів він не генерує ніяких попереджень.

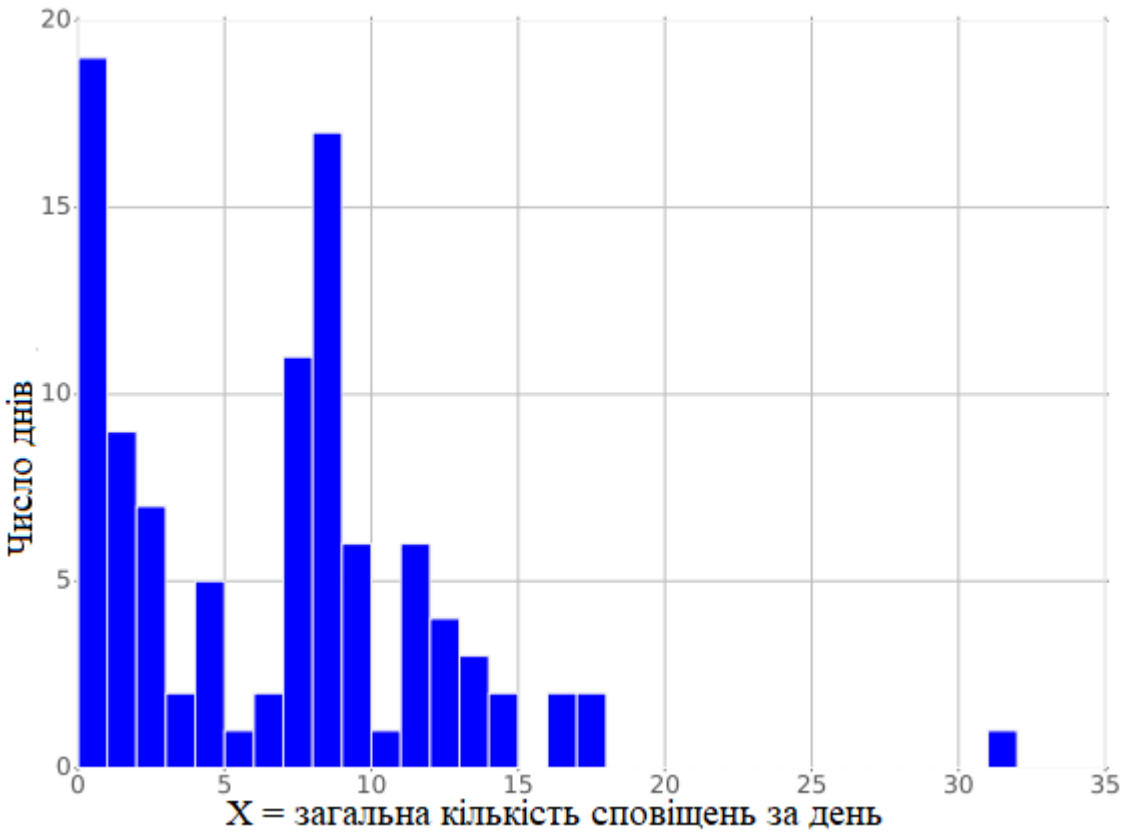


Рисунок 1.6 - Гістограма загальної кількості щоденних попереджень, що генеруються детектором реального часу в 100 випадково вибраних днів.

Під час ручного дослідження 15 521 попередження, створеного в ході нашого процесу маркування істини співробітники служби безпеки LBNL відстежували,

скільки часу у них пішло на розслідування цих попереджень. Дивно, але співробітники служби безпеки LBNL повідомили, що один аналітик може обробити весь місячний обсяг попереджень менш ніж за 15 хвилин; і, таким чином, в середньому на аналіз одного дня йде менше однієї хвилини сповіщень.

Такий швидкий час обробки виник тому, що аналітики змогли розробити двопрохідний робочий процес, який дозволив їм швидко відкидати більше 98% повідомлень під час швидкого сортування, зі швидкістю 2 секунди на оповіщення; а потім проводити більш глибокий аналіз з 2% повідомлень, зі швидкістю 30 секунд на оповіщення. Перший прохід настільки швидкий, тому що для переважної більшості попереджень детектора аналітик міг швидко визначити, чи представляє лист правдоподібну загрозу, перевіривши рядок «Тема», рядок «Від» і URL-адресу листа. Для більш ніж 98% попереджень ця трійка інформації вказувала на те, що лист з високою ймовірністю не містить атаку з використанням цільового фішингу облікових даних. Наприклад, листи з такими темами, як "Ніколи більше не втрачайте ключі, гаманець або гаманець!" і "УВАГА: Ваші проблеми зі шлунком нарешті пояснені. Дивіться відео тут" точно не є атаками цільового фішингу.

У той час як більш трудомісткі 2% попереджень містили в основному помилкові спрацьовування, аналітики виявили два цікавих класи попереджень. По-перше, крім виявлення цільового фішингу, наш детектор виявив 41 електронний лист від "звичайних" фішингових кампаній. Аналітики розпізнали звичайний фішинг від цільового фішингу, перевіривши, чи містилося в електронному листі та HTTP-відповіді, отриманому по URL-адресі, вміст, спеціально спрямований на LBNL. По-друге, за іронією долі, детектор видав 40 попереджень, в яких людина, що натисне на посилання в листі, що не є одним з одержувачів листа, а скоріше співробітником служби безпеки LBNL. Ці кліки були частиною звичайних розслідувань, що проводяться співробітниками служби безпеки LBNL; наприклад, у відповідь на повідомлення користувача про підозрілу листі.

Порівняння виявлення аномалій

У розділі 1.5 представили DAS - нову просту техніку виявлення аномалій на немаркованих даних. Тепер оцінимо ефективність DAS в порівнянні з традиційними методами виявлення аномалій без спостереження. методами виявлення аномалій без спостереження.

Було протестовано три найпоширеніші методи виявлення аномалій з літератури по машинному навчання: Kernel Density Estimation (KDE), Gaussian Mixture Models (GMM) і k-Nearest Neighbors (kNN) [53]. Щоб порівняти ефективність виявлення в реальному часі кожного з цих класичних методів з ефективністю DAS в реальному часі, ми запустили кожен з цих класичних методів, використовуючи ті ж процедури навчання і оцінки, які ми використовували для оцінки нашого детектора в реальному часі. Зокрема, з огляду на дату кожної з 19 атак і моделі імперсонації, ми витягли однакові точні значення ознак для всіх подій, пов'язаних з кліком по електронній пошті, які відбулися протягом тридцятиденного періоду, що закінчується датою атаки; тридцятиденний період збігається з періодом, який ми використовували для створення набору порівняння детектора. Потім нормалізували значення цих характеристик і перевірили кожен з трьох класичних методів виявлення аномалій на цьому наборі подій "клік в листі" для кожної дати атаки. Для кількісного порівняння розрахували (1) кількість атак, які були б виявлені кожним класичним методом, якби він використовував той же бюджет, що і наш детектор реального часу, і (2) щоденний бюджет, який знадобився б класичним методом для виявлення всіх атак , які виявив детектор на основі DAS.

Як і інші методи машинного навчання, ці класичні алгоритми вимагають від користувача установки різних гіперпараметрів, які впливають на продуктивність алгоритму. Для нашої оцінки ми протестували кожен класичний метод при різних значеннях гіперпараметрів і представили результати для тих гіперпараметрів, які дали найкращі результати (тобто порівняння DAS з найкращого версією цих класичних методів).

У таблиці 1.3 наведені результати цього експерименту. При однаковому щоденному бюджеті в 10 повідомлень в день, всі три традиційних методу виявили менше 25% атак, виявлених DAS. Більш того, для того, щоб класичний метод (KDE), що показав найкращі результати, виявив стільки ж атак, скільки і DAS, йому буде потрібно щоб щоденний бюджет був напорядок вище ніж 10 попереджень в день.

Таблиця 1.3 - Порівняння класичних методів виявлення аномалій з DAS на тому ж наборі даних і тих же характеристиках.

Алгоритм	Виявленно	Щоденний б'юджет
kNN	3/19 17/19	10 2,455
GMM	4/19 17/19	10 147
KDE	4/19 17/19	10 91
DAS	17/19	10

Щоб проілюструвати, чому стандартні методи без спостереження працюють так погано, на двох графіках на рисунку 1.7 показані характеристики репутації відправника для випадкової вибірки з 10 000 подій, пов'язаних з кліками латеральних зловмисників в електронній пошті. Лівий графік показує значення характеристик для фактичних попереджень, згенерованих нашим детектором на основі DAS (червоним кольором), а правий графік показує значення характеристик для попереджень, відібраних KDE з використанням того ж бюджету, що і наш детектор. KDE вибирає масу точок в правому верхньому куті, що ілюструє одне з обмежень стандартних методів, що обговорювалися в розділі 1.5: вони не беруть до уваги спрямованість значень ознак. Оскільки надзвичайно великі значення ознаки зустрічаються нечасто, KDE оцінює ці події як високо аномальні, навіть якщо вони відповідають доброякісним сеансам входу в систему, коли користувач випадково увійшов в

систему з нового IP-адреси в житловому місті неподалік від LBNL.

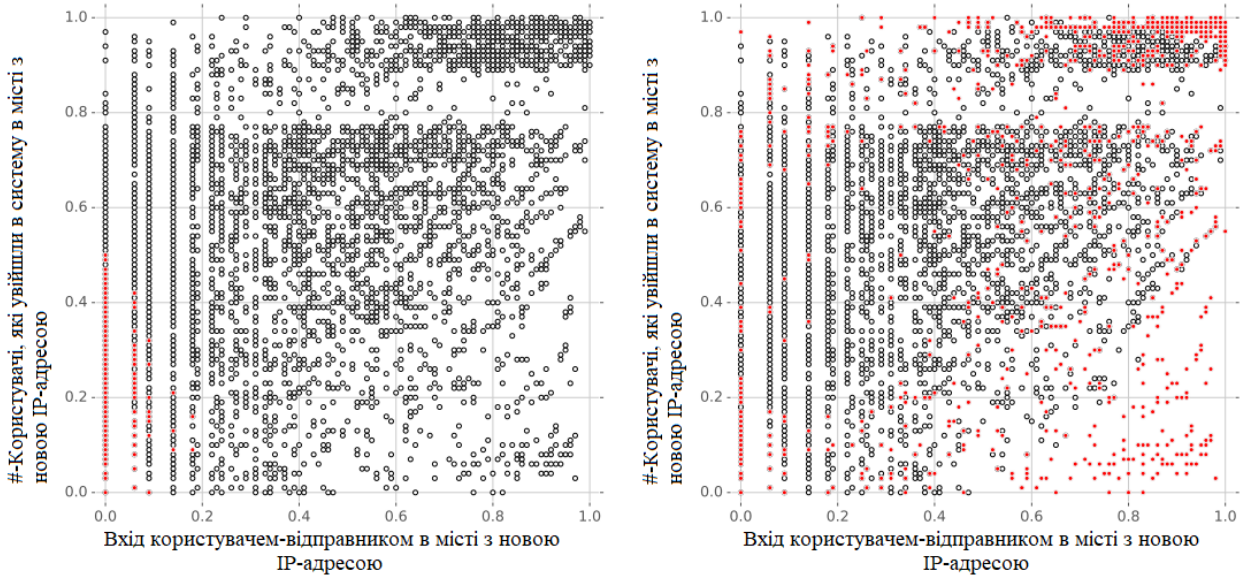


Рисунок 1.7 - Обидва графіка показують значення характеристик репутації відправника (масштабовані між [0, 1]) випадкової вибірки з 10 000 подій, пов'язаних з кліками латеральних зловмисників в електронній пошті. Заповнені червоні точки позначають події, які викликали оповіщення в межах денного бюджету DAS (лівий малюнок) і KDE (правий малюнок).

Вибір KDE цих доброякісних логінів ілюструє ще одне обмеження стандартних методів: вони часто вибирають події, які є аномальними тільки в одному вимірі, не беручи до уваги наші знання про те, що атака буде аномальної у всіх вимірах. Незважаючи на те, що в правому нижньому кутку представлені логіни співробітників, коли кілька інших співробітників увійшли в систему з того ж міста, вони не викликають підозр, оскільки вони відповідають звичайному законному процесу входу в систему: доброякісні входи віддалених співробітників, які живуть і працюють в містах, розташованих далеко від головного кампусу LBNL. Таким чином, DAS може значно перевершити стандартні методи виявлення аномалій без спостереження, тому що це дозволяє нам включити знання домену знання про особливості при прийнятті рішень DAS.

2 ВИЯВЛЕННЯ І РОЗУМІННЯ ПОВЕДІНКИ ЗЛОВМИСНИКІВ

У цьому розділі представимо огляд двох категорій пов'язаних робіт. По-перше, ми вивчаємо літературу за влучним висловом і пом'якшення наслідків бічних фішингових атак, коли зловмисник намагається поширитися з одного зламаного облікового запису підприємства на інші облікові записи співробітників; ця робота пов'язана з висновками, представленими в розділі 3. Далі ми розглянемо попередні роботи щодо запобігання переходу зловмисників з однієї зламаної корпоративної машини на інші внутрішні машини в організації, що пов'язано з роботою, яку ми описуємо в розділі 4.

2.1 Пов'язані роботи і передумови

Підприємства стикаються з цілою низкою атак з боку наполегливих і досвідчених противників. Тому жоден окремий спосіб захисту не може бути успішним поодинці проти цього складного ландшафту загроз. Швидше, співтовариство безпеки вже давно виступає за підхід "захист в глибину": ідея про те, що користувачі і організації повинні об'єднати набір взаємодоповнюючих засобів захисту для відбиття атак.

Шкідлива активність у зламаних облікових записах електронної пошти

Виявлення бічних (латеральних) фішингових атак: У великій літературі пропонується безліч методів виявлення традиційних фішингових атак [28, 31, 36, 39, 55,67], а також більш складних атак цільового фішингу [37, 40, 41, 42, 56]. Однак порівняно невелика кількість цих робіт присвячено бічним фішинговим атакам, на відміну від більш традиційних фішингових атак, що виходять з зовнішніх акаунтів.

При ознайомленні, як використовувати метрики соціальних графів для виявлення шкідливих електронних листів, відправлених з зламаних акаунтів у підході автора метод виявляє зламані облікові записи з коефіцієнтом помилкових спрацьовувань в межах 20-40%[57]. На жаль, на практиці багато організацій

обробляють десятки тисяч електронних листів від співробітників в день, тому показник помилкових спрацьовувань в 20% призведе до тисяч помилкових попереджень кожен день. IdentityMailer, запропонований Стрінгіні(Stringhini) та іншими [42], виявляє бічні фішингові атаки шляхом навчання поведінкових моделей на основі тимчасових моделей, метаданих і стильометрія для кожного користувача. Якщо новий лист відхиляється від поведінкової моделі співробітника, їх система відзначає його як атаку. Незважаючи на перспективність, їх підхід дає коефіцієнт помилкових спрацьовувань в діапазоні 1-10%, що знову ж таки не має сенсу на практиці, з огляду на великий обсяг доброякісних листів і низьким базовим рівенем фішингу. Крім того, їх система вимагає навчання поведінкової моделі для кожного співробітника, що тягне за собою дорогий технічний борг для роботи в масштабах компанії.

Характеристика скомпрометованих хмарних акаунтів: Хоча попередні роботи показують, що зловмисники часто використовують фішинг для компрометації облікових записів, і що зловмисники іноді здійснюють фішинг з цих зламаних облікових записів, мало хто вивчав природу бічного фішингу більш глибоко і масштабно. Вивчаючи вибірку фішингових листів, веб-сторінок і зламаних облікових записів з джерел даних Google, одне з попередніх досліджень захоплення акаунтів показало, що зловмисники часто використовують ці зламані облікові записи для відправки фішингових листів, веб-сторінок і зламаних аккаунтів[58]. Однак вони прийшли до висновку, що автоматичне виявлення таких атак є складним завданням. Онаолапо(Onaolapo) і ін. вивчали що зловмисники роблять із захопленими обліковими записами [59], але вони не спостерігали випадків бічних фішингових атак. Дослідження зламаних облікових записів Twitter, проведене окремо від облікових записів електронної пошти, показало, що інфекції, мабуть, поширюються латерально через соціальну мережу. Однак їх набір даних не дозволяв безпосередньо спостерігати сам вектор бічної атаки [60], а також не давав уявлення про характеристики зламаних корпоративних акаунтів.

2.2 Захист від бічного переміщення між машинами

У попередніх роботах, розглянутих досі, основна увага приділялася тому, як зловмисники можуть використовувати зламану електронну пошту або аккаунт в соціальних мережах. У решти цього розділу ми розглянемо попередні роботи, які досліджують, як організації можуть зменшити шкоду, яку може завдати зловмисник через зламану внутрішню машину. З огляду на методи і результати, які представимо в наступних розділах розглядаємо перший і останній напрям попередньої роботи як додаткові напрямки до даної роботи. Зокрема, робота в главі 4 зосереджена на розробці виявлення атак бічного переміщення. Перший напрямок, що визначає активне посилення і зміну політики безпеки, дозволяє організації впровадити ефективну політику найменших привілеїв і визначити особливо небезпечні машини, які вимагають додаткового моніторингу [61, 62, 63, 64]. Хоча ці превентивні заходи роблять бічне переміщення більш складним, вони часто не можуть повністю усунути всі можливі шляхи переміщення. Дійсно, модель загроз (розділ 4.1) передбачає, що організації використовують ці підходи для реалізації і зосереджені на виявленні поширених типів атак бічного переміщення, які, тим не менш, можуть відбутися. Третій напрям суміжних робіт - судове розслідування і реагування на відому атаку, передбачає, що організація вже визначила існування атаки; тобто у підприємства є ефективна стратегія виявлення, на якій ми зосередимося в наступних розділах. Наприклад, організація може використовувати методи криміналістики, розроблені в попередній роботі, в поєднанні з Hopper, новою системою, яку представляємо в розділі 4, для ефективного усунення наслідків атаки і пом'якшення шкоди, заподіяної атакою бічного переміщення, виявленої нашим підходом.

Попередні роботи по виявленню бічного переміщення були зосереджені на моделюванні внутрішніх входів в систему як графа переміщень між машинами, а потім на застосуванні традиційних методів машинного навчання, таких як наприклад, виявлення аномалій, для виявлення атак [65, 66, 67, 68, 69, 70, 71]. Кент і ін. [66] пропонують підхід з контрольованим навчанням для виявлення скомпрометованих

облікових даних користувача шляхом навчання моделі логістичної регресії для виявлення випадків, коли обліковий запис отримує доступ до незвичайного набору машин; їх класифікатор досягає істинно позитивного результату в 28% і помилково зазначає 1 з кожних 800 користувачів як скомпрометованих. Лю і ін. [67] представили Latte, специфічну для Windows систему для виявлення бічного переміщення, яка використовує набір жорстко закодованих кодів подій Windows для виявлення аномальних двоходових шляхів входу в систему, які також включають операцію віддаленого виконання файлу після останнього переходу. log2vec [68] - це структура для перетворення необроблених даних журналу в уявлення ознак графа, заснована на поєднанні правил ручної кореляції і методів вбудовування графа. Аналогічним чином, Боуман і ін. пропонують несамостійний конвеєр навчання графів, який неявно групує схожих користувачів і машини на основі їх попередніх взаємодій при вході в систему, а потім попереджає про будь-яких аномальних входах. На 18-денному тестовому наборі даних їх підхід дозволяє виявити 85% шкідливих логінів, створених в ході однієї вправи "червоної команди", при коефіцієнті помилкових спрацьовувань 0,9%.

Серед найбільш ефективних попередніх робіт Сіадат і Мемон (Siadatiand, Memon)пропонують систему виявлення, яку називають SAL, оскільки вона виявляє структурно аномальні логіни [71]. Оцінюючи свій підхід на даних реального підприємства за один місяць, вони показали, що SAL може виявити 82% випадково згенерували логінів атак при коефіцієнті помилкових спрацьовувань 0,3%. Хоча попередні роботи забезпечують хороші відправні точки для виявлення, навіть кращі з цих систем генерують занадто багато помилкових спрацьовувань для практичного використання, і оцінюють свій підхід на невеликому наборі абстрактних, нереалістичних атак (наприклад, випадково згенерований шлях атак). Як конкретне порівняння опишемо роботу SAL на нашому наборі даних в розділі 4.5.

3 ВИЯВЛЕННЯ ТА ХАРАКТЕРИСТИКА БІЧНИХ АТАК

3.1 Історія питань

В цілому, гучне освітлення цільових фішингових атак на великі організації, такі як Google, RSA і Демократичний національний комітет, записало і сформувало ментальні моделі фішингових атак на підприємства [5, 72, 73]. У цих гучних випадках, а також а також у багатьох інцидентах цільового фішингу, обговорюваних в науковій літературі [46, 74, 75], атаки відбуваються з зовнішніх акаунтів, створених супротивниками з національних держав, які спритно створюють або підробляють ім'я та адресу електронної пошти фішингового аккаунта, щоб він був схожий на відомого і легітимного користувача. Однак в останні роки як в промисловості [76, 77, 78], так і в наукових колах [42, 58, 59, 79] відзначається появу і зростання бічних фішингових атак: нова форма фішингу, яка спрямована на різні організації та вже заподіяла фінансових збитків в мільярди доларів [80]. При бічній фішинговій атаці противник використовує зламаний обліковий запис підприємства, щоб відправляти фішингові листи новому колу одержувачів. Ця атака особливо підступна, оскільки зловмисник автоматично отримує вигоду від неявної довіри до зламаного облікового запису: довіру як з боку одержувачів, так і звичайних систем захисту електронної пошти.

Для виконання роботи, описаної в цьому розділі, було розглянуто роботи Barracuda Networks, великої компанії, що займається питаннями безпеки, з метою розробки нових методів виявлення з використанням набору даних історичних електронних листів і повідомлень про інциденти від 92 організацій, які є активними клієнтами Barracuda Networks. Ці організації надали компанії Barracuda дозвіл на доступ до поштових скриньок співробітників Office 365 з метою дослідження і розробки засобів захисту від бічного фішингу. Відповідно до політики Barracuda, всі отримані електронні листи зберігаються в зашифрованому вигляді, і клієнти мали можливість в будь-який момент відкликати доступ до своїх даних.

З огляду на делікатність даних тільки уповноважені співробітники компанії

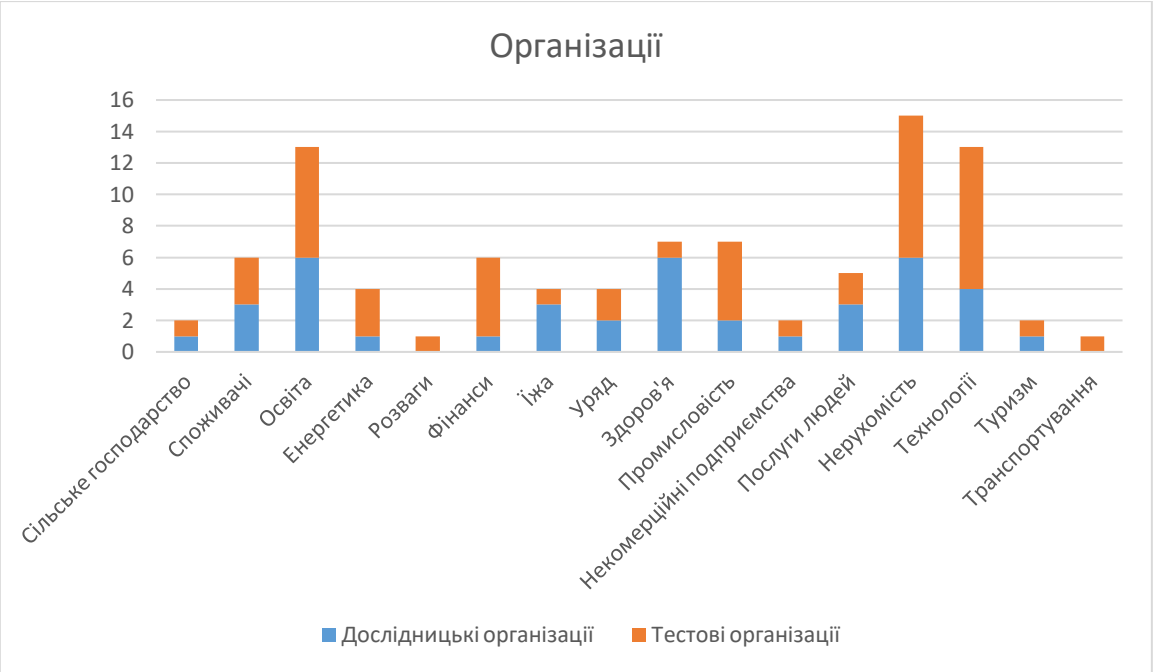
Barracuda в період проведення дослідження мали доступ до даних. Ніяка особиста ідентифікує інформація або конфіденційні дані не були передані нікому зі співробітників компанії Barracuda.

3.2 Дані

У цьому підрозділі використовуємо набір даних, що складається з електронних листів, відправлених співробітниками 92 англомовних організацій; 23 організації були взяті з випадкової вибірки підприємств, в яких були повідомлення про бічний фішинг, а 69 були взяті з випадкової вибірки всіх організацій. Серед цих підприємств 25 організацій мали менше 100 облікових записів користувачів, 34 - від 101 до 1000 облікових записів, а 33 - понад 1000 облікових записів. Нерухомість, технології та освіту склали три найбільш поширені галузі в нашому наборі даних - 15, 13 і 13 підприємств відповідно; на діаграмах 3.1 і 3.2 показано розподіл секторів економіки і розмірів організацій набору даних в розбивці на дослідні та тестові організації.

Схеми

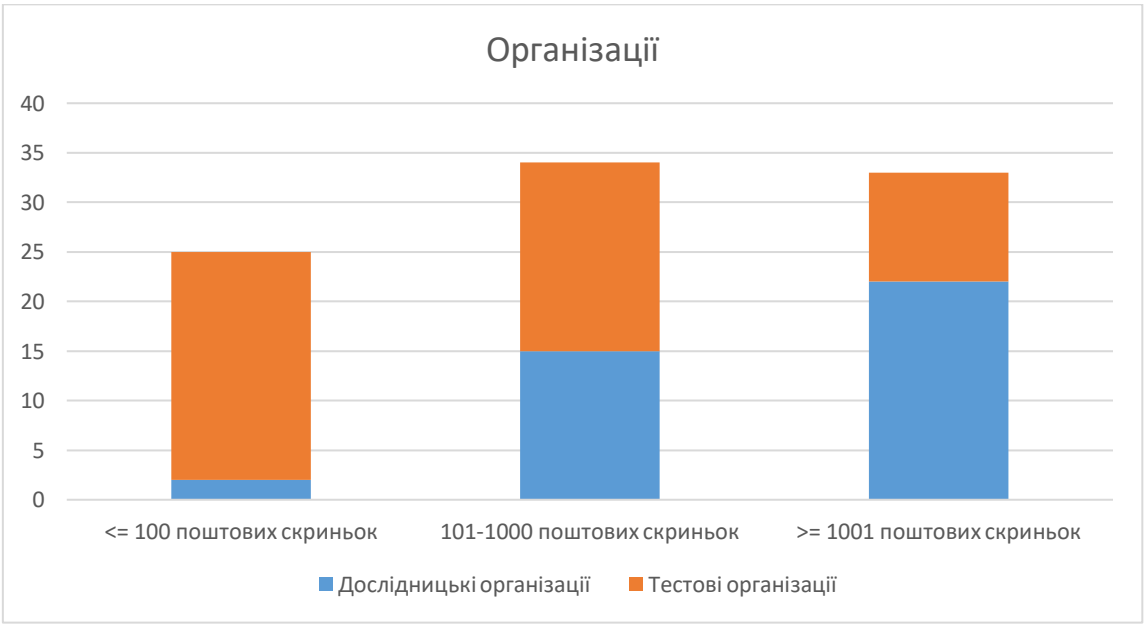
Організації в нашому наборі даних використовують Office 365 в якості постачальника послуг електронної пошти. На високому рівні кожен об'єкт електронної пошти містить: унікальний ідентифікатор Office 365; метадані електронного листа (інформація заголовка SMTP), яка описує такі властивості, як тимчасова мітка відправлення листа, одержувачі, передбачуваний відправник і тема та тіло листа: зміст електронного повідомлення в повному форматі HTML. У документації Office 365 описана повна схема кожного об'єкта електронної пошти [81]. Крім того, для кожної організації у нас є набір перевірених доменів: домени, про володіння якими організація заявила.



Діаграма 3.1 - Розбивка економічних секторів в 52 дослідницьких організаціях нашого набору даних в порівнянні з 40 тестових організацій.

Базова істина

Набір бічних фішингових листів було отримано з двох джерел: (1) повідомлення про атаки, отримані з організації Barracuda, а також повідомлення про атаки, передані користувачами в свою організацію, і (2) повідомлення, відмічені детектором, які вручну перевірили і позначили перед включенням.



Діаграма 3.2 - Розподіл обсягів організацій.

Щоб вручну позначити лист як фішинговий чи ні, вивчили вміст повідомлення, метадані Office 365 і заголовки інтернет-повідомлень [82], щоб визначити, чи містить лист фішинговий контент, і чи нанадісланий лист зі зламаного облікового запису. Наприклад, якщо метадані Office 365 показують, що копія листа перебувала в папці "Відправлені" співробітника, або якщо заголовки показували, що лист лист пройшов відповідні перевірки SPF або DKIM [17], то позначили лист як бічний фішинг. У розділі А.2 додатку докладно описана наша процедура маркування.

Крім того, для невеликої вибірки URL-адрес в цих бічних фішингових листах співробітники Barracuda відкрили фішингову URL-адресу в браузері, встановленому на віртуальній машині, щоб краще зрозуміти кінцеву мету атаки. Щоб мінімізувати потенційну шкоду і побічні ефекти, ці співробітники відвідували тільки ті фішингові URL, які не містили унікальних ідентифікаторів. Більшість фішингових URL, які досліджували, вели на проміжну веб-сторінку SafeBrowsing, що, ймовірно, відображає використання історичних листів, а не те, з чим користувачі могли б зіткнутися в даний час. Однак більш шкідливі URL постійно вели на фішингові веб-сайти, призначені для отримання облікових даних і виглядають як легітимна сторінка входу в Office 365; на рисунку 3.1 показаний анонімізований приклад одного з фішингових веб-сайту.

В цілому набір даних містить 1 902 бічних фішингових листів (унікальних по темі, відправнику і часу відправлення), відправлених 154 зламаними обліковими записами співробітників з 33 організацій. 1 694 з цих листів повідомили користувачі, а інші були виявлені детектором (розділ 3.3); детектор також виявляє багато атак, про які повідомили користувачі. Серед атак, про які повідомили користувачі, 40 листів (з 12 зламаних акаунтів) містили підроблений грошовий переказ або шкідливе вкладення, а в інших 1 862 листах використовувався шкідливий URL.

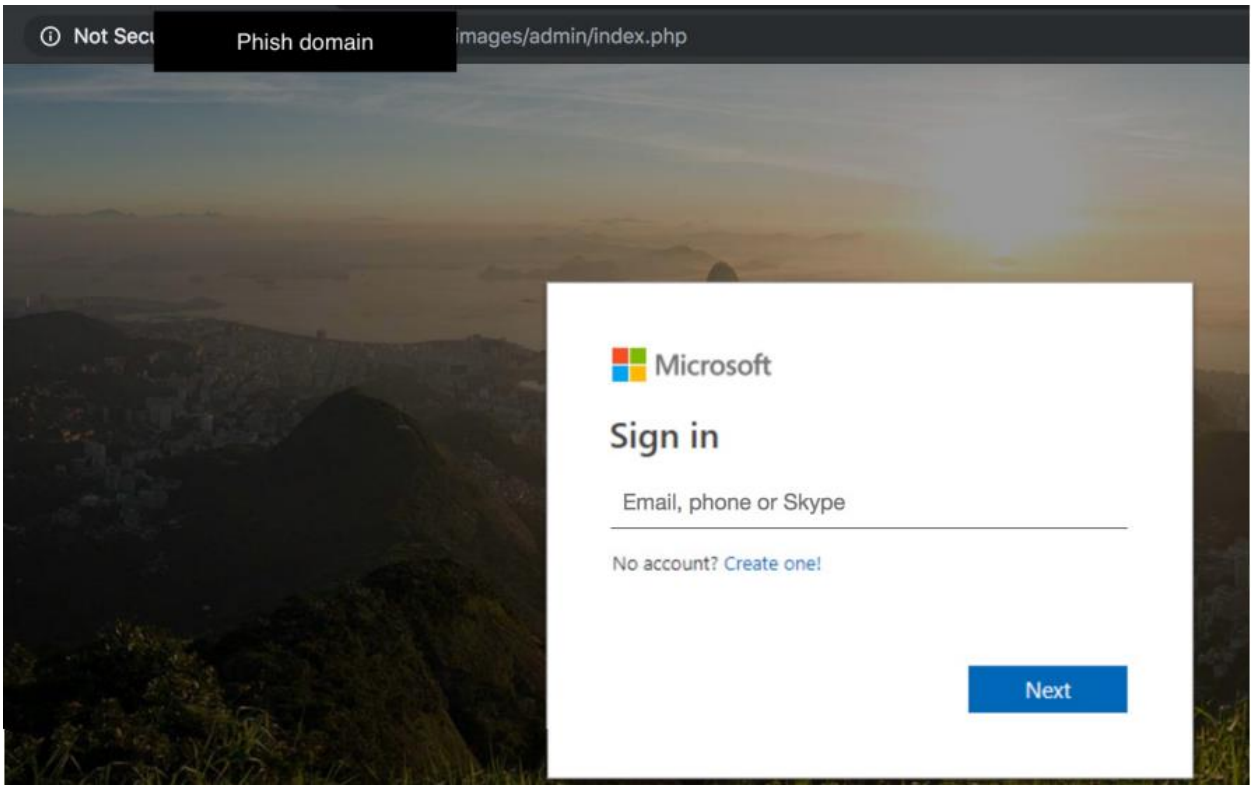


Рисунок 3.1 - Анонімізований скріншот веб-сторінки, на яку вів URL-адрес бокового фішингового листа.

У цьому підрозділі зосередили нашу стратегію виявлення фішингу на основі URL, з огляду на поширеність цього вектора атаки. Така спрямованість означає, що аналіз і методи виявлення не відображають весь простір бічних фішингових атак. Незважаючи на це обмеження, атаки в наборі даних охоплюють десятки організацій, що дозволило вивчити поширений клас корпоративного фішингу, який являє собою важливу загрозу.

3.3 Виявлення бічного фішингу

Прийнявши модель загрози латерального зловмисника, зосередилися на фішинг листах, відправлених зі зламаної облікового запису співробітника, де в якості експлойта використовується шкідливий URL-адресу.

Було досліджено три стратегії виявлення бічних фішингових атак, але в підсумку з'ясували, що одна з стратегій виявляє майже всі атаки, виявлені усіма трьома підходами. На високому рівні дві менш плідні стратегії виявляли атаки шляхом пошуку електронних листів, що містять (1) рідкісну URL-адресу та (2) повідомлення, текст якого, по всій видимості, може бути використаний для фішингу (наприклад, текст, схожий з текстом відомої фішингової атаки). Оскільки наша основна стратегія виявлення виявила всі атаки, крім двох, виявлених іншими стратегіями, але при цьому виявила більш ніж в десять разів більше атак, було відкладення двох менш успішних підходів до Додатку А.3.

Характеристики: Детектор витягує три набори ознак. Перший набір складається з двох ознак, спрямованих на популярну поведінку, яку спостерігали раніше: звернення до багатьох одержувачів. Отримавши листа, спочатку витягаємо кількість унікальних одержувачів з заголовків. Крім того, обчислюємо схожість з набору одержувачів цього листа з найближчим набором історичних одержувачів в будь-якому листі, відправленому співробітником за попередній місяць. Цю останню ознаку (схожість) називаємо оцінкою ймовірності одержувача листа.

Далі доповнюємо набір функцій нашого детектора, витягуючи дві функції, які визначають, чи може лист містити експлойт. Оскільки робота зосереджена на атаках на основі URL-адрес, цей набір ознак відображає, чи містить лист потенційно небезпечний URL.

По-перше, для кожного листа, отримуємо глобальну характеристику репутації URL, яка визначає найбільш рідкісний URL який містить лист. Отримавши листа, ми здобуваємо все з URL-адреси та тіла листа і ігноруємо їх, якщо вони підпадають під

дві категорії: ми виключаємо всі URL-адреси, чий домен включений в список перевірених доменів організації, а також виключаємо всі URL, чий відображений текст з гіперпосиланням точно збігається з URL-адресою місця призначення гіперпосилання. Наприклад, в атаці в лістингу 3.1 атака, що відображається текст фішинговою гіперпосилання був "Натисніть тут", що не збігається з адресою призначення гіперпосилання (фішингових сайтів), тому наша процедура збереже цей URL. На відміну від цього, сигнатура Аліси підпис з лістингу 3.1 може містити посилання на її особистий сайт, наприклад, www.alice.com; наша процедура проігнорує цей URL, так як відображається текст www.alice.com відповідає призначенню гіперпосилання.

Від: "Аліса" <alice@company.com>

Кому: "Боб" <bob@company.com>

Тема: Компанія X (новий контракт)

Новий договір

Переглянути документ [цей текст посилається на фішинговий веб-сайт]

З повагою,Аліса [підпис]

Лістинг 3.1: Знеособлений приклад бокового фішингу повідомлень, в якому використовується підроблений контракт документ

Останній критерій фільтрації заснований на припущенні, що фішинговий URL намагається замаскувати себе і не показує справжнє місце призначення безпосередньо користувачеві. Після цих кроків фільтрації ми витягуємо числову характеристику, зіставляючи кожен URL з його зареєстрованим доменом, а потім переглядаючи рейтинг кожного домена на сайті Cisco Umbrella Top 1 Million [83]; для будь-якого незареєстрованого домену присвоюємо йому рейтинг за замовчуванням в 10 мільйонів. Два особливих випадки ми розглядаємо по-різному. Для URL-адрес на доменах з скороченями детектор намагається рекурсивно дозволити коротке посилання до кінцевого адресата. Якщо це вдається, використовуємо глобальний рейтинг домену кінцевого URL; в іншому випадку ми розглядаємо URL як виходить з

домена без рейтингу (10 мільйонів). Для URL-адрес на сайтах хостингу контенту (наприклад, Google Drive або Sharepoint) у нас немає хорошого способу визначити йогопідозрілість без отримання вмісту і його аналізу (що пов'язане з рядом практичних труднощів). В результаті розглядаємо всі URL-адреси на сайтах хостингу контенту так, як якщо б вони перебували на доменах без рейтингу.

Після ранжирування домену кожного URL-адреси ми встановлюємо глобальну репутацію URL-адреси електронної пошти як найгірший (найвищий) рейтинг домену серед його URL. Інтуїтивно припускаємо, що фішери рідко розміщують фішингові сторінки на популярних сайтах, тому більш висока глобальна репутація URL вказує на більш підозрілий лист. В принципі, мотивований противник може обійти цю особливість; наприклад, якщо противник може скомпрометувати один з перевірених доменів організації, він може розмістити свій фішинговий URL з цього скомпрометованого сайту і уникнути точного рейтингу. Однак не виявили таких випадків в нашому наборі бокового фішингу, про який повідомили користувачі. Крім того, оскільки мета даної роботи - почати вивчення практичних методів виявлення та розробити великий набір інцидентів бічного фішингу для аналізу, цієї функції досить для наших потреб.

На додаток до цієї глобальної метрики репутації витягуємо локальну метрику, яка характеризує рідкість URL-адреси по відношенню до доменів URL-адрес, які співробітники організації зазвичай відправляють. З огляду на набір URL, вбудованих в електронний лист, зіставляємо кожен URL з його повним доменним ім'ям (FQDN - fully-qualifieddomainname) і підраховуємо кількість днів за попередній місяць, коли хоча б одне відправлене співробітником електронного листа включало URL-адресу в FQDN. Потім беремо мінімальне значення за всіма URL-адресами листів; це мінімальне значення називаємо характеристикою репутації локального URL-адреси. Інтуїтивно зрозуміло, підозрілі URL матимуть як низьку глобальну репутацію. Однак оцінка показала, що ця характеристика локальної репутації URL має невелику цінність: URL-адреси з низьким значенням локальної репутації URL майже завжди

мають низьке значення глобальної репутації URL, і навпаки.

Класифікація: Щоб визначити, чи є лист фішинговим чи ні, ми навчили класифікатор Random Forest [84] використовувати вищезазначені характеристики. Для навчання класифікатора взяли всі повідомлення користувачів про бічних фішингових листах в нашій навчальной базі даних і об'єднали їх з набором ймовірних доброякісних листів. Ми створюємо цей набір "доброякісних" листів шляхом випадкової вибірки підмножини листів у вікні навчання, які не були зареєстровані як фішингові; ми вибираємо 200 таких доброякісних листів для кожного атакуючого листа, щоб сформувати набір доброякісних листів для навчання. Дотримуючись стандартної практиці машинного навчання, ми вибрали гіперпараметри для нашого класифікатора і точне співвідношення вибірки (200: 1) за допомогою перехресної перевірки на цих навчальних даних. Більш докладно процедура навчання описана в Додатку А.2.

Після того, як у нас є навчений класифікатор, отримуємо новий лист, детектор витягує його ознаки, передає їх в цей класифікатор і видає рішення класифікатора.

3.4 Характеристика бічного фішингу

У цьому підрозділі представляємо аналіз реального бічного фішингу з використанням всіх відомих атак в нашій базі даних(як навчальної, так і тестової). Протягом семи місяців в цілому 33 організації піддалися бічним фішинговим атакам, причому більшість з цих зламаних організацій було скоєно кілька інцидентів. Вивчаючи тематичний зміст повідомлень та стратегії націлювання на одержувачів стратегії атак, аналіз показує, що більшість латеральних фішерів в вибірці даних не займаються активним вивченням електронної пошти зламаного облікового запису для створення персоналізованих фішингових атак. Швидше, ці зловмисники діють опортуністично і покладаються на звичайний фішингових контент. Цей висновок дозволяє припустити, що простір корпоративного фішингу розширюється за межі його історичної асоціаціїзі складними групами і противниками з числа національних держав.

У той же час, ці атаки все ж вдаються, і значна частина зловмисників дійсно демонструють деякі ознаки витонченості та уваги до деталей. В якості оцінки успіху бічних фішингових атак, принаймні 11% зловмисників з нашої вибірки даних успішно компрометують принаймні один інший обліковий запис співробітника. Що стосується більш витончених тактик, 31% латеральних фішерів докладають деякі ручні зусилля для ухилення від виявлення або підвищення успішності атаки. Крім того, більше 80% атак в нашому наборі даних відбуваються в звичайний час їхнього зламаного облікового запису. Взяті разом, результати свідчать про те, що бічні фішингові атаки представляють собою поширену корпоративну загрозу, яку реальні зловмисники використовують для розширення свого шкідливого доступу(рисунок 3.2).

На додаток до вивчення атак на рівні інциденту, в даному підрозділі також розглядаються атаки на рівні латерального фішера при вивченні різних моделей поведінки зловмисників. Як описано в розділі 3.1, фахівці-практики часто називають такі захоплені облікові записи АТО, і в даному розділі ми використовуємо терміни

"захоплений обліковий запис", "латеральний фішер" і ATO як синоніми.

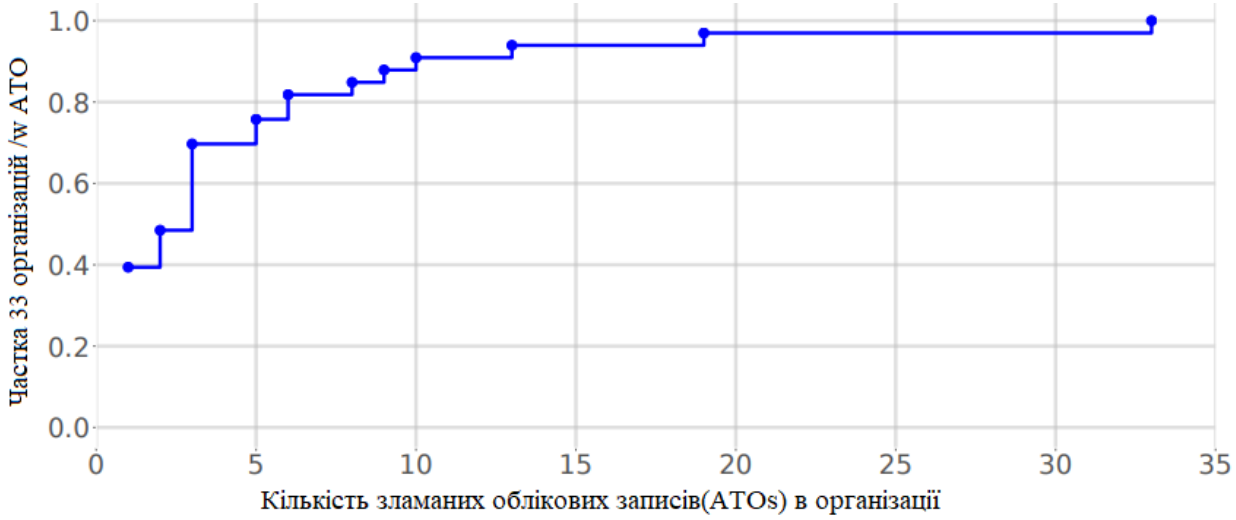


Рисунок 3.2 - Частка організацій з x зламаними обліковими записами, які відправили хоча б один бічний фішинговий лист.

Орієнтування одержувачів і коефіцієнт конверсії

У цьому розділі оцінюємо коефіцієнт конверсії бічних фішингових атак в базі даних і обговорюємо чотири стратегії націлювання на одержувача, які відображають поведінку більшості зловмисників в базі даних. Аналіз показує, що зловмисники прагнули поширитися латерально по організаціям своїх жертв в більш ніж половині зламаних облікових записів в нашому наборі даних. Однак, з огляду на велику кількість одержувачів, на яких направлено більшість цих атак, використання бічного фішингу в нашому наборі даних, мабуть, відображає майстерність опортуністичних зловмисників, у яких, можливо, немає часу або навичок для використання більш складних методів бічного переміщення.

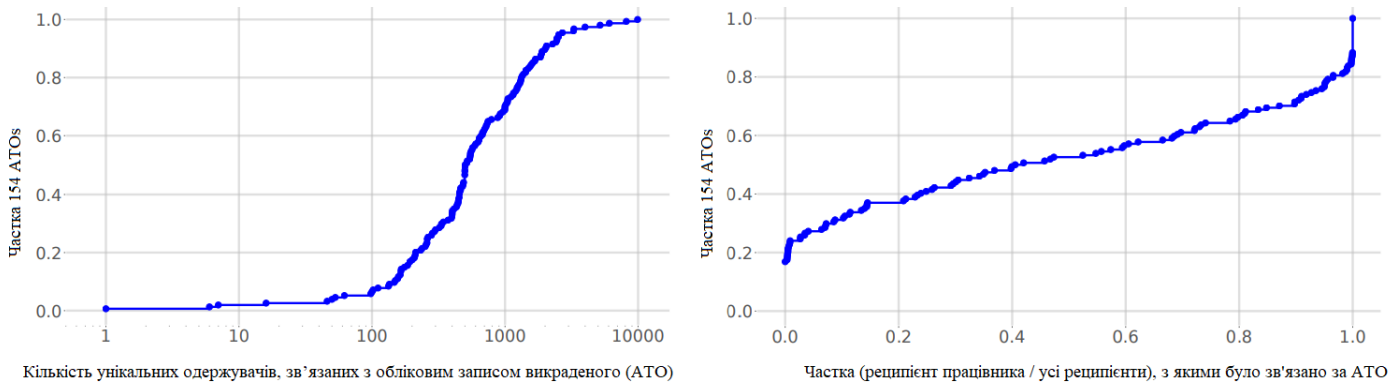


Рисунок 3.3: Лівий CDF показує розподіл загального числа одержувачів фішингових атак по ATO. На Правий CDF показує частку ATO, в яких %x від загального числа одержувачів фішингу складають співробітники.

Обсяг одержувачів і розрахунковий коефіцієнт конверсії: В сукупності латеральні фішери в нашому наборі даних зв'язуються від 101 276 унікальними одержувачами, з яких 41 740 належать до тієї ж організації. Як показано на рисунку 3.3, більше 94% зловмисників відправляють свої фішингові листи більш ніж 100 одержувачам; по відношенню до загальної популяції всіх латеральних фішерів, цей відсоток, ймовірно, завищує поширеність зловмисників з великим "об'ємом одержувачів", оскільки детектор використовує характеристики, пов'язані з одержувачами.

Крім цієї стратегії стримування, ми також оцінили, що в нашому наборі даних бічних фішингових атак важко обдурити окремого співробітника, і тому для захоплення нового аккаунта може знадобитися націлитися на багатьох одержувачів. Раніше в цьому розділі ми виявили, що успішно скомпрометували 23 нових аккаунта. Якщо розділити кількість успішно зламаних облікових записів на кількість співробітників, на яких вони націлилися, то середня швидкість розмови для наших зловмисників склала 1 новий зламаний обліковий запис на 542 співробітника. Метод визначення успішності атаки не охоплює всі випадки, тому показник кількості розмов може також недооцінювати успішність цих атак на практиці. Але якщо розрахунковий коефіцієнт конверсії точно відповідає істинному показником, це пояснює, чому зловмисники зв'язуються з такою великою кількістю одержувачів, не

дивлячись на підвищений ризик виявлення.

Стратегії націлювання на одержувачів: знаємо, що деякі латеральні фішери вибирають своїх жертв, використовуючи інформацію в захопленому акаунті, щоб націлитися на знайомих користувачів. На жаль, набір даних не містить інформації про будь-які розвідувальні дії, які зловмисник виконував для вибору адресатів фішингу.

Таблиця 3.1: Зведення стратегій націлювання на одержувачів ATOs.

Стратегія націлювання на одержувачів ATOs	
Обліково-агностичний	63
По всій організації	39
Бічна організація	2
Цільовий одержувач	44
Безрезультатно	6

Замість цього емпірично досліджуємо набори одержувачів для зловмисників з набору даних, щоб визначити правдоподібні стратегії того, як ці зловмисники могли вибирати жертв. Чотири стратегії вибору одержувачів, узагальнені в таблиці 3.1 і пояснені нижче, відображають поведінку всіх, крім шести, зловмисників в нашому наборі даних. Щоб допомогти оцінити, чи мають одержувачі значущі відносини, вирахували недавні контакти кожного: набір всіх адрес електронної пошти, на які користувач відправив хоча б один лист протягом 30 днів, що передували фішинговим листам. Хоча деякі (28,6%) спеціально націлюються на багатьох з недавніх контактів свого аккаунта, більшість цих латеральних фішерів, схоже, більше зацікавлені або в контактах з багатьма довільними одержувачами, або у відправці фішингових листів великої частини організації зламаного облікового запису.

Зловмисники, які не залежать від облікового запису: Починаючи з найменш цілеспрямованого поведінки, 63 в у наборі даних відправляли свої атаки широкому колу одержувачів, більшість з яких не були тісно пов'язані з зламаними акаунтами. Ми називаємо цю групу "Атакуючі без урахування аккаунта" і ідентифікуємо їх за допомогою двох еврик.

По-перше, ми відносимо зловмисника до категорії "Account-agnostic", якщо менше 1% одержувачів належать до тієї ж організації, що і користувач, і подальше вивчення їх одержувачів не виявляється сильною зв'язку з членством. Якщо розглянути правий графік на малюнку 3.3, то 37 націлених набори на одержувачів, в яких менше 1% одержувачів належать до тієї ж організації, що і відправник. Щоб виключити можливість того, що одержувачі цих зловмисників все ж пов'язані з членством, вирахували частку одержувачів, які фігурували в останні контакти кожного; для всіх 37 можливих користувачів з діагностикою аккаунта, менш 17% від загального числа одержувачів атаки були вказані в недавні контакти. Серед цих 37 облікових записів, 33 з них зв'язувалися з одержувачами в 10 і більше організаціях (унікальні поштові домени одержувачів), 2 з них націлені виключно на акаунти Gmail або Hotmail, тому позначили цих 35 зловмисників як Account-agnostic.

По-друге, розширюємо пошук зловмисників, які залежать від облікового запису, шляхом пошуку зловмисників, де менше 50% всіх одержувачів також належать до організації, і де користувач зв'язується з одержувачами з багатьох різних організацій; зокрема, одержувачі фішингових листів належали до більш ніж в два рази більшій кількості унікальних доменів, ніж всі адреси електронної пошти в недавніх контактах. Цей пошук виявив 63 користувачів. Щоб відсіяти зловмисників з цього набору, які могли використовувати недавні контакти зламаної облікового запису, ми виключили все, де більше 17% (17% - це максимальний відсоток серед користувачі з першої евристики діагностики акаунтів). Після застосування цієї останньої умови, наша друга евристика ідентифікує 54 зловмисника з діагностикою акаунтів.

Об'єднання і дедуплікація за обома критеріями дає в цілому 63 Account-agnostic зловмисників (40,9%): латеральні фішери, які переважно орієнтовані на одержувачів, які не мають тісних зв'язків з зламаним аккаунтом або його організацією.

Атакуючі в масштабах організації: Office 365 надає функцію «Групи», в якій перераховані різні групи, до яких належить обліковий запис [85]. Для деяких

підприємств ця функція перераховує більшість, якщо не всіх співробітників організації. Таким чином, латеральні фішери, бажаючі закинути широку фішингову мережу можуть застосувати просту стратегію відправки своєї атаки всім співробітникам організації. Ми називаємо таких ATOs (Organization-wide attackers) - атакуючі в масштабах організації, і виявляємо їх двома способами.

По-перше, ми шукаємо всіх зловмисників, у яких принаймні половина одержувачів фішингових листів належить до організації ATOs, і де принаймні 50% співробітників організації отримали фішинговий лист; цей пошук дав в цілому 16 ATOs. Ми оцінюємо список співробітників організації, складаючи набір всіх адрес електронної пошти співробітників, які відправляли або отримували електронну пошту від кого-небудь протягом усього місяця фішингового інцидента. Для всіх цих 16 ATOs менше 11% адресатів, на яких вони націлилися, також фігурують в їх недавніх контактах. У поєднанні з тим фактом, що кожен з цих ATOs зв'язується з більш ніж 1300 одержувачами, їх поведінку дозволяє припустити, що їх первинною метою є фішинг якомога більшого числа одержувачів підприємства, а не націлювання на користувачів, особливо близьких до зламаної облікового запису. Відповідно, відносимо їх до категорії зловмисників, що діють в масштабах всієї організації.

Наша друга евристика шукає зловмисників, чий набір одержувачів майже повністю складається із співробітників, але при цьому більшість співробітників організації не обов'язково отримують фішингові листи. Звертаючись до рисунка 3.3, можна помітити, що 36 ATOs, які є кандидатами в масштабах організації, надіслали понад 95% своїх фішингових листів товаришам по службі. Однак нам знову необхідно виключити і врахувати ATOs, які використовують останні контакти свого зламаного облікового запису. З першої евристики для всієї організації, розглянутої раніше, побачили, що менше 11% одержувачів, що атакували зловмисники в масштабах всієї організації, були з недавніх контактів ATOs. Використовуючи це значення в якості остаточного порога для другого набору кандидатів в загальноорганізаційні зловмисники, визначили 29 загальноорганізаційних

зловмисників, більше 95% одержувачів яких належать до організації ATOs, але менше 11% одержувачів були з числа недавніх контактів ATOs; таке поєднання передбачає, що атакуючий прагне в першу чергу скомпрометувати інших співробітників, але які не обов'язково мають особистий зв'язок з зламаної обліковим записом.

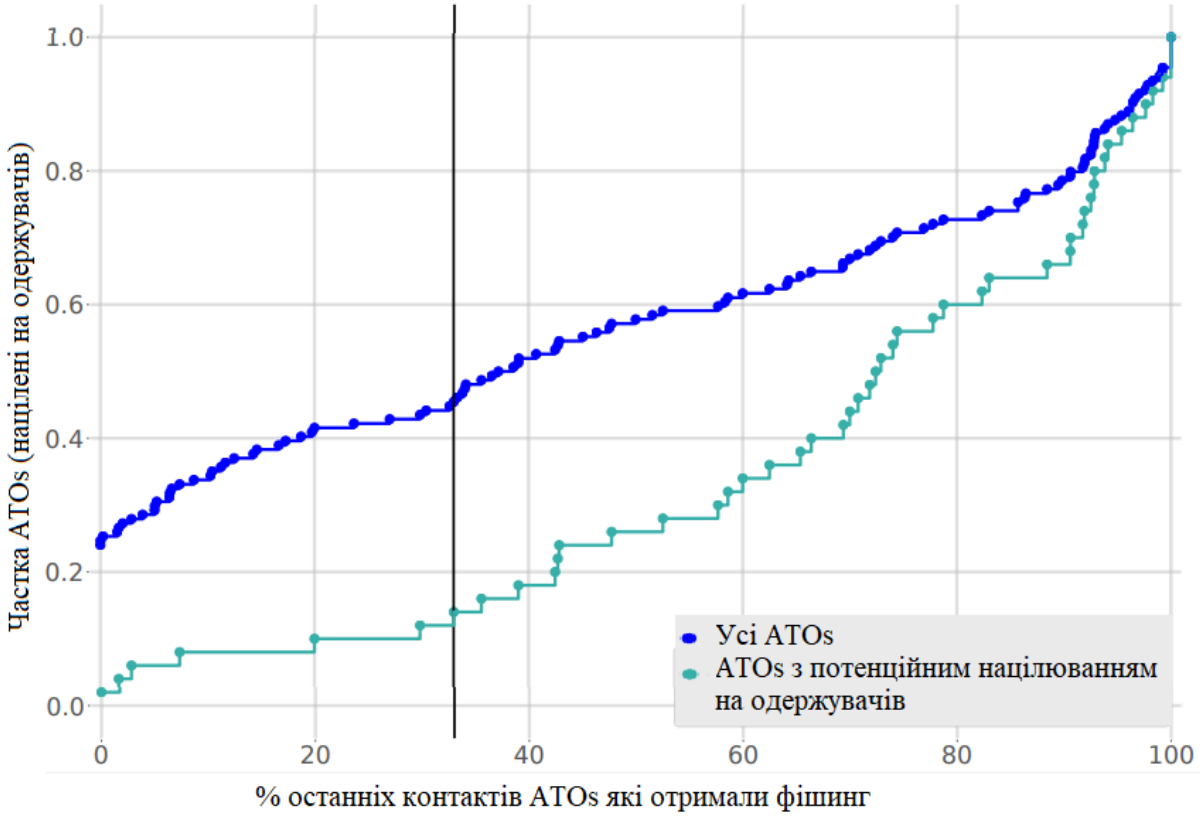


Рисунок 3.4 - CDF: вісь x показує, який % недавніх контактів ATOs отримав бічний фішинговий лист.

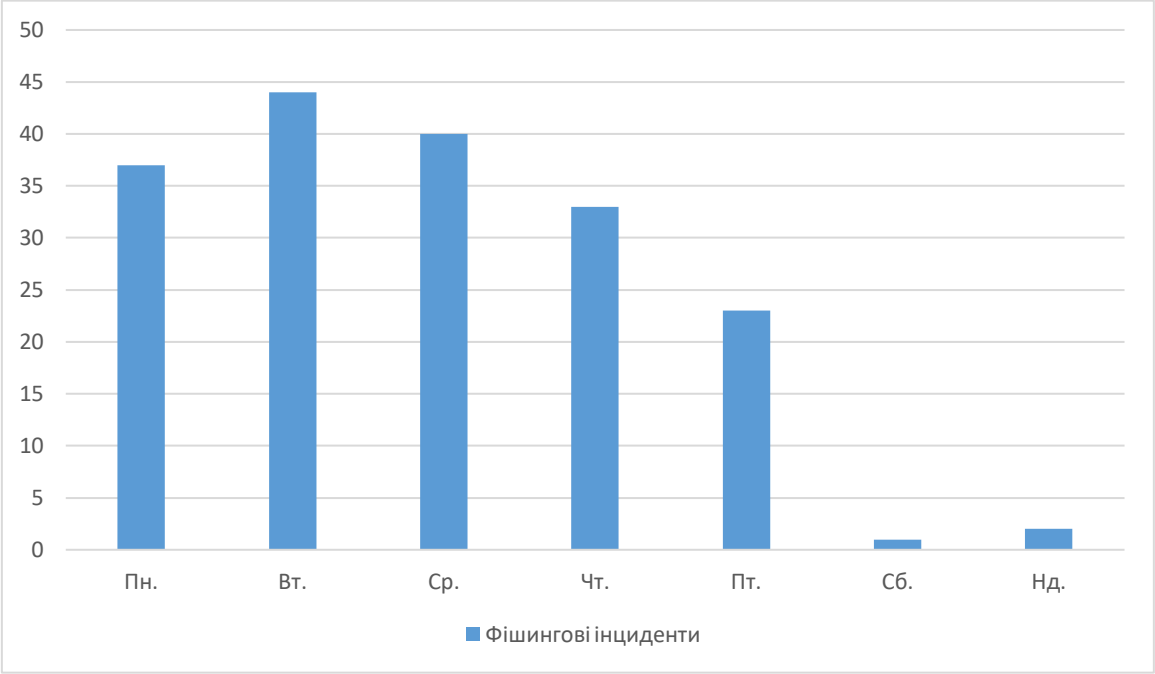
Агрегування і дедуплікація двох вищевказаних наборів латеральних фішерів дає в цілому 39 зловмисників в масштабах всієї організації (25,3%), які спрямовують свої фішингові атаки на багатьох співробітників.

Зловмисники, націлені на одержувачів: Для решти, не віднесених до категорії 50 ATOs, не можемо остаточно визначити стратегії нападу на одержувачів, так як наш набір даних не надає нам повний набір інформації, доступної зловмисникам. Проте, на рисунку 3.4 представлені деякі докази того, що 44 з цих зловмисників дійсно використовують попередні зв'язки захопленого аккаунта. Зокрема, 44

зловмисника відправили свої атаки принаймні на 33% адрес в недавні контакти ATOs. Оскільки ці ATOs відправляли атаки принаймні на кожен третій з останніх контактів ATOs, ці зловмисники, очевидно, були зацікавлені в атаці на значну частину користувачів з відомими зв'язками з зламаного облікового запису. Таким чином, позначили ці 44 ATOs як «Атакуючі, націлені на одержувачів».

Тимчасові аспекти бічного фішингу

Оскільки зловмисники можуть жити або діяти не в тому самому географічному регіоні, що і зламаний акаунт, в попередніх роботах пропонувалося використовувати ознаки, які фіксують незвичайні тимчасові характеристики, властиві фішинговим листам [38, 42, 86]. Всупереч цій інтуїції, в нашому наборі даних більшість бічних фішингових атак відбувається в "нормальний" час дня і тижня. По-перше, в 98% випадків бічного фішингу зловмисник відправляв фішингових лист в будній день. Крім того, більшість зловмисників в нашій базі даних відправляють свої фішингові листи в звичайний час їхньої роботи справжнього аккаунта.



Діаграма 3.1 - Кількість інцидентів бічного фішингу по днях тижня.

Дні тижня: Як видно на діаграмі 3.1, все, крім трьох, бічні фішингові інциденти сталися протягом робочого дня (з понеділка по п'ятницю). Така картина говорить про те, що зловмисники відправляють свої фішингові листи в ті ж дні, коли співробітники

зазвичай відправляють свої доброякісні листи, і що день тижня буде неефективним або слабким сигналом для виявлення. Більш того, 67% інцидентів відбуваються в першій половині тижня (пн-ср), що вказує на те, що латеральні фішери в нашій вибірці даних не дотримуються фольклорної стратегії, згідно з якою зловмисники воліють починати свої атаки в п'ятницю. [87].

Час (години) доби: Крім роботи протягом звичайної робочого тижня, більшість зловмисників відправляють свої бічні фішингові листи в звичайні робочі години своїх зламаних акаунтів. Щоб оцінити (ненормальність) час відправлення атаки, для кожного АТО зібрали всі листи, які аккаунт відправив за 30 днів до першого бічного фішингового листа. Потім зіставили час відправки кожного з цих історичних (і, імовірно, доброякісних) листів з часом дня в 24-годинному масштабі, сформувавши таким чином розподіл типових годин дня, в які кожен зламаний акаунт зазвичай відправляв свої листи. Нарешті, для кожного бічного фішингового інциденту ми вирахували проценти для години дня фішингових листів щодо розподілу годин дня для історичних листів АТО. Наприклад, фішингові інциденти з процентами 0 або 100 були відправлені в більш ранній або більш пізньої пори дня, ніж будь-який лист, відправлений власником справжнього аккаунта в попередні 30 днів.

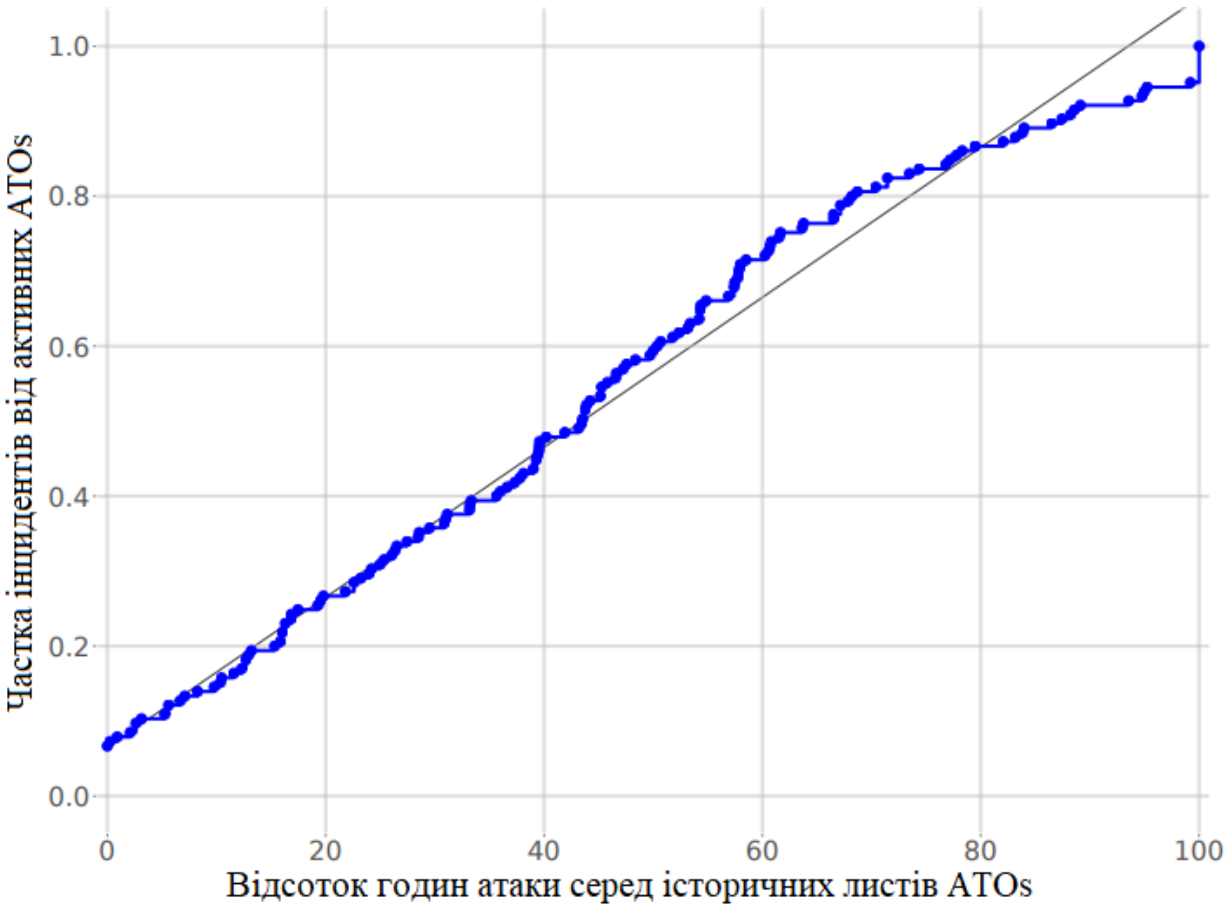


Рисунок 3.5 - CDF частки інцидентів з активних АТО, в яких час (годину) діб потрапляє в межі x-го проценту годин, в які були відправлені доброякісні електронні листи АТО за попередні 30 днів.

На рисунку 3.5 показані всі інциденти бічного фішингу, відправлені активним АТО, а також процентне співвідношення годин і днів, в які було відправлено перший лист фішингових інциденту, по відношенню до історичних листів зламаного облікового запису. З 180 інцидентів 15 були відправлені "неактивними" (спокійними) АТО, які відправили нуль листів за усі 30 днів, що передували їх боковим фішинг листам; рисунок 3.5 виключає ці інциденти. З решти 165 інцидентів, відправлених активним АТО, 18 інцидентів повністю виходять за рамки історичних годин роботи зламаного облікового запису, що дозволяє припустити, що функція пошуку листів, відправлених в нетипове для користувача час, могла б допомогти виявити ці атаки. Однак, в інших 147 інциденти час доби фішингових листів рівномірно покриває весь процентний діапазон. Як показано на рисунку 3.5, процентний розподіл годин

фішингу дуже схожий на CDF рівномірного випадкового розподілу. CDF рівномірного випадкового розподілу (пряма лінія y = x). Цей результат показує, що для більшості інцидентів в нашому наборі даних (147 з 180), час дня коли АТО відправив атаку, не дає істотного сигналу, так як час відправки атаки відображає часовий розподіл історичної активності справжнього користувача по електронній пошті.

Таким чином, на основі атак в нашому наборі даних виявили, що існують дві слабких ознаки, пов'язаних з часом: пошук спокійних акаунтів, які раптово починають відправляти підозрілі листи (15 інцидентів), і пошук підозрілих листів, відправлених повністю поза історично активного тимчасового вікна аккаунта (18 інцидентів). Крім цих двох особливостей і невеликої частки фішингових атак, які вони відображають ні день тижня, ні час доби та не дають істотних сигналів для виявлення.

4 МОДЕЛЮВАННЯ ТА ВИЯВЛЕННЯ БІЧНОГО ПЕРЕМІЩЕННЯ

4.1 Модель безпеки

Зловмисникам часто доводиться поширюватися за межі первісного «плацдарму» зламаної ними машини на інші машини в пошуках серверів і необхідних облікових даних, які забезпечують їм потрібні дані і можливості бічного переміщення та охоплює внутрішні переміщення зловмисника між машинами в межах підприємства для досягнення своєї мети.

Нижче опишемо мету безпеки і типи бічного переміщення, які покликана виявити наша система виявлення Hopper.

Цілі виявлення: Ми розробляємо Hopper з урахуванням чотирьох цілей:

1. Покладається тільки на зазвичай збираються журнали аутентифікації підприємства.

2. Генерує дуже низький обсяг помилкових спрацьовувань, прийнятний для аналізу сайтами.

3. Виявляє широкий спектр атак бічного переміщення.

4. Ефективно працює при відсутності позначених примірників атак.

Таблиця 4.1 - Інформація, що міститься для кожної події входу в систему в наших даних.

Вузли (джерело + машини призначення)	Входи
Ім’я хосту	Мітка часу
Клієнт проти сервера	Цільове ім’я користувача
Ім’я користувача власника (лише для клієнтів)	

Ми вважаємо Hopper є успішним, якщо він видає попередження для будь-якого входу в систему, зробленого зловмисником, що здійснює бічне переміщення. При підтвердженні наявності атаки служба безпеки організації команда безпеки організації може використовувати методи судової експертизи з попередніх робіт [70, 88, 89] для проведення додаткового аналізу і усунення наслідків.

Hopper націлений на виявлення багатьох поширених типів бічного переміщення в реальному світі, генеруючи при цьому на прийнятну кількість помилкових спрацьовувань. Балансуючи між цими цілями, Hopper може не виявити нові випадки бічного переміщення, запропоновані в попередній роботі, такі як атаки, коли противник переміщається строго за допомогою перехоплення або копіювання послідовності логінів, зроблених легітимними користувачами і атаки, при яких противник успішно "отруює" набір даних входу в систему організації історичними логінами, завдяки чому кінцевий шлях атаки виглядає доброякісним і часто відвідуваним.

Модель загрози: Як і в попередній роботі, зосередилися на виявленні інтерактивних і заснованих на облікових даних атак бічного переміщення [71]. Відповідно до цієї моделі загроз, припускаємо, що зловмисникові вдалося скомпрометувати первісну «опорну» машину на підприємстві, але йому (1) згодом необхідно отримати додаткові повноваження для доступу до даних або систем, які вони в кінцевому підсумку шукають, і (2) переміщатися між машинами через події входу в систему або віддаленого виконання команд, які використовують набір облікових даних для аутентифікації, а не шляхом доступу до машин через уразливості. Крім того, ця модель загроз фокусується на зловмисників, які вручну виконують кожну з операцій переміщення (входу в систему) під час атаки, на відміну від атаки, яка встановлює шкідливе ПЗ, що переміщається в нові системи автономно.

Описана модель загроз відображає поведінку багатьох реальних атак з бічним переміщенням, починаючи від цілеспрямованих атак національних держав [90, 91, 92, 93, 94, 95, 96] до новіших, укритті форм ransomware [97, 98]. У нашій моделі загроз не розглядаються зловмисники, чия первісна атака вже отримала бажаний доступ, оскільки в таких атаках відсутнє бічне переміщення. Ми відзначаємо, що організації можуть використовувати методи, розроблені в додаткових попередніх роботах [61, 62, 63, 64] для обмеження користувачів і машин, які мають прямий доступ до конфіденційних даних або потужним функціям, тим самим зменшуючи перспективи

атак. функціональності, тим самим знижуючи ймовірність таких атак.

Етика

Ця робота включала в себе співпрацю між науковими та промисловими колами. У нашому дослідженні використовувався існуючий історичний набір даних про входах співробітників між внутрішніми машинами в Dropbox, який підприємства зазвичай збираються для пом'якшення атак і забезпечення безпеки внутрішнього середовища. Доступ до цих даних мали тільки уповноважені співробітники служби безпеки Dropbox; ніякі конфіденційні дані або особиста інформація не передавались за межі Dropbox. Крім того, машини, що зберігають і працюють безпосередньо з даними клієнтів Dropbox, знаходяться в окремій, сегментированной інфраструктурі; наше дослідження не зачіпало цю інфраструктуру і не мало доступу до даних клієнтів. Цей проект пройшов внутрішню перевірку і отримав схвалення з боку команд з юридичних питань, конфіденційності та безпеки компанії Dropbox.

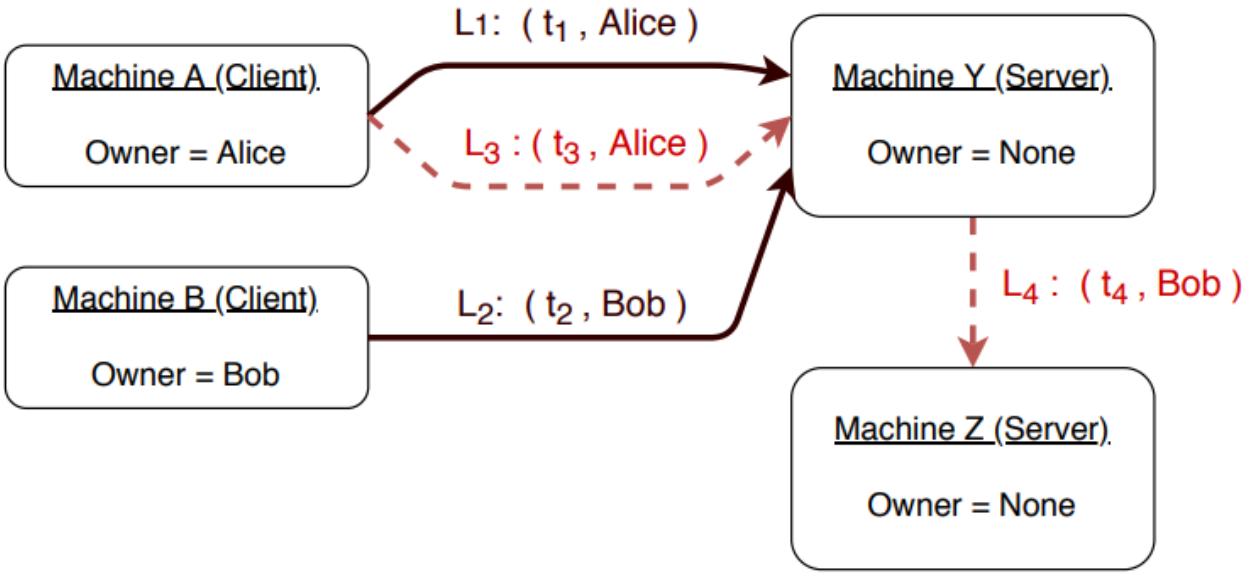


Рисунок 4.1 - Приклад простого графа входу в систему. Суцільні чорні ребра (L_1 і L_2) відповідають доброякісним подій входу в систему. Пунктирні червоні ребра (L_3 і L_4) відповідають шляху атаки бічного переміщення. Кожен вузол графа являє внутрішню машину підприємства (наприклад, машини A, B, Y, Z) і містить інформацію в стовпці 1 таблиці 4.1. Кожне ребро в графі відповідає унікальній події входу в систему і містить інформацію в стовпці 2 таблиці 4.1.

4.2 Hopper: Огляд системи

Система, Hopper, виявляє підозрілі шляхи входу в систему, далі ми описуємо граф, який Hopper будує з журналів віддалених входів, обговорюємо проблеми, що виникли при застосуванні стандартного підходу до виявлення аномалій до нашого набору даних, і стисло описуємо архітектуру Hopper.

Наш підхід: Hopper будує граф призначених для користувача логінів між внутрішніми машинами і потім виявляє бічний рух, визначаючи «підозрілі» шляхи в цьому графі. Підозрілий шлях відповідає послідовності логінів, зроблених одним суб'єктом, що володіє двома властивостями: (1) шлях має принаймні один вхід, де агент використовує набір облікових даних, який не збігається з його власними, (2) шлях отримує доступ принаймні до однієї машині, до якої агент не має доступу під своїми обліковими даними.

Мотивуюча інтуїція: Цей підхід використовує просте, але потужне спостереження: у багатьох реальних корпоративних атаках противники здійснюють бічне переміщення, щоб отримати додаткові облікові дані і доступ до нових машин, до яких у їх первісного плацдарму не було доступу [91, 92, 93, 94, 95, 96, 99]. Наприклад, у багатьох організаціях доступ до конфіденційних даних і / або потужним внутрішнім можливостям вимагає спеціального набору облікових даних і привілегів, яких немає у більшості корпоративних користувачів. Таким чином, при бічному переміщенні зловмисника виникають шляхи, що використовують новий набір облікових даних (властивість 1) і доступ до машин, до яких їх первісна жертва не могла отримати доступ (властивість 2). Більш того, ці підозрілі характеристики також відповідають руху, якого ми не очікуємо від доброякісних шляхів: користувачі повинні отримувати доступ до машин під своїми власними обліковими даними, і вони повинні входити тільки в ті машини, які у них є законні привілеї для доступу. Кожне з властивостей нашої атаки відповідає порушень цих очікуваних моделей поведінки. Ми показуємо, що цей підхід дає на порядки менше попереджень ніж традиційний підхід до виявлення аномалій, наприклад, запропонований в попередній роботі.

Граф логінів

З огляду на набір логінів, Hopper будує спрямований мультиграф, який відображає взаємодію між користувачами і машинами на підприємстві. На рисунку 4.1 показаний простий приклад графа входів в систему побудованого Hopper. Кожен вхід створює спрямоване ребро в графі, де джерело ребра і вузол призначення відповідають машині, яка ініціює і приймаючої вхід, відповідно. Грані представляють унікальні, відмічені часом входи в систему від джерела до машини призначення; кілька входів в систему між двома однаковими машинами створюють кілька ребер. Кожне ребро анотується цільовим ім'ям користувача, яке є обліковим записом, під яким був виконаний вхід на машині призначення; цільове ім'я користувача визначає, під яким користувачем і дозволами буде працювати сесія входу на цільовій машині. У таблиці 4.1 показана інформація, що міститься в наших даних про входах, яка використовується для маркування кожного ребра і вузла в графі.

Шляхи входів і причинні користувачі: Шлях входу в систему відповідає серії пов'язаних ребер, де кожне ребро «викликано» одним і тим же учасником: тобто шлях - це послідовність пов'язаних логінів, які один актор зробив від початкової машини шляху до кінцевого пункту призначення шляху. Наприклад, на рисунку 4.1 зловмисник компрометує корпоративну машину Аліси (A) і здійснює серію внутрішніх входів, які утворюють двуххоповий шлях бокового переміщення від машини A до машини Z. Спочатку зловмисник входить в обліковий запис Аліси на машині Y, використовуючи облікові дані первісної жертви, показані як L_3. Потім зловмисник компрометує облікові дані Боба на машині Y і використовує їх для входу в обліковий запис Боба на машині Z, позначену L_4. Для кожного з входів в систему на цьому шляху атаки Аліса є причинним користувачем, оскільки всі входи були зроблені (викликані) користувачем, початківцям з машини Аліси.

Аномалії в масштабі

Попередні роботи виявляють бічний рух шляхом виявлення входів в систему, які проходять через рідкісні ребра графа, виходячи з припущення, що рух

зловмисника буде відбуватися між користувачами або машинами, які рідко взаємодіють один з одним [65, 67, 71]. Хоча ці підходи інтуїтивно зрозумілі, в кінцевому підсумку вони дають занадто багато помилкових спрацьовувань для практичного використання через різноманітного спектра рідкісної, але доброякісної поведінки, яка зустрічається на великих підприємствах. Навіть після застосування кроків, які робить Hopper для усунення артефактів і автоматизації, ми виявили, що десятки тисяч логінів створюють "рідкісні" ребра графа внашому наборі даних. На рисунку 4.2 показано кількість входів в систему, які простий детектор аномалій відзначив би, якби він попередив про будь-якому вході в систему, чия грань зустрічалася X ≤ днів за останні 60 днів попередніх входів в систему, після їх дедуплікації, щоб видавати тільки одне попередження на унікальну грань кожен день. Навіть при самому агресивному порозі в 0 попередніх днів, тобто краях, які ніколи не виникали в недавній історії, цей підхід до виявлення аномалій все одно видасть більше 24 000 попереджень по всьому набору даних (більше 1 600 попереджень на місяць).

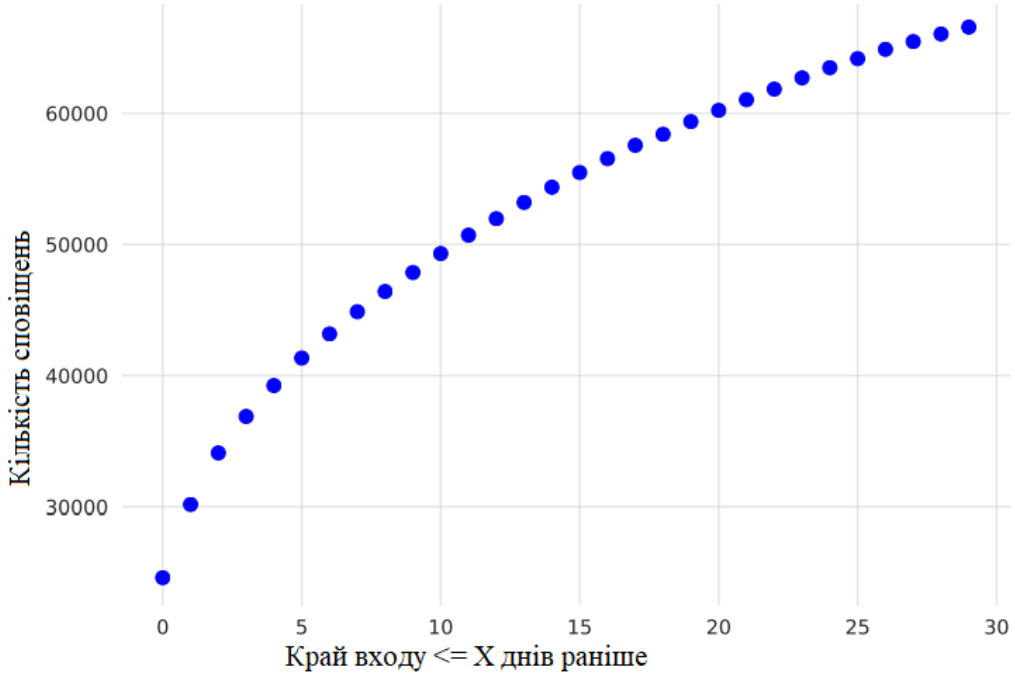


Рисунок 4.3 - Кількість входів в систему з рідкісними ребрами графа, які відбулися за X ≤ днів протягом 60 днів, попередніх входів в систему.

4.3 Генерація причинно-наслідкових шляхів входу в систему

Хоча стандартні журнали машинної аутентифікації містять багатий набір інформації, вони відображають точкову активність, в якій відсутній контекст про ширшу активності користувача, пов'язаної з входом в систему, наприклад, від кого і де стався вхід, хто і звідки увійшов в систему. Наприклад, на рисунку 4.1, якщо розглядати логін L_4 ізольовано, детектор не має уявлення про те, чи відображає Боб точно користувача, відповідального за виконання входу в систему, або ж це інший користувач, наприклад, той, хто виконав вхід в систему, або ж інший користувач, наприклад Аліса, вкрав облікові дані Боба і використовує їх для входу в систему. Таким чином, для кожного входу (L_i), який відбувається, на першому етапі Hopper запускає «механізм причинності», який грубо визначає більш широкий шлях руху, до якого належить логін, «причинного» користувача, відповідального за ініціювання шляху руху, відповідального за ініціювання шляху руху. Для цього Hopper використовує евристику, засновану на часі, для виведення набору «причинних шляхів» для L_i де кожен шлях, створений Hopper, відповідає унікальній послідовності пов'язаних входів в систему, яка включає L_i і сталася протягом максимального часу для сеансу входу в систему.

У деяких випадках Hopper може чисто вивести один шлях і причинного користувача для нового входу, що дозволяє Hopper класифікує шлях входу в систему, використовуючи простий набір правил виявлення (у підрозділі 4.4). Однак при деяких входах в систему Hopper буде генерувати безліч передбачуваних шляхів з різними причинними користувачами, створюючи невизначеність щодо справжнього причинного користувача і шляхи для Li. Щоб впоратися з цим останнім випадком, Hopper покладається на алгоритм оцінки аномалій при ухваленні рішення про те, чи слід попереджати про передбачуваному шляху (підрозділ 4.4). У решти цього розділу опишемо, як механізм причинності Hopper визначає набір шляху для нового входу в систему, а подробиці виявлення та оповіщення відкладемо до розділу 4.4.

Кожен з причинно-наслідкових шляхів, які Hopper виводить для L_i, містить

інформацію, наведену в таблиці 4.2: шлях фокальний стрибок (логін, L_i для якого Hopper побудував шлях), список всіх логінів (хопов) на розрахунковий шлях, причинний користувач шляху і "ймовірність" шляху, яка висловлює впевненість Hopper в тому, що поточний шлях відображає істинний причинний шлях L_i; Hopper оцінює цю ймовірність як простий дріб: 1 / (кількість причинно-наслідкових шляхів, які Hopper вивів для фокального стрибка).

Таблиця 4.2 - Інформація, що міститься в передбачуваному причинно-наслідковому шляху, створеному механізмом причинності Hopper.

Компонент шляху	Опис
Фокальний стрибок	Остаточний стрибок шляху, або шлях перемикання облікових данних (Розділ 4.3)
Список стрибків	Список входів на шляху (таблиця 4.1)
Причинний користувач	Ім’я користувача, працівника, чия машина ініціювала шлях
Імовірність шляху	Частка причинних шляхів, щоб Hopper зробив висновок про фокальний стрибок

Визначення причинно-пов'язаних входів: Hopper створює набір причинно-наслідкових шляхів, виконуючи пошук в зворотному напрямку від L_i щоб визначити послідовність причинно пов'язаних логінів, які включають L_i. Два логіна причинно пов'язані, якщо вони (1) утворюють зв'язний набір ребер в графі логінів і (2) відбуваються протягом T годин один від одного. Більш формально, ми говоримо, що L_k є причинно-наслідковим, що входять входом для L_i якщо місце призначення L_k вихідної машини L_i і L_k стався протягом 24 годин до моменту часу L_i. Ми вибрали поріг в 24 години в якості непрямого показника максимальної тривалості сеансу входу в систему в Dropbox; завдяки конфігурації машини і мережі, сесії, що перевищують цю тривалість, вимагають сеанси, що перевищують цю тривалість,

вимагають повторної аутентифікації вихідної машини, що створює нову подію входу в систему в нашому наборі даних. Для Наприклад, на рисунку 4.2, L_1, L_2 і L_3 є причинно-наслідковими зв'язками для L_4, якщо вони сталися протягом 24 годин до t_4.

Використовуючи правило причинно-наслідкового зв'язку, може вивести набір шляхів входу в систему, визначивши всі причинні зв'язку входи в систему для L_i і потім рекурсивно повторюючи цей пошук для кожного з цих причинних логінів. Цей процес схожий на методи перевірки походження і відстеження слідів, які відстежують потік інформації від стоку (кінцева машина L_i) назад до його джерела (наприклад, клієнтської машині в корені шляху входу L_i) [89, 100, 101]. Як і в попередніх методах відстеження потоків, наївне відстеження у зворотному напрямку створює «вибух залежностей», коли кожен крок назад може експоненціально збільшити кількість (потенційних) причинно-наслідкових шляхів, які виводить Hopper, але тільки один з цих виведених шляхів відповідає істинному причинному шляху L_i. Щоб вирішити цю проблему, механізм причинності Hopper використовує двухпрохідний алгоритм, який генерує тільки довгі шляхи (три або більше стрибків) для логінів, які, ймовірно, можуть належати до шляху атаки.

Механізм причинності Hopper приймає три вхідних сигнали: нова подія входу (L_i), для якого потрібно визначити шляхи, набір всіх недавніх входів в систему і поступово оновлюваний «список спостереження» підозрілих шляхів входу в систему. Перший прохід механізму причинності Hopper спрямований на прийняття жодного рішення про L_i шляхом створення набору одноходових або двоходових причинно-наслідкових шляхів, а потім передає ці шляхи генератору попереджень. Якщо генератор попереджень Hopper не може прийняти чисте рішення про доброякісності або шкідливість, другий прохід механізму причинності продовжує відстежувати шлях і постійно відправляє оновлений, довший шлях генератору попереджень для кожного майбутнього входу в систему.

Перше проходження: Побудова коротких причинно-наслідкових шляхів. З огляду на вхід в систему, перший прохід механізму причинності Hopper створює або

однохоповий шлях, або набір двуххопових шляхів, в залежності від того, чи відбувається вхід з клієнта або сервера.

Нехай L_i представляє вхід в систему з вихідної машини Y на кінцеву машину Z з цільовим користувачем Боб. Якщо Y є сервером, то Hopper створює набір двуххопових шляхів, виконуючи одну ітерацію зворотного трасування для виведення набору входів причинних логінів для L_i. Потім Hopper зіставляє кожен з цих вхідних логінів з L_i, щоб сформувати набір двуххопових шляхів, кожен з яких містить інформацію в таблиці 4.1. Для кожного шляху Hopper встановлює L_i як фокального хопу і додає ці два логіна в список хопів шляху. Hopper встановлює причинного користувача шляху рівним цільовому імені користувача входу хопу. Імовірність шляху дорівнює 1 поділеній на загальну кількість двуххопових шляхів, які Hopper визначив для L_i. Наприклад, на рисунку 4.1, якщо припустити, що L_1, L_2 і L_3 відбулися протягом 24 годин до L_4, Hopper створить 3 причинно-наслідкових шляхи для L_4. У кожному з цих шляхів L_4 буде вказано як фокальний стрибок і буде мати "Імовірність шляху" яка дорівнює 1/3. І шлях атаки (L_3 - L_4), і шлях від L_1 до L_4 вказуватимуть на Алісу в якості причинного користувача, а в передбачуваному шляху від L_2 до L_4 як причинного користувача буде вказано Боб.

На відміну від цього, якщо Y є клієнтом, Hopper не виконує жодного зворотного відстеження. Замість цього, механізм причинності Hopper виводить єдиний одноходовий шлях зі списком хопів і фокусним хопом, рівним L_i, бере власника Y і розглядає це ім'я користувача як причинного користувача. Як причинного користувача: клієнти зазвичай відповідають початку шляху переміщення користувача, і при вході в систему з цих машин слід використовувати облікові дані їх власника, оскільки головною метою механізму причинності Hopper є висновок причинного користувача і відповідного шляху нового входу в систему, Hopper не потрібно робити ніяких зворотних висновків, оскільки вхід клієнта в систему відзначає початок шляху, і ці машини вже містять інформацію про причинного користувача входу (тобто власника клієнта). У прикладі з рисунку 4.1, Hopper генерує

три однохопових шляху: по одному для кожного з L_1, L_2 і L_3.

Другий прохід: Відстеження довгих причинно-наслідкових шляхів

Другий етап роботи механізму причинності Hopper дозволяє йому відстежувати і будувати шляхи довільної довжини (три або більше хопов). Для цього Hopper веде контрольний список попередніх шляхів, які алгоритм виявлення Hopper не зміг чітко позначити як доброякісні або підозрілі. Спочатку цей список порожній. В процесі обробки і виявлення причинно-наслідкових шляхів для нових входів в систему Hopper поступово додає деякі з цих шляхів в список спостереження, якщо генератор попереджень Hopper не може видати двійкову мітку для шляху. Другий прохід механізму причинності Hopper відстежує шлях в цьому списку і перевіряє, розширює чи був новий вхід в систему будь-якого з цих шляхів; якщо так, то Hopper повторно відправляє на кожен з них знову розширені шляхи на перекласифікацію.

Розширення шляхів у списку спостереження: Для кожного нового входу (L_i), який відбувається, Hopper виконує обидва проходи свого механізм причинності. Під час другого проходу Hopper запускає алгоритм виведення шляху, щоб визначити всі шляхи зі списку спостереження, які розширює L_i; тобто там, де Hopper визначає, що кінцевий стрибок шляху зі списку спостереження є причинним, що є входом для L_i. Якщо L_i розширює будь-які шляхи сторожового списку, Hopper створює новий "розширений шлях" (P_2), копіюючи вміст шляху списку спостереження, а потім додаючи L_i в "список хопів" P_2. (Всі інші атрибути шляху залишаються тими ж, що і в початковому шляху списку годин). Потім Hopper бере P_2 і передає його в свій алгоритм генерації попереджень, який видає або двійковий вердикт (доброякісний або підозрілий), або додає P_2 в список спостереження.

Обрізка сторожового списку: Коли Hopper перебирає шляху зі списку спостереження під час другого проходу, він видаляє всі шляхи, де останній стрибок стався більш ніж за T годин до L_i, де T відображає максимальну тривалість віддаленого сеансу (наприклад, 24 години). Оскільки час між останнім входом в систему і L_i перевищує максимальну тривалість сеансу, шлях зі списку

спостереження не міг викликати L_i, якщо тільки причинний користувач не виконав повторне підключення до сеансу або новий вхід в систему; обидва ці випадки створять нову подію входу в систему (і відповідні шляхи) в наших даних.

Покриття механізму причинності: Цей двопрохідний підхід дозволяє Hopper ефективно відстежувати шляхи багатохопових атак і ідентифікувати входи, які включають (потенційно) підозріле використання облікових даних. Будь-який шлях, що включає перемикання облікових даних, буде містити принаймні один вхід, який демонструє це перемикання, що буде ідентифіковано першим проходом механізму причинності Hopper. Кожен раз, коли зловмисник переходить на новий набір облікових даних, Hopper генерує додатковий набір причинно-наслідкових шляхів з новим фокусним стрибком, який відображає логін, який змінив облікові дані. Для будь-якого з цих шляхів, якщо зловмисник не отримує негайний доступ до нового місця призначення під час зміни облікових даних, алгоритми виявлення Hopper (описані нижче) додає цей шлях в список спостереження. Цей крок дозволяє другому проходу механізму причинності продовжити відстеження цих потенційно небезпечних шляхів і місць призначення, до яких вони звертаються в майбутньому.

4.4 Виявлення та оповіщення

Для кожного шляху входу в систему етап генерації попереджень Hopper вирішує, чи є цей шлях доброякісним або підозрілим, і робить одну з трьох дій: генерує попередження, відкидає шлях або додає його в список причинно-наслідкових зв'язків.

Hopper приймає це рішення, з огляду на три вхідних сигнали: шлях входу в систему для класифікації (таблиця 4.2), набір історичних подій входу в систему для вилучення ознак, і наданий користувачем «бюджет», який контролює, скільки щоденних попереджень повинен видавати Hopper. З огляду на ці вхідні дані, Hopper зіставляє шлях зі сценарієм руху: один з чотирьох доброякісних сценаріїв, один з двох сценаріїв атаки, або жоден з цих сценаріїв атаки або доброякісних сценаріїв. Шляхи, відповідні доброякісному сценарієм, не генерують оповіщення. Для інших шляхів, Hopper зіставляє шлях до сценарію атаки і застосовує відповідний сценарій детектор, який витягує набір характеристик для шляху, а потім застосовує або набір правил, або алгоритм оцінки аномалій, щоб визначити, чи є шлях підозрілим, і в цьому випадку видає попередження . Якщо шлях не відповідає ніякому доброякісному сценарієм і не генерує попередження жодним з детекторів, то Hopper додає шлях до свого списку спостереження для подальшого відстеження.

Доброякісні сценарії руху: Hopper спочатку намагається визначити, чи відповідає новий шлях одному з чотирьох сценаріїв доброякісного руху. У першому доброякісному сценарії Hopper відзначає шлях як доброякісний, якщо кожен з його логінів використовує облікові дані причинного користувача (тобто цільове ім'я користувача логінів завжди збігається з ім'ям користувача причини); оскільки ці шляхи не змінюють облікові дані, Hopper відкидає їх. Як описано в Додатку B.1, Hopper також позначає шлях як доброякісний, якщо він відповідає одному з трьох інших доброякісних сценаріїв з низьким рівнем ризику: одноланцюгові шляхи від нових машин або машин в процесі перепрофілювання для нового власника; однохопові шляхи, що використовують службові облікові записи з

обмеженимидозволами; і шляхи, які відвідують (наданий доменом) набір захищених машин, які не дозволяють перемикання облікових даних.

Сценарії атак: Якщо шлях не відповідає жодному з цих доброякісних сценаріїв, то Hopper визначає чи відповідає він таким двома сценаріями атак; якщо так, то застосовується відповідний детектор, щоб побачити чи повинен він видати попередження. По-перше, якщо шлях містить вхід, який перемикає облікові дані і механізм причинно-наслідкових зв'язків дає нам високу ступінь довіри до передбачуваного шляху, Hopper застосовує простий набір правил на основі специфікації, щоб класифікувати шлях як підозрілий чи ні; ми називаємо цей шлях "чистий перемикач облікових даних". Однак, через недосконаліть інформації, Hopper не завжди впевнений у висновках, зроблених його механізмом причинності. Наприклад, на рисунку 4.1, якщо L_2 і L_3 відбулися незадовго до L_4, механізм причинності Hopper НЕ буде впевнений, чи слід йому причинно пов'язувати L_4 з L_2 або L_3, тому він виведе обидва можливих шляхи. Хоча шлях L_3 - L_4 містить перемикач повноважень, інший шлях (L_2 - L_4) не містить його, тому Hopper не може бути впевнений, чи представляє це справжнє перемикання повноважень або просто неточність в умовиводах про причинності. Через невизначеність щодо цих передбачуваних шляхів, другий детектор оцінює, наскільки підозрілим і аномальним є кожен такий шлях за допомогою імовірнісного алгоритму підрахунку балів і попереджає, якщо шлях має одну з найбільш підозрілих оцінок в недавній історії.

Особливості траєкторії: Hopper використовує набір історичних "навчальних" логінів (наприклад, за останні 30 днів) для вилучення наступних трьох характеристик для шляху P. Нехай A означає початкову машину шляху, Z - кінцевий пункт призначення, L_i - вхід в систему в P, який переключився з облікових даних причинного користувача на новий набір облікових даних, а L_{i-1} - вхід, що безпосередньо передує переключенню облікових даних.

Спочатку Hopper обчислює історичну частоту краю L_{i-1}, де частота дорівнює кількість днів, коли в навчальних даних стався успішний вхід з точно таким же

вихідним, кінцевим і цільовим ім'ям користувача. По-друге, Hopper обчислює історичну частоту краю для кожного наступного стрибка в залишку P і бере найменше значення частоти серед цих стрибків; тобто тобто історичну частоту самого рідкісного входу, починаючи з L_i і до останнього стрибка шляху. Нарешті, Hopper обчислює кількість історичних днів, коли будь-який успішний шлях входу з'єднує машину A і машину Z.

Інтуїтивно зрозуміло, що ці три характеристики оцінюють рідкість шляху в залежності від довжини, дозволяючи Hopper відстежувати і порівнювати шляхи атаки довільної довжини. Перша функція допомагає Hopper визначити, коли атакуючий переходить на машину, до якої також мають доступ користувачі з додатковими привілегіями: користувачі зазвичай отримують доступ до обмеженого набору серверів, пов'язаних з їх робочими функціями, і кожен внутрішній сервер зазвичай надає конкретну послугу (тобто функціональність команди). В результаті, логіни, що дозволяють зловмисникові отримати доступ до облікового запису інших ролей і посадових функцій, можуть виникати рідко. Друга особливість відображає інтуїцію, згідно з якою користувачі зазвичай мають спільні робочі процеси і відповідні шляхи входу в систему для доступу до серверів, які мають відношення до конкретних функцій для доступу до серверів, що мають відношення до їх роботи; тому незвично, що логіни двох різних команд відбуваються з одного і того ж проміжного сервера. Зокрема, ми очікуємо, що шляхи входу на важливі сервери зазвичай відбуватимуться з невеликого набору вихідних машин (наприклад, або з клієнтських машин, або з обмежених серверів, доступ до яких мають лише такі ж привілейовані користувачі); така поведінка призведе до того, що шляхи атак від інших користувачів матимуть підозріле (низька) значення для другої характеристики. Остання характеристика відображає загальну рідкість шляху, не залежну від довжини: шлях більш підозрілий, якщо його кінцеві машини рідко мали шлях, що з'єднує їх в попередній історії.

Алгоритм 2. Алгоритм оцінки аномалій Hopper

Sub-Score (P, F, L):

1: $Sum_F \leftarrow 0$

2: N ← 0 (загальна кількість справжніх причинно-наслідкових шляхів)

3: для кожного шляху X в L зробити:

4: якщо P має менше значення для F, ніж X:

5: $Sum_F \leftarrow Sum_F + C_x$

де C_x = ймовірність шляху для X

6: $N \leftarrow N + C_x$,

7: $Sub\text{-}Score_F \leftarrow Sum_F / N$

$$Score(P, L): \prod_F Sub - Score(P, F, L)$$

AlertGen (P, A (історичні оповіщення), L (історичні шляхи)):

1: для кожного шляху X в A зробіть:

2: якщо Score (P, L) ≥ Score (X, L):

3: Оповіщення на P

Скоринг аномалій: З огляду на шлях P і його особливості, алгоритм 2 показує процедуру оцінки аномалій, яку використовує Hopper для кількісної оцінки підозрілості шляху і прийняття рішення про попередження. Інтуїтивно зрозуміло, що алгоритм оцінки аномалій Hopper генерує попередження для P, якщо він має один з найбільш підозрілих наборів ознак за останній час.

Алгоритм оповіщення Hopper (AlertGen) приймає три вхідних сигнали: шлях для оцінки (P), набір історичних шляхів (L) для обчислення. шляхів (L) для обчислення оцінки аномалії P і набір історичних попереджень (A) для шляхів з неясною причинно-наслідковим зв'язком. Щоб сформувати набір історичних шляхів для оцінки аномалії P, Hopper виконує ітерації по кожному логіну в історичних навчальних даних і використовує механізм причинності Hopper для створення набору всіх двуххопових шляхів для кожного історичного входу; сукупна колекція всіх цих двуххопових шляхів утворює набір історичних шляхів (L). Для ефективності, Hopper може обчислювати це як пакетне завдання на початку кожного тижня і повторно

використовувати цей набір історичних шляхів для підрахунку за весь тиждень. історичний набірскладається з B × H найбільш підозрілих шляхів протягом історичного вікна, де H - це кількість днів в історичному вікні.

З огляду на ці три вхідних даних, Hopper обчислює загальний бал аномалії для P, який представляє собою частку історичних шляхів, де P мав більше (або стільки ж) підозрілих значень ознак. Потім Hopper порівнює оцінку аномалії P з оцінками для всіх історичних шляхів попередження та генерує попередження для P, якщо його оцінка перевищує оцінку будь-якого шляху попередження; тобто якщо P принаймні так само підозрілий, як і попередній шлях попередження, то Hopper видає попередження. Якщо алгоритм оцінки Hopper не генерує попередження для P, він додає шлях до свого списку спостереження для додаткового моніторингу та можливої реклассификации.

Обчислення балів: Концептуально, оцінка аномальності шляху P відповідає кумулятивному хвості ймовірності: наскільки більш підозрілим (малоймовірним) є шлях P по відношенню до тих шляхів, де добросовісні користувачів? Як показано в алгоритмі 2, Hopper обчислює цей бал шляхом обчислення подшкали для кожної з характеристик шляху, а потім перемножує їх для отримання загальної оцінки. Суб-оцінки кожної ознаки та обчислює частку історичних шляхів де P мав більш підозріле значення ознаки.

Однак обчислення цієї суб-оцінки ускладнюється обмеженнями у висновку шляхів Hopper: історичні шляхи (L), які генерує Hopper, містять безліч помилково припущених шляхів, які не відповідають істинному причинному шляху.

Щоб врахувати ці типи зміщення розподілу, що вносять неточністі для кожної суб-оцінки обчислюється зважена частка числа історичних шляхів в яких P мав більш підозріле значення ознаки. Зокрема, коли Hopper генерує набір шляхів для входу L_i в історичному наборі даних, він анотує кожен шлях з "Імовірністю шляху" (позначається як C), яка дорівнює 1, поділеній на загальну кількість причинно-наслідкових шляхів, які Hopper визначив для L_i. Коли Hopper обчислює суб-оцінку

для P, він використовує C для зниження ваги впливу кожного історичного шляху на суб-оцінку P. При наївному обчисленні без цього зважування, Hopper збільшив би суб-оцінку P на 1 для кожного історичного шляху, де у P було більш підозріле значення ознаки. Замість цього, Hopper фактично збільшує суб-оцінку P на зважену частку 1 × C для кожного історичного шляху, де P має більш підозрілу характеристику.

4.5 Оцінка

Ми оцінили Hopper, використовуючи наш 15-місячний набір даних з Dropbox, вимірюючи швидкість виявлення (частка виявлених атак) і загальний обсяг генеруємих ним попереджень. Наші дані не містять відомих атак бічного переміщення, але в них є одна атака бічного переміщення, проведена професійної "червоною командою" Dropbox. Крім того, для більш ретельної оцінки ми створили і впровадили в наш набір даних реалістичний набір з 326 змодельованих атак; як описано в Додатку B.2, ці атаки охоплюють широкий спектр атак і рівнів скритності зловмисників.

Таблиця 4.3 - Зведення ефективності виявлення атак Hopper.

Сценарій атаки	Виявлені атаки (чітка причинність)	Виявлені атаки (неясна причинність)	Швидкість виявлення
Розвідувальна атака	30	7	*37/41
Дослідницька атака (прихована)	6	62	68/69
Агресивне росповсюдження	34	4	*38/41
Агресивне росповсюдження	35	34	69/69
Цільова атака	28	10	*38/40
Цільова атака (прихована)	5	57	62/67
Атака Червоної команди	0	1	1/1
Усі атаки	138	174	312/327

Ми розділили наш набір даних на вікно навчання (з 1 січня 2019 року по 1 березня 2019 роки), яке ми використовували для бутстрапа компонентів вилучення ознак і оцінки Hopper, що вимагають історичних даних, і 13-місячне вікно оцінки (з 1 березня 2019 року по 1 квітня 2020 року). За час оцінки набір даних містить 713 617 425 успішних входів в систему і 2 941 173 входу після застосування кроків фільтрації даних Hopper. Як описано нижче, ми використовували логіни в цьому вікні оцінки для розрахунку коефіцієнта помилкових спрацьовувань і коефіцієнта виявлення Hopper. Для будь-якого компонента виявлення, що вимагає історичних даних для навчання, ми використовували вікно попередніх 30 днів; для нашого алгоритму оцінки аномалій ми використовували початковий бюджет в 5 попереджень в день.

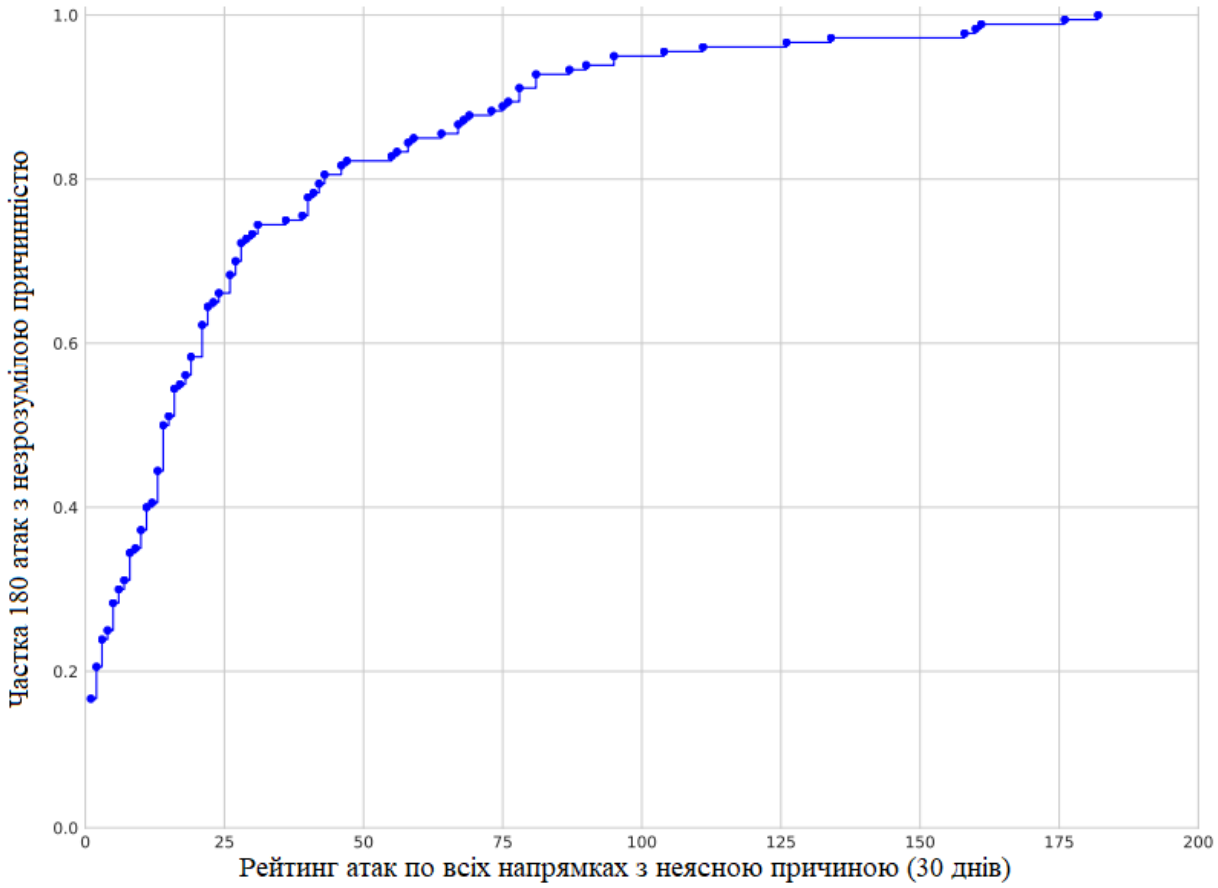


Рисунок 4.3 - Рейтинг шляхів атак з неясною причинно-наслідковим зв'язком, щодо всіх попереджень, згенерованих Hopper протягом 30-денного вікна.

Коефіцієнт виявлення атак: Для кожної з 326 атак, синтезованих нашою системою, ми ввели логіни атаки в наші оціночні дані і запустили Hopper в день (дні),

коли сталася атака. Для вправи "червоної команди" ми вивчили попередження, які видав Hopper в день атаки. Якщо Hopper генерував попередження для будь-якого шляху атаки, розпочатої симулювати зловмисником або червоною командою, то ми вважали, що Hopper успішно виявив атаку.

Як показано в останній колонці таблиці 4.3, Hopper успішно виявив в цілому 312 атак, включаючи атаку бічного переміщення, проведену експертної "червоною командою" Dropbox. Hopper виявив 138 атак за допомогою свого детектора шляхів з явним перемиканням. У всіх цих атаках (таблиця 4.3, колонка 2) імітований зловмисник використав новий набір облікових даних при вході в систему зі своєї первісної машини-плацдарму, або з сервера, до якого легітимний користувач (з новими обліковими даними) недавно не звертався, що дозволило Hopper визначити траєкторію руху, на якій зловмисник переключився на використання несподіваного набору облікових даних. Оскільки цей компонент детектора Hopper не покладається на частоту окремих ребер графа, створених рухом зловмисника, він успішно виявив кілька прихованих атак, в яких противник намагався ухилитися від виявлення переміщаючись тільки між машинами з попередньою історією з'єднань.

Однак більшість змодельованих атак створювали шляху з неясною причинно-наслідковим зв'язком, або тому, що атака швидко використовувала нові дані, які нещодавно використовувалися на сервері, або тому, що атака імітувала таємного противника, який намагався уникнути виявлення, використовуючи тільки нові облікові дані з машин, де легітимний користувач був недавно або в даний час активний. Виявлення цих шляхів покладається на детектор аномалій Hopper. При нашому первинному бюджеті в 5 попереджень в день, Hopper успішно ідентифікував 174 з цих атак, а також бічну атаку, змодельовану "червоною командою" Dropbox.

Помилкові негативні результати і чутливість до бюджету: З 15 атак, які Hopper не зміг виявити, ми виявили, що Hopper пропустив 9 атак через помилки атрибутів в даних для входу в систему, які ми використовували для цього дослідження. Для кожного з цих 9 помилкових спрацьовувань логіни атак, синтезовані нашою

системою, мали неправильні мітки клієнта і сервера і відсутня або неправильна інформація про володіння машиною. Коли ми замінили цю неповну або відсутню інформацію в логіни атак більш точним маркуванням атрибутів (отриманих з сучасних джерел даних в Dropbox), ми виявили, що Hopper може успішно виявити усі 9 таких помилково негативні результатів за допомогою свого чіткого детектора перемикання облікових даних.

Крім того, Hopper не зміг виявити 6 прихованих атак при щоденному бюджеті в 5 попереджень. Якщо ми збільшити бюджет детектора аномалій Hopper на 1 додаткове сповіщення в день, Hopper зможе успішно виявити всі, крім одного з цих помилково негативні результати; збільшення щоденного бюджету на 2 оповіщення в день дозволить Hopper виявити всі 6 помилкових спрацьовувань. Як показано на рисунку 4.3, Hopper відносить більше 75% цих атак з неясною причинно-наслідковим зв'язком до 30 кращих шляхів за весь місяць в якому сталася атака, і більше 90% атак в 120 кращих шляхах за весь місяць. (Тобто, в середньому, найбільш підозріле попередження і в межах чотирьох кращих попереджень в день відповідно). Цей розподіл показує, що набір функцій і алгоритм оцінки Hopper послідовно ідентифікують навіть ці приховані атаки з низькою кількістю помилкових спрацьовувань.

Загальна кількість попереджень і помилкових спрацьовувань: Щоб розрахувати коефіцієнт помилкових спрацьовувань Hopper, ми запустили Hopper на всіх легітимних логінах за кожен з 396 днів в наших оціночних даних і підсумовували попередження, які він видав. За винятком однієї атаки "червоної команди", в базі даних інцидентів Dropbox не виявлено ніяких відомих випадків бокового переміщення під час нашого вікна оцінки. Таким чином, ми консервативно позначили всі попередження, згенеровані Hopper на даних для входу в систему, як помилкові спрацьовування, якщо вони не були пов'язані з діяльністю "червоної команди".

При щоденному бюджеті в 5 попереджень для свого детектора неясною причинності, Hopper видав 3544 попередження протягом 396 днів: в середньому 9

попереджень в день і загальний коефіцієнт помилкових спрацьовувань 0,000005 для 713 мільйонів необроблених логінів наших даних. З усіх попереджень детектор шляхів з явним перемиканням повноважень видав 2 399 попереджень, а решта 1 145 попереджень були отримані від детектора аномалій Hopper. В деякі дні детектор аномалій Hopper видавав менше 5 попереджень, тому що (1) не в кожен день було 5 досить підозрілих шляхів з невизначеністю причинно-наслідкового зв'язку (наприклад, у вихідні та святкові дні), і (2) наша кластеризация попереджень привела до того, що в деякі дні було менше попереджень.

Таблиця 4.4 - Порівняння ефективності детекторів Hopper та SAL[71] протягом 13 місяців.

Детектор	Швидкість виявлення	Мін. загальна кількість сповіщень
SAL (CCS 2017) [71]	156/327	3556
	312/327	28771
Hopper	312/327	3544

Аналізуючи випадкову вибірку попереджень Hopper, ми визначили кілька загальних доброякісних причин, які пояснюють ці ймовірні помилкові спрацьовування. По-перше, наш детектор аномалій видає набір найбільш підозрілих попереджень кожен день; проте багато з цих шляхів відображають неточність в алгоритмі причинно-наслідкових зв'язків Hopper. Зокрема, ми помітили, що багато днів включають попередження для шляхів, які включають в себе потенційне перемикання повноважень між одним системним адміністратором (причинним користувачем) і іншим системним адміністратором (цільовим користувачем), що відбувається через декілька машин, до яких ці системні адміністратори часто часто отримують доступ і ініціюють вхід в систему. Оскільки ці шляхи включають тільки облікові дані адміністратора, Hopper може зменшити кількість помилкових спрацьовувань, відфільтрувавши їх, так як будь-який перемикання між ними швидше за все, надає обмежений додатковий доступ. По-друге, багато хто з наших

попереджень відповідають входам з клієнтських машин, які (1) "перемикаються" з облікових даних власника на використання рідко використовуваного облікового запису служби або (2) відображають виконання системним адміністратором сценарію повторного надання доступу до існуючого ноутбука, щоб перепризначити його новому користувачеві. У Додатку В.3 ці помилкові спрацьовування описані більш детально.

Порівняння з попереднім рівнем техніки

При порівнянні продуктивності Hopper з найбільш успішною попередньою роботою - детектором структурно-аномальних входів (SAL)[71]. На високому рівні SAL виявляє атаки бічного переміщення, генеруючи набір попереджень-кандидатів, що складається з усіх входів, які проходять через рідкісне ребро в графі входів (де рідкість - це поріг, що задається користувачем). Далі SAL вивчає набір "доброякісних шаблонів входу", який використовує додаткові властивості машин і користувачів, що беруть участь в вході (наприклад, тип машини і команда користувача). Потім SAL видає попередження для кожного кандидата, яке не відповідає доброякісному шаблоном входу в систему. (У додатку В.4 подається детальніший опис SAL).

При застосуванні SAL зі змінним двомісячним вікном навчання на всіх постфільтрованних входах в систему в нашому вікні оцінки. Як описано в Додатку В.4, SAL приймає два порога, заданих користувачем, для навчання і класифікації, відповідно. Для розрахунку результатів, ми досліджували ряд комбінацій порогів для SAL і вибрали комбінації порогів, які дали мінімальну кількість попереджень для виявлення (1) всіх атак в наших даних і (2) половини атак в наших даних. Ми повідомляємо про кількість попереджень, отриманих SAL після дедуплікації попереджень, щоб включити тільки одну грань (джерело, пункт призначення і цільовий користувач) в день, і ми вважаємо SAL успішним, якщо він видав попередження для будь-якої з шкідливих граней на шляху атаки. Додаток В.4 описує цю процедуру більш детально і показує продуктивність SAL при кожній комбінації, яку ми пробували. Велика різниця в обсягах попереджень, що видаються SAL і

Hopper, обумовлена фундаментальними відмінностями в тому, як вони визначають шляхи бічного переміщення. SAL використовує традиційний підхід до виявлення аномалій, який теоретично може виявити будь-яку атаку бічного переміщення при досить низьких порогових значеннях. Хоча цей більш загальний підхід може виявити атаки, які не може виявити Hopper, наші результати показують, що Hopper може виявити переважний клас бічних переміщень в нашій моделі загроз з більш ніж у 8 разів меньшу кількістю помилкових спрацьовувань. Більш того, Hopper може успішно виявляти ці атаки без необхідності ручного налаштування порогових значень, яка потрібна для оптимальної роботи SAL.

ВИСНОВКИ

У кваліфікаційній роботі було описано метод захисту корпоративної мережі від ускладнених атак. Запропонований метод дозволяє блокувати три основні методи корпоративних атак, а саме: цільовий фішинг, бічний фішин та бічне переміщення.

Аналізуючи те, що організації регулярно стають жертвами різних складних атак, що призводить до фінансових збитків в мільярди доларів, крадіжці конфіденційних даних і порушення роботи критично важливої інфраструктури і послуг [6, 7, 8, 47, 102]. Дана робота прокладає багатообіцяючий шлях вперед через новий набір заснованих на даних ідеях і методах, які можуть пом'якшити ці руйнівні загрози.

Основні результати кваліфікаційної роботи:

1. Досліджено методи виявлення вдосконалених атак і їх проблеми з перевагами та недоліками, щоб визначити найпоширеніші методи виявлення.

2. Розглянуто популярніші методи атак та способи зменшення помилкових спрацювань.

3. Розроблено метод захисту для пом’якшення і виявлення атак в корпоративній мережі.

4. Розроблено ментальні моделі, які покращать розуміння та здатнійсть виявляти фішингові атаки.

5. Розроблено новий набір емпіричних результатів та алгоритмів виявлення.

6. Протестовано розроблену систему виявлення.

Перелік джерел посилань

1. A. Breeden, S. Chan, and N. Perlroth. Macron campaign says it was target of ‘mas-sive’ hacking attack. Режим доступу: https://www.nytimes.com/2017/05/05/ world/europe/france-macron-hacking.html, 02.05.17.

2. CloudMark. Spear phishing: The top ten worst cyber attacks.Режим доспупу: https://blog.cloudmark.com/wp-content/uploads/2016/01/ cloudmark top ten infographic.png.

3. Д. Олівейра, Х. Роча, Х. Янг, Д. Елліс, С. Доммараджу, М. Мурадоглу, Д. Вейр, А. Соліман, Т. Лін та Н. Ебнер. Розбір фішингових листів для літніх і молодих людей: про взаємодію зброї впливу та сфер життя у прогнозуванні схильності до фішингу. На конференції ACM про людський фактор у обчислювальних системах (CHI), 2017.

4. Р. Лі, М. Ассанте та Т. Конвей. Аналіз кібератаки на українську електромережу. Центр обміну та аналізу інформації про електроенергію (E-ISAC), 2016.

5. L. Vaas. How hackers broke into John Podesta, DNC Gmail accounts. Режим доступу: https://nakedsecurity.sophos.com/2016/10/25/how-hackers-broke-into- john-podesta-dnc-gmail-accounts/, 09.10.16.

6. J. Finkle and S. Heavey. Target says it declined to act on early alert of cyber breach. Режим доступу: http://www.reuters.com/article/us-target-breach-idUSBREA2C14F20140313, 05.03.14.

7. R. Hackett. Anthem, a major health insurer, suffered a massive hack. Режим доступу: http://fortune.com/2015/02/05/anthem-suffers-hack/, 25. 02.15.

8. E. Nakashima. Chinese breach data of 4 million federal workers. Режим доступу: https://www.washingtonpost.com/world/national-security/chinese- hackers-breach-federal-governments-personnel-office/2015/06/ 04/889c0e52-0af7-11e5-95fd-d580f1c5d44e story.html, 14.06.15.

9. A. Picchi. Ransomware’s mounting toll: Delayed surgeries and school closures. Режим доступу: https://www.cbsnews.com/news/ransomware-attack-621-hospitals-

cities-and-schools-hit-so-far-in-2019/, 16.10.19.

10. CERT. Ransomware. Режим доступу: https://www.us-cert.gov/Ransomware, 2019.

11. A. Press. Ransomware attack cripples san bernardino city unified school district's computer system. Режим доступу: https://abc7.com/ransomware-attack-cripples-san-bernardino-school-districts-computer-system/5635301/, 02.10.19.

12. N. Wetsman. Woman dies during a ransomware attack on a german hospital. Режим доступу: https://www.theverge.com/2020/9/17/21443851/death-ransomware-attack-hospital-germany-cybersecurity, 05.09.20.

13. A. Peterson. The sony pictures hack, explained. Режим доступу: https:// www.washingtonpost.com/news/the-switch/wp/2014/12/18/the- sony-pictures-hack-explained/, 15.12.14.

14. CarbonBlack. Звіт про глобальну загрозу: Кібератаки натупного покоління. Режим доступу: https: //www.carbonblack.com/resources/threat-research/year-of-the-next-gen-cyberattack/, 2019.

15. B. E. Strom, A. Applebaum, D. P. Miller, K. C. Nickels, A. G. Pennington, and C. B. Thomas. Mitre att&ck: Design and philosophy. Technical report, 2018.

16. О технологии SPF.Режим доступу: https://support.google.com/a/ answer/33786?hl=ru, 05.09.21.

17. Domainkeys identified mail. Режим доступу: https://www.sparkpost.com/ resources/email-explained/dkim-domainkeys-identified-mail/

18. D. Organization. DMARC. https://dmarc.org/, 2016.

19. С. Ніколс. Минуло 15 років, а ми досі повідомляємо про атаки омографів. Режим доступу: https://www.theregister.com/2020/03/04/ homograph attacks still happening/, 09.03.20.

20. D. Wind. Sophisticated Spear Phishing Campaigns using Homograph At- tacks. https://www.offensity.com/de/blog/sophisticated-spear- phishing-campaigns-using-homograph-attacks/, 16.05.19.

21. Дж. Чен, В. Пакссон та Дж. Цзян. Композиція вбиває: приклад автентифікації

відправника електронної пошти. У симпозіумі з безпеки USENIX (USENIX), 2020.

22. X. Xy і Г. Ван. Наскрізні вимірювання атак підробки електронної пошти. У симпозіумі з безпеки USENIX (USENIX), 2018.

23. J. Dutson, D. Allen, D. Eggett, and K. Seamons. Don't punish all of us: Measuring user attitudes about two-factor authentication. In IEEE European Symposium on Security and Privacy Workshops (EuroS&PW), 2019.

24. J. Reynolds, N. Samarin, J. Barnes, T. Judd, J. Mason, M. Bailey, and S. Egelman. Empirical measurement of systemic 2fa usability. In USENIX Security Symposium (USENIX), 2020.

25. L. Constantin. Phishing attacks that bypass 2-factor authentication are now easier to execute.Режим доступу : https://www.csoonline.com/ article/3399858/phishing- attacks-that-bypass-2-factor-authentication-are-now-easier- to-execute.html, 06.19.

26. J. S. Railton and K. Kleemola. London calling: Two-factor authentication phishing from iran. https://citizenlab.org/2015/08/iran two factor phishing/, August 2015.

27. D. R. Staff. Fbi: Phishing can defeat two-factor authentication.Режим доступу:https://www.darkreading.com/attacks-breaches/fbi-phishing-can- defeat-two-factor-authentication/d/d-id/1336070, 10.19.

28. C. Гарера, Н. Провос, М. Чу та А. Д. Рубін. Структура для виявлення та вимірювання фішингових атак. У працях семінару ACM 2007 року з повторюваного шкідливого коду, 2007.

29. P. Prakash, M. Kumar, R. R. Kompella, and M. Gupta. Phishnet: predictive blacklisting to detect phishing attacks. In INFOCOM, 2010 Proceedings IEEE, pages 1–5. IEEE, 2010.

30. S. Sheng, B. Wardman, G. Warner, L. Cranor, J. Hong, and C. Zhang. An empirical analysis of phishing blacklists. In Sixth Conference on Email and Anti-Spam (CEAS), 2009.

31. C. Whittaker, B. Ryner, and M. Nazif. Large-scale automatic classification of

phishing pages. In NDSS, volume 10, 2010.

32. Y. Zhang, J. I. Hong, and L. F. Cranor. Cantina: a content-based approach to detecting phishing web sites. In Proceedings of the 16th international conference on World Wide Web, pages 639–648. ACM, 2007.

33. A. Oest, P. Zhang, B. Wardman, E. Nunes, J. Burgis, A. Zand, K. Thomas, A. Doupé, and G.-J. Ahn. Sunrise to sunset: Analyzing the end-to-end life cycle and effectiveness of phishing attacks at scale. In USENIX Security Symposium (USENIX), 2020.

34. A. Oest, Y. Safaei, P. Zhang, B. Wardman, K. Tyers, Y. Shoshitaishvili, A. Doupé, and G.-J. Ahn. PhishTime: Continuous Longitudinal Measurement of the Effectiveness of Anti-phishing Blacklists. In USENIX Security Symposium (USENIX), 2020.

35. P. Zhang, A. Oest, H. Cho, R. Johnson, B. Wardman, S. Sarker, A. Kpravelos, T. Bao, R. Wang, Y. Shoshitaishvili, A. Doupé, and G.-J. Ahn. CrawlPhish: Large-scale Analysis of Client-side Cloaking Techniques in Phishing. In IEEE Symposium on Security and Privacy(S&P), 2021.

36. A. Bergholz, J. H. Chang, G. Paaß, F. Reichartz, and S. Strobel. Improved phishing detection using model-based features. In CEAS, 2008.

37. S. Duman, K. Kalkan-Cakmakci, M. Egele, W. Robertson, and E. Kirda. Emailpro- filer: Spearphishing filtering with header and stylometric features of emails. In Computer Software and Applications Conference (COMPSAC), 2016 IEEE 40th Annual, pages 408–416. IEEE, 2016.

38. H. Gascon, S. Ullrich, B. Stritter, and K. Rieck. Reading between the lines: content-agnostic detection of spear-phishing emails. In International Symposium on Research in Attacks, Intrusions, and Defenses (RAID), 2018.

39. I. Fette, N. Sadeh, and A. Tomasic. Learning to detect phishing emails. In Proceedings of the 16th international conference on World Wide Web, pages 649–656. ACM, 2007.

40. M. Khonji, Y. Iraqi, and A. Jones. Mitigation of spear phishing attacks: A content-based authorship identification framework. In Internet Technology and Secured Transactions (ICITST), 2011 International Conference for, pages 416–421. IEEE, 2011.

41. M. Zhao, B. An, and C. Kiekintveld. Optimizing personalized email filtering thresholds to mitigate sequential spear phishing attacks. In AAAI, pages 658–665, 2016.

42. G. Stringhini and O. Thonnard. That ain't you: Blocking spearphishing through behavioral modelling. In Detection of Intrusions and Malware, and Vulnerability Assessment, pages 78–97. Springer, 2015.

43. S. Axelsson. The base-rate fallacy and the difficulty of intrusion detection. ACM Transactions on Information and System Security (TISSEC), 2000.

44. A. Cidon, L. Gavish, I. Bleier, N. Korshun, M. Schweighauser, and A. Tsitkin. High preci- sion detection of business email compromise. In USENIX Security Symposium (USENIX),2019.

45. P. Bright. Spearphishing + zero-day: RSA hack not "extremely sophisti- cated". http://arstechnica.com/security/2011/04/spearphishing-0- day-rsa-hack-not-extremely-sophisticated/, April 2011.

46. W. R. Marczak, J. Scott-Railton, M. Marquis-Boire, and V. Paxson. When governments hack opponents: A look at actors and technology. In USENIX Security, pages 511–525, 2014.

47. S. Reilly. Records: Energy department struck by cyber attacks. http://www.usatoday.com/story/news/2015/09/09/cyber-attacks-doe- energy/71929786/, September 2015.

48. M. S. Schmidt and D. E. Sanger. Russian hackers read obama's unclassified emails, officials say. http://www.nytimes.com/2015/04/26/us/russian-hackers-read- obamas-unclassified-emails-officials-say.html, Apr 2015.

49. S. Sheng, M. Holbrook, P. Kumaraguru, L. F. Cranor, and J. Downs. Who falls for phish?: a demographic analysis of phishing susceptibility and effectiveness of

interventions. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, pages 373–382. ACM, 2010.

50. E. Bursztein and V. Eranti. Internet-wide efforts to fight email phishing are working. https://security.googleblog.com/2013/12/internet-wide- efforts-to-fight-email.html, Feb 2016.

51. H. He and E. A. Garcia. Learning from imbalanced data. IEEE Transactions on knowledgeand data engineering, 21(9):1263–1284, 2009.

52. N. Chawla, N. Japkowicz, and A. Kotcz. Editorial: special issue on learning from imbal- anced data sets. ACM SIGKDD Explorations Newsletter, 6(1):1–6, 2004.

53. V. Chandola, A. Banerjee, and V. Kumar. Anomaly detection: A survey. ACM Comput. Surv., 41(3):15:1–15:58, 2009.

54. A. Lazarevic, L. Ertoz, V. Kumar, A. Ozgur, and J. Srivastava. A comparative study of anomaly detection schemes in network intrusion detection. In Proceedings of the 2003 SIAM International Conference on Data Mining, pages 25–36. SIAM, 2003.

55. S. Abu-Nimeh, D. Nappa, X. Wang, and S. Nair. A comparison of machine learning tech- niques for phishing detection. In Proceedings of the anti-phishing working groups 2nd annual eCrime researchers summit, pages 60–69. ACM, 2007.

56. A. Cidon, L. Gavish, I. Bleier, N. Korshun, M. Schweighauser, and A. Tsitkin. High Preci-sion Detection of Business Email Compromise. In Proc. of 28th Usenix Security, 2019.

57. X. Hu, B. Li, Y. Zhang, C. Zhou, and H. Ma. Detecting Compromised Email Accounts from the Perspective of Graph Topology. In Proc. of 11th ACM CFI, 2016.

58. E. Bursztein, B. Benko, D. Margolis, T. Pietraszek, A. Archer, A. Aquino, A. Pitsillidis, and S. Savage. Handcrafted Fraud and Extortion: Manual Account Hijacking in the Wild. InProc. of 14th ACM IMC, 2014.

59. J. Onaolapo, E. Mariconti, and G. Stringhini. What Happens After You Are Pwnd: Under-standing the Use of Leaked Webmail Credentials in the Wild. In Proc. of 16th ACM IMC, 2016.

60. K. Thomas, F. Li, C. Grier, and V. Paxson. Consequences of Connectivity: Characterizing Account Hijacking on Twitter. In Proc. of 21st ACM CCS, 2014.

61. J. Dunagan, A. X. Zheng, and D. R. Simon. Heat-ray: combating identity snowball attacks using machinelearning, combinatorial optimization and attack graphs. In ACM SIGOPS 22nd Symposium on Operating Systems Principles (SOSP), 2009.

62. Google. Класифікація: ROC та AUC. https://developers.google.com/machine-learning/crash-course/classification/roc-and-auc, 2019.

63. S. Freitas, A. Wicker, D. H. Chau, and J. Neil. D2m: Dynamic defense and modeling of ad- versarial movement in networks. In Proceedings of the 2020 SIAM International Conference on Data Mining, 2020.

64. A. Robbin, R. Vazarkar, and W. Schroeder. Bloodhound: Six degrees of domain admin. https://bloodhound.readthedocs.io/en/latest/index.html/, 2020

65. A. Bohara, M. A. Noureddine, A. Fawaz, and W. H. Sanders. An unsupervised multi- detector approach for identifying malicious lateral movement. In IEEE 36th Symposium onReliable Distributed Systems (SRDS), 2017.

66. A. D. Kent, L. M. Liebrock, and J. C. Neil. Authentication graphs: Analyzing user behaviorwithin an enterprise network. Computers & Security, 2015.

67. Q. Liu, J. W. Stokes, R. Mead, T. Burrell, I. Hellen, J. Lambert, A. Marochko, and W. Cui. Latte: Large-scale lateral movement detection. In IEEE Military CommunicationsConference (MILCOM), 2018.

68. F. Liu, Y. Wen, D. Zhang, X. Jiang, X. Xing, and D. Meng. Log2vec: A Heterogeneous Graph Embedding Based Approach for Detecting Cyber Threats within Enterprise. In Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security, 2019.

69. E. Purvine, J. R. Johnson, and C. Lo. A graph-based impact metric for mitigating lateral movement cyber attacks. In ACM Workshop on Automated Decision Making for Active Cyber Defense, 2016.

70. F. Wilkens, S. Haas, D. Kaaser, P. Kling, and M. Fischer. Towards Efficient

Reconstruction of Attacker Lateral Movement. In Conference on Availability, Reliability and Security - ARES, 2019.

71. H. Siadati and N. Memon. Detecting structurally anomalous logins within enterprise net- works. In ACM SIGSAC Conference on Computer and Communications Security (CCS), 2017.

72. K. Poulsen. Google disrupts chinese spear-phishing attack on senior u.s. officials. https://www.wired.com/2011/06/gmail-hack/, Jul 2011.

73. K. Zetter. Researchers uncover rsa phishing attack, hiding in plain sight. https://www.wired.com/2011/08/how-rsa-got-hacked/, Aug 2011.

74. S. Le Blond, C. Gilbert, U. Upadhyay, M. G. Rodriguez, and D. Choffnes. A broad view ofthe ecosystem of socially engineered exploit documents. In NDSS, 2017.

75. S. Le Blond, A. Uritesc, C. Gilbert, Z. L. Chua, P. Saxena, and E. Kirda. A look at targeted attacks through the lense of an ngo. In USENIX Security, pages 543–558, 2014.

76. A. Cidon. Threat Spotlight: Office 365 Account Takeover — the New "Insider Threat". https://blog.barracuda.com/2017/08/30/threat-spotlight- office-365-account-compromise-the-new-insider-threat/, Aug2017.

77. F. Labs. A sobering day. https://labs.ft.com/2013/05/a-sobering- day/?mhq5j=e6, May 2013.

78. S. Ragan. Office 365 phishing attacks create a sustained insider nightmare for it. https://www.csoonline.com/article/3225469/office-365-phishing- attacks-create-a-sustained-insider-nightmare-for-it.html, Sep 2017.

79. G. Ho, A. Sharma, M. Javed, V. Paxson, and D. Wagner. Detecting credential spearphishingin enterprise settings. In USENIX Security 17, pages 469–485, 2017.

80. FBI. BUSINESS E-MAIL COMPROMISE THE 12 BILLION DOLLAR SCAM, Jul 2018. https://www.ic3.gov/media/2018/180712.aspx.

81. Microsoft Graph: тип ресурсу повідомлення. https://developer.microsoft.com/ en-us/graph/docs/api-reference/v1.0/resources/ message. Accessed:2018-11-01.

82. J. Palme. Common Internet Message Headers. https://tools.ietf.org/html/ rfc2076.

83. D. Hubbard. Cisco Umbrella 1 Million. https://umbrella.cisco.com/blog/2016/12/14/cisco-umbrella-1-million/, Dec 2016.

84. Random forest. https://en.mathof.org/Random forest, 2019.

85. Microsoft. Огляд людей - Outlook Web App. https://support.office.com/ en-us/article/people-overview-outlook-web-app-5fe173cf-e620- 4f62-9bf6-da5041f651bf. Accessed: 2018-11-01.

86. M. Egele, G. Stringhini, C. Kruegel, and G. Vigna. COMPA: Detecting Compromised Accounts on Social Networks. In Proc. of 20th ISOC NDSS, 2013.

87. F. Y. Rashid. Don't like Mondays? Neither do attackers. https:// www.csoonline.com/article/3199997/don-t-like-mondays-neither- do-attackers.html, Aug 2017.

88. W. U. Hassan, M. A. Noureddine, P. Datta, and A. Bates. Omegalog: High-fidelity attack investigation via transparent multi-layer log analysis. In Network and Distributed System Security Symposium, 2020.

89. M. N. Hossain, J. Wang, O. Weisse, R. Sekar, D. Genkin, B. He, S. D. Stoller, G. Fang, F. Piessens, and E. Downing. Dependence-preserving data compaction for scalable forensicanalysis. In USENIX Security 18, 2018.

90. CERT. Advanced persistent threat activity targeting energy and other critical infrastructuresectors. https://www.us-cert.gov/ncas/alerts/TA17-293A, 2017.

91. S. Hawley, B. Read, C. Brafman-Kittner, N. Fraser, A. Thompson, Y. Rozhansky,and S. Yashar. Apt39: An iranian cyber espionage group focused on personal in-formation. https://www.fireeye.com/blog/threat-research/2019/ 01/apt39-iranian-cyber-espionage-group-focused-on-personal- information.html, Jan 2019.

92. NCSC. Joint report on publicly available hacking tools. https://www.ncsc.gov.uk/ report/joint-report-on-publicly-available-hacking-tools, 2018.

93. Mandiant. Apt1: Exposing one of china's cyber espionage units. https: //www.fireeye.com/content/dam/fireeye-www/services/pdfs/ mandiant-apt1-

report.pdf, 2013.

94. Novetta. Operation SMN: Axiom Threat Actor Group Report. http://www.novetta.com/wp-content/uploads/2014/11/Executive Summary- Final1.pdf, Nov 2014.

95. C. T. U. R. Team. Bronze union: Cyberespionage persists despite disclosures. https://www.secureworks.com/research/bronze-union, Jun 2017.

96. F. Plan, N. Fraser, J. O'Leary, V. Cannon, and B. Read. Apt40: Examining a china-nexus espionage actor. https://www.fireeye.com/blog/threat-research/2019/ 03/apt40-examining-a-china-nexus-espionage-actor.html, Mar 2019.

97. S. Gatlan. Microsoft shares tactics used in human-operated ransomware attacks. https://www.bleepingcomputer.com/news/security/microsoft- shares-tactics-used-in-human-operated-ransomware-attacks/, Mar 2020.

98. L. Tung. Ransomware: These sophisticated attacks are delivering 'devastating' payloads, warns microsoft. Режим доступу: https://www.zdnet.com/article/ransomware-these- sophisticated-attacks-are-delivering-devastating-payloads- warns-microsoft/, Mar 2020.

99. A. Dahan. Operation cobalt kitty, 2017.

100. W. U. Hassan, A. Bates, and D. Marino. Tactical provenance analysis for endpoint detection and response systems. In IEEE Symposium on Security & Privacy 20, 2020.

101. M. N. Hossain, S. Sheikhi, and R. Sekar. Combating dependence explosion in forensic analysis using alternative tag propagation semantics. In IEEE Symposium on Security & Privacy 20, 2020.

102. T. C. of Economic Advisors. The cost of malicious cyber activity to the u.s. economy. https://www.whitehouse.gov/wp-content/uploads/2018/03/The- Cost-of-Malicious-Cyber-Activity-to-the-U.S.- Economy.pdf, Mar2018.

103. MITRE. Mitre att&ck mat. https://attack.mitre.org/, 2015–2019.

104. Retraining models on new data. https://docs.aws.amazon.com/machine-learning/latest/dg/retraining-models-on-new-data.html, 2019.

ДОДАТОК А - ВИЯВЛЕННЯ АТАК ЦІЛЬОВОГО ФІШИНГУ ОБЛІКОВИХ ДАНИХ ТА ХАРАКТЕРИСТИКА БІЧНОГО ПЕРЕМІЩЕННЯ

А.1 Вектори ознак і компаратора для кожного субдетектора

У таблицях А.1, А.2, А.3 представлені ознаки, які використовує кожен з наших субдетекторов цільового фішингу, а також компаратори, які DAS використовує для кожної ознаки, щоб обчислити аномалію події. Кожен компаратор відображає монотонну природу кожної числової ознаки: компаратор, рівний «≤» висловлює, що ми очікуємо, що менші значення ознаки будуть вказувати на більш підозрілі подія, в той час як компаратор «≥» вказує на те, що великі значення відповідають більш підозрілим подіям. «Host» URL означає повне доменне ім'я (FDQN) URL.

Таблиця А.1 - Зведений вектор ознак для поддетектора спуферів імен та "підозрілість", які ми надаємо DAS для кожної ознаки (розділ 1).

Особливості спуфера імен	Порівняння для DAS
Хост (FQDN) URL-адреси на яку натиснули	≤
Відвідування хоста натисканням на URL до отримання листа	≤
Тижнів, коли «Від» надіслав електронний лист протягом ≥ 5 днів	≥
Днів, коли ім’я «Від» та адреса «Від» з'явилися разом в електронних листах	≤

Таблиця А.2 - Зведений вектор ознак для нашого піддетектора раніше не помічених зловмисників і компаратор «підозрілості».

Особливості раніше непомічених зловмисників	Порівняння для DAS
Хост (FQDN) URL-адреси на яку натиснули	≤
Відвідування хоста натисканням на URL до отримання листа	≤
Днів, коли відправлявся електронних лист	≤
Днів, коли адресат відправляв електронну пошту	≤

Таблиця А.3 - Зведення вектора ознак для нашого детектора бічних атак і «підозрілості» компаратора, який надає DAS для кожної ознаки (Розділ 1).

Особливості бічних атак	Порівняння для DAS
Хост (FQDN) URL-адреси на яку натиснули	≤
Відвідування хоста натисканням на URL до отримання листа	≤
Співробітники, які раніше входили в систему з того ж міста, що і нова сесія IP-адреси	≤
Попередні входи поточного співробітника в систему з того ж міста, що і нова сесія IP-адреси	≤

A.2 Деталі реалізації та оцінки детектора

Маркування фішингових листів

Маркування листа як фішингового або доброякісного: При ручному маркуванні листа ми починали з вивчення п'яти частин інформації: чи був лист зареєстрований як фішинговий, зміст повідомлення, підозрілий URL, який був позначений, і чи має його домен сенс у контексті одержувачі листа та відправник. За винятком кількох інцидентів, ми могли легко визначити фішинговий лист, виконавши вказані вище дії. Наприклад: лист про «загальний документ Office 365», надісланий сотням непов'язаних одержувачів, посилання на документ якого вказувало на скорочений домен bit.ly (що не належить компанії Microsoft); або лист з описом «проблеми безпеки облікового запису», надісланий співробітником, що не належить до IT-відділу, де URL-адреса «скидання облікового запису» вказувала на незв'язаний домен. У більш складних випадках ми аналізували всі відповіді та переадресації в ланцюжку електронних листів і позначали лист як фішинговий, якщо він отримував кілька відповідей/переадресацій, що виражають тривогу чи підозру, або якщо зламаний аккаунт зрештою надсилав відповідь, в якій говорилося, що він говорив, що відправляв лист фішингу. Нарешті, як описано в розділі 3, ми відвідали підозрілі URL-адреси, що не викликають побічних ефектів, з вибірки помічених фішингових листів. Всі URL-адреси, які ми відвідали, вели або на проміжну сторінку попередження (наприклад, Google SafeBrowsing) або на підроблену сторінку входу до системи. Листи, які були відзначені нашим детектором, але які виглядали доброякісними на основі вивчення всієї вищезгаданої інформації, ми консервативно позначили як помилкові спрацьовування. У багатьох випадках хибні спрацьовування були очевидними; наприклад, листи, в яких «підозрілий URL», відзначений нашим детектором, перебував у підписі відправника та посилався на його особистий сайт.

Тренувальні вправи в порівнянні з реальними фішинговими листами: Крім того, щоб відрізнити помилкове спрацьовування від атаки, ми перевірили, що наші

випадки бічного фішингу є реальними атаками, а не тренувальні вправи. По-перше, на підставі заголовків бічних фішингових листів ми перевірили, що все відправляють облікові записи, що є легітимними обліковими записами підприємства. По-друге, всі облікові записи, крім п'яти, відправили одне або декілька не пов'язаних з фішингом листів в попередньому місяці. Ці два дали нам впевненість в тому, що фішингові листи були відправлені з існуючих, легітимних акаунтів і таким чином, вони представляли собою реальні атаки; тобто тренувальні вправи зазвичай не захоплюють існуючі облікові записи, через потенційного репутаційного збитку, який це може завдати (і команди безпеки підприємств, з якими ми раніше працювали, не роблять цього). Більш того, жоден з інцидентів латерального фішингу в нашій базі даних не були тренувальними вправами, відомими компанії Barracuda, і жоден з URL-адрес латерального фішингу не використав домени відомих компаній.

Налаштування моделі і гіперпараметри

Більшість моделей машинного навчання, включаючи Random Forest, вимагають від користувача завдання різних гіперпараметрів, які регулюють процес навчання моделі. Щоб визначити оптимальний набір гіперпараметров для нашого класифікатора, ми слідували кращим практикам машинного навчання, проводячи триразовий перехресний пошук по сітці всіх комбінацій гіперпараметров, перерахованих нижче [26].

1. Кількість дерев: 50 - 500, з кроком 50 (тобто 50, 100, 150, ..., 450, 500).

2. Максимальна глибина дерева: 10 - 100, з кроком 10.

3. Мінімальний розмір аркуша: 1, 2, 4, 8

4. Співвідношення вибірки (доброякісних / атакуючих) електронних листів: 10, 50, 100, 200

Оскільки наш навчальний набір даних містив всього кілька десятків інцидентів, ми використовували три складання, щоб гарантувати, що кожна складка в крос-валідації містила кілька примірників атак. У наших експериментах використовувалася модель Random Forest з 64 деревами, максимальною глибиною 8,

мінімальним розміром листа 4 елементи, і понижувальною вибіркою 200 доброякісних листів на 1 атакуючий лист, оскільки ця конфігурація дала найвищий показник AUC [87]. Але ми відзначаємо, що багато комбінацій гіперпараметрів дали схожі результати.

А.3 Додаткові підходи до виявлення

У цьому розділі ми досліджуємо дві "неоптимальні" текстові стратегії для виявлення бічного фішингу, про які ми згадували раніше в розділі 3. Спочатку ми опишемо характеристики і процес класифікації для кожної з цих двох стратегій, а потім оцінимо їх ефективність, використовуючи тимчасове вікно і дані оцінки. Нарешті, ми оцінюємо продуктивність детектора, який поєднує обидві ці додаткові стратегії з основною стратегією і перекриття в виявленні по всім трьом підходам.

Детектор нечіткого збігу фішингу

Стратегія виявлення "нечіткого збігу фішингу" слідує природній інтуїції: якщо текст у новому електронному листі близько збігається з текстом в інших відомих фішингових листах, то новий лист, ймовірно, також є атакою фішингу. У контексті схеми фішингових атак «приманка-експлойт» [103] цей підхід виявляє фішингові листи шляхом пошуку тематичних приманок, які раніше з'являлися і продовжують повторюються в нових фішингових атаках.

Характеристики: Як вхідні дані детектор приймає набір відомих фішингових листів. Отримавши новий лист, детектор отримує ознаку подібності, нечіткий бал подібності фішингового листа, який вимірює подібність між текстом нового листа і кожним відомим фішинговим листом.

Для кожного листа детектор нормалізує текст повідомлення, видаляючи підпис та інший автогенерований текст за допомогою бібліотеки Talon [58], видалення малих літер і розділових знаків. Потім детектор лексем нормалізує нормалізований текст і перетворює їх у набір трьохграмних послідовних слів. Для обчислення подібності між двома електронними листами ми використовуємо схожість Жаккара між цими двома наборами 3-грам. Після обчислення парної подібності нового листа з усіма відомими фішинговими листами, детектор надає найбільше значення подібності як нечіткий бал фішингової подібності.

Щоб забезпечити низьку кількість помилкових спрацьовувань, ми поєднуємо

цю характеристику змісту повідомлення з додатковою ознакою, яка характеризує, чи містить лист потенційно небезпечну дію. Зокрема, для кожного листа ми отримуємо глобальну ознаку репутації URL-адреси, яка кількісно визначає найрідкіснішу URL-адресу, що міститься в листі

Класифікація: Отримавши новий лист, ми отримуємо його нечіткий бал фіш-подібності та глобальну ознаку репутації URL. Оскільки наша увага зосереджена на фішингу на основі URL-адрес, якщо лист не містить URL-адрес, ми класифікуємо його як доброякісний. В іншому випадку ми застосовуємо набір простих, консервативних порогових значень до цих двох характеристик: якщо текст листа більш ніж на 50% схожий на відомий фішинг (його нечіткий бал подібності до фішингу ≥ 50%) і містить домен, чий глобальний рейтинг не входить до топ-100 000 доменів, то ми класифікуємо лист як фішинговий; в іншому випадку детектор маркує його як доброякісне.

Детектор збігів шаблонів

У дусі нашої попередньої стратегії, наш детектор шаблонів також намагається ідентифікувати електронні листи зміст повідомлень яких має схильність до фішингу. Однак, на відміну від попередньої стратегії, для якої розпізнавання фішингового тексту був потрібний набір історичних фішингових листів, наш детектор шаблонів намагається автоматично вивести потенційний фішинговий зміст з великого корпусу переважно доброякісних листів. Оскільки фішингові листи часто намагаються маскуватися під легітимного користувача або сервіс, цей другий підхід до виявлення намагається створити набір шаблонних текстів: «популярні» тексти, які користувачі часто зустрічаються і асоціюються у користувачів з доброякісними послугами. Наприклад, кілька атак в нашій первісній вибірці бічних фішингових листів містили повідомлення, які виглядали майже ідентично законним листами від популярного сервісу Docusign.

Особливості: Ми знаходимо ці шаблонні тексти, беручи електронні листи за останній місяць, а потім зіставляючи кожен лист з кортежем, що складається з домену

відправника листа і упорядкованого за алфавітом набору зареєстрованих доменів для всіх URL, вбудованих в лист. Далі ми зберігаємо тільки ті листи, в яких домен відправника листа і кожен домен його доменної групі входять в топ-100 000 доменів; ми також видаляємо всі листи, якщо домен відправника листа належить популярному провайдеру персональної пошти (наприклад, Gmail, AOL, Comcast і т.д.). Крім того, ми перевіряємо, що цей кортеж (домен електронної пошти відправника, доменна група) зустрічається не менш ніж в 50 отриманих електронних листах для кожної організації для 10% наших організацій. Ці вимоги допомагають гарантувати, що у нас є набір «популярних» електронних листів, в якому авторитетний відправник надіслав листа, всі його URL належать популярним доменам, і безліч різних організацій отримують цей тип електронної пошти. Нарешті, щоб витягти набір «популярних текстів», ми групуємо всі електронні листи і вибираємо лист, текст якого має найбільшу 3-грамову схожість з жаккарда серед всіх листів в групі. Потім ми підсумовуємо ці подібності для кожного листа і вибираємо лист в кожній групі з найбільшим сумарним показником. Отриманий набір листів ми називаємо набором шаблонів.

Тепер, коли у нас є набір шаблонів, ми можемо наслідувати попередній стратегії. По-перше, ми обчислюємо бал подібності шаблонів, який вимірює схожість нового листа з будь-яким відомим шаблоном, шляхом вилучення 3-грам послідовних слів з нового листа, обчислення подібності по жаккард 3-грами листи з 3-грамами кожного шаблону і вибираємо найвищу оцінку подібності. Нарешті, ми здобуваємо глобальну характеристику репутації URL нового листа, використовуючи ту ж процедуру, що і раніше.

Класифікація: Щоб класифікувати новий лист як латеральний фішинг чи ні, ми застосовуємо ті ж порогові значення до двох ознак цього детектора, як і в нашому Fuzzy Phish Detector: якщо текст нового листа більш ніж на 50% схожий на будь-якої шаблон, і лист містить домен, який не входить в топ-100 тис. доменів, наш детектор позначає цього листа як атаку; в іншому випадку він розглядає лист як доброякісний.

Оцінка: Нечіткий детектор фішингу

Для оцінки нашого нечіткого детектора фішингу ми зібрали всі електронні листи з повідомлень користувачів про бічний фішинг, а потім використовували цей набір фішингових листів в якості набору відомих фішингових листів для визначення нечіткого бала подібності з фішингом нового листа (додаток А.2). Для забезпечення тимчасової точності, ми порівнювали новий лист тільки з відомими фішинговими листами, відправленими не менше ніж за 24 години до цього.

Навчання і налаштування: На всьому наборі даних для навчання (розділ 3) наш нечіткий детектор фішингу згенерував попередження для 2 бічних фішингових інцидентів (про які повідомили користувачі) і видав 0 помилкових спрацьовувань. Незважаючи на відсутність помилкових спрацьовувань, цей підхід пропустив 38 інцидентів, про які повідомили користувачі. Ці помилкові спрацьовування відбуваються з двох причин. По-перше, більшість відомих атак в нашому наборі даних використовують дуже короткі електронні повідомлення, часто складаються з декількох слів повідомлення, часто складаються з декількох слів з вбудованих в фішинговий URL (наприклад, «Новий контракт ... Переглянути тут»), за яким слідує підпис зламаного користувача. Хоча ми використовували кілька методів для видалення підписів користувачів під час аналізу подібності текстів, при нашому масштабі в десятки мільйонів електронних листів, ми зіткнулися з великою кількістю підписів, які не змогли видалити наші методи. Ця нездатність видалити підписи мала місце для кількох відомих фішингових листів, що призвело до того, що вони генерували низькі показники схожості тексту з новими фішинговими листами; для фішингових листів з короткими повідомленнями, велика частина тексту складалася з підпису зламаного користувача. По-друге, дуже мало фішингових листів дійсно використовували один і той же або схожий текст. Наприклад, серед фішингових листів з короткими повідомленнями ми спостерігали безліч ітерацій концептуально схожого тексту, в яких використовувалися різні формулювання (наприклад, «Новий контракт» проти «Аліса поділилася з вами документом X»).

Результати виявлення: При розгляді нашого тестового набору даних (розділ 5), даний підхід видав попередження для 12 інцидентів. З них 8 дійсно є бічним фішингом; інші 4 випадки - помилкові спрацьовування. Для тих же причин, що і в навчальній базі даних, цей детектор демонструє високий відсоток помилкових спрацьовувань, пропустивши 57 інцидентів, про які повідомили користувачі. Проте, не дивлячись на високий відсоток помилкових спрацьовувань цієї стратегії, ми виявили, що вона практично не дає помилкових спрацьовувань в тестовій базі даних, що складається з десятків мільйонів листів. Більш того, серед 8 виявлених нею інцидентів латерального фішингу, 4 інциденту не були повідомлені користувачем.

Оцінка: Детектор шаблонів

Навчання і налаштування: Перед витяганням ознак або класифікацією нового листа наш детектор шаблонів використовує листи за останній місяць для створення набору шаблонів. Потім, отримавши новий лист, цей підхід витягує його ознаки і класифікує його, як описано раніше в Додатку А.2.

Запустивши цей підхід на нашому наборі даних для навчання, ми виявили, що ця стратегія правильно зазначила 4 інцидентів як бічний фішинг (при цьому 2 інциденту не були зареєстровані користувачем). Вона згенерувала тільки одне додаткове попередження, яке виявилося фішинговим навчальним листом. У всіх цих випадках зловмисники ретельно імітували зміст легітимного лист Docusign, але замінивши основне посилання «загальний документ» фішинговою URL-адресою.
Вивчивши 38 пропущених інцидентів, про які повідомили користувачі, ми виявили, що детектор шаблонів просто не пристосований для виявлення більшості бічних фішингових листів: текст пропущених атак просто не схожий ні на одне з популярних реальних листів. Наприклад, для багатьох фішингові листи часто представляли собою лише коротке повідомлення, наприклад, «Будь ласка, подивіться прикріплений файл» або «Будь ласка, подивіться прикріплений файл».

Результати виявлення: В тестовій базі даних наш детектор шаблонів видав попередження тільки для 8 інцидентів, всі з яких фішингові листи, які прийшли з

зовнішніх джерел, які підробляють підроблене ім'я користувача в організації-жертві. Як і у випадку з навчальною базою даних, всі інциденти бокового фішингу, про які повідомили користувачі в нашій тестовій базі даних, містять фішингові повідомлення, які не збігаються з текстом законного популярного листа. Таким чином, виходячи з мотивів і припущень, що лежать в основі нашого детектора шаблонів, до даного підходу буде складно виявити подібні атаки. Проте, як і у випадку з нашим навчальним набором даних, ця стратегія виявлення дала 0 помилкових спрацьовувань на десятках мільйонів листів, при цьому було відзначено кілька фішингових інцидентів (хоча і спровокованих зовнішньої підробкою).

Комбінований детектор

Ми можемо об'єднати наші три стратегії виявлення (включаючи наш основний підхід, розділ 3) в один детектор, позначивши новий лист як бічний фішинг, якщо будь-який з методів вважає його фішинговим. У цьому розділі ми також називаємо кожну із стратегій «субдетектором».

Результати комбінованого виявлення: На всьому тестовому наборі даних цей комбінований детектор досяг показника відгуку 87,3%, точності 23,3% і коефіцієнта помилкових спрацьовувань 0,00036% (одне помилкове спрацьовування на 277 000 відправлених співробітниками електронних листів). Хоча його точність нижче бажаної, загальний низький рівень помилкових спрацьовувань може дозволити цьому детектору бути життєздатним в експлуатації.

Наша тестова база даних складається з 52 організацій з нашої навчальної бази даних, а також набору з 40 нових організацій; ми виявили, що наш детектор показав зіставні результати на обох базах даних. Якщо розділити показники ефективності тестової бази даних, то наш детектор досяг показника відгуку 81,8% і точності 24,8% для наших навчальних організацій в порівнянні з показником відгуку 91,0% і точності 22,8% для організацій, виключених з бази данних.

Накладення детекторів: З 97 інцидентів тестового набору даних, виявлених нашим сукупним детектором, 90 були виявлені тільки одним субдетектором, а 7 інцидентів були виявлені двома субдетекторами. З решти 90 інцидентів первинна стратегія виявлення, розглянута в Главі 5, виявила всі інциденти, крім одного, а останній інцидент був отриманий від Fuzzy Phish Detector. Як ми досліджували раніше в Додатку А.2, цей результат відображає той факт, що текст фішингових листів часто змінюється з часом, в результаті чого наші дві стратегії, засновані на схожості тексту, пропускають нові атаки, які виявляє наш основний підхід.

ДОДАТОК В - ВИЯВЛЕННЯ АТАК БІЧНОГО ПЕРЕМІЩЕННЯ

В.1 Сценарії доброякісного руху

При розробці і аналізі попереджень, що видаються Hopper за перші три місяці наших даних (з 1 січня 2019 року по 1 квітня 2019 року) ми помітили три поширені доброякісні причини для шляхів, за якими Hopper визначає можливу підміну облікових даних у причинного користувача. Ми розробили набір евристик для визначення того, чи підпадає шлях під один із цих сценаріїв, і в цьому випадку не генеруємо попередження. В цілому, ці три випадки маркують приблизно 170 000 шляхів як доброякісні (з приблизно 10 мільйонів шляхів, створених механізмом причинності Hopper).

Перший клас доброякісних шляхів відповідає однохоповим шляхам (входам), які відносяться до трьох підкатегорій: входи нового користувача, нової машини або входи, пов'язані з наданням машини. Входи нового користувача або нової машини іноді призводять до того, що Hopper робить висновок про невідповідність між причинним користувачем входу в систему і цільовим користувачем входу в систему, якщо база даних інвентаризації організації ще не перемаркувала нового співробітника як власника машини. Ми обчислюємо вік користувача і машини (тобто різницю між поточним входом і першим появою користувача в базі даних інвентаризації та журналах організації) і придушуємо будь які попередження для користувачів або машин віком менше одного тижня. Логіни, пов'язані з наданням машин, коли системний адміністратор повторно створює і налаштовує машину, щоб скинути її налаштування і перепризначити машину новому власнику, також створювали безліч одноланцюгових шляхів з перемикачем «чисті повноваження». В рамках цього процесу системний адміністратор запускає сценарій, який аутентифікує і входить на різні

спеціалізовані сервери для настройки машини (наприклад, установка операційної системи і необхідного програмного забезпечення з внутрішніх серверів, настройка нового користувача машини і дозволів на контролерах домену організації і т.д.). Оскільки в цих входах використовуються облікові дані системного адміністратора, Hopper зробить висновок, що відбулася підміна облікових даних, оскільки системний адміністратор (цільове ім'я користувача) НЕ дорівнює власнику вихідної машини (причинний користувач). Щоб визначити події входу в систему, пов'язані з перепрофілюванням машин, Hopper перевіряє наявність трьох властивостей (1) місце призначення входу належить до набору серверів візуалізації машин і ініціалізації в рамках підприємства, (2) цільовий користувач відповідає системному адміністратору, і (3) вхід відбувається з однієї з виділених підмереж, які використовуються для візуалізації машин в будь-якому з офісів Dropbox (на основі наданої доменом інформації про підмережі середовища). Якщо Hopper класифікує логін з цими трьома властивостями, він не запускає свій механізм визначення причинно-наслідкових зв'язків і не генерує попередження. В цілому, Hopper виявив 125 743 доброякісних шляхів, які відповідають одному з цих випадків входу на нову машину або (повторного) забезпечення. По-друге, використання службових облікових записів призводить до появи 42 008 одноланцюгових шляхів, які в іншому випадку Hopper позначив би як випадки явного перемикання облікових даних. У цих входах легітимний користувач дійсно виконував вхід, використовуючи набір облікових даних (цільове ім'я користувача), який не збігався з його власним; однак ці входи відповідають очікуваному переключенню облікових даних в рамках доступу до служби на підприємстві. Наприклад, один набір таких логінів включає користувачів, що виконують сценарій для запуску тестових і налагоджувальних завдань при створенні нової версії додаткиів організації; частина цього сценарію включає віддалені команди, що видаються на машини для збирання та тестування службовим обліковим записом (наприклад, користувач = «Test-services»). Hopper

визначає набір цих імен користувачів сервісів, ідентифікуючи будь-яке ім'я користувача, яке не збігається з ім'ям користувача співробітника і що більше десяти різних вихідних машин використовувалися для успішного входу в систему по набору історичних даних. Щоб переконатися, що імена користувачів, виведені Hopper, не пропонують широкомасштабного доступу або високопрівілегірованних можливостей, Hopper виводить набір виведених службових облікових записів для підтвердження службою безпеки організації і використовує тільки набір затверджених службових імен при фільтрації входів з цього доброякісного сценарію. Оскільки ці облікові записи призначені для виконання обмежених і специфічних службових операцій, організації можуть знизити ризик латерального переміщення за допомогою цих облікових даних, налаштувавши їх з обмеженим набором прав доступу до певного набору машин[47], на відміну від створення повноцінного інтерактивного сеансу.

Останній доброякісний сценарій включає в себе входи на бастіонний вузол і з нього. Організації часто сегментируют частини своєї мережі для підвищення ефективності, обслуговування і безпеки, розміщуючи набір машин за захищеним вузлом бастіону [104]. Щоб отримати доступ до сервера в цьому сегменті мережі, користувач повинен спочатку пройти тунель і аутентифікацію через бастіон сегмента мережі.

Таблиця В.1 - Розподіл успішних атак за сценаріями, змодельованих нашою системою атак.

	Розвідка	Агресивний розкид	Цілеспрямова-ний
Відсутність скритностності	41	41	40
Активна скритність	14	14	13
Пріорітетна скритність	41	41	40

Повна скритність	14	14	13

Корпоративна мережа Dropbox містить кілька таких мережевих сегментів і близько 2000 машин входи в систему на машинах за бастіонним вузлом приводили до того, що Hopper генерував причинно-наслідкові шляхи з невизначеною причинністю. Ці входи створювали плутанину в механізмі причинності Hopper, тому що вони відбувалися близько за часом з входом іншого користувача в машину в сегменті мережі; оскільки обидва входи повинні проходити через бастіонний вузол сегмента і відбуватися близько за часом, Hopper не може сказати, який вхід в бастіон викликав якийсь вихідний вхід з бастіону в кінцевий пункт призначення шляху. Оскільки машини бастіонів відповідають захищеним вузлам, виконують обмежений набір операцій (аутентифікація і пересилання з'єднань) і часто не дозволяють користувачам встановлювати логіни на самому вузлі, ці шляхи не пропонують можливості для зміни облікових даних. Таким чином, враховуючи список вузлів бастіону в організації, Hopper не попереджає жоден з цих одноходових або двохопових шляхів.

B.2 Синтез реалістичних атак

Щоб доповнити нашу оцінку додатковими реалістичними атаками, ми розробили структуру для синтезу логінів бічного переміщення, які відповідають одному з дванадцяти сценаріїв атаки. Кожен сценарій складається з пари параметрів, які визначають одну з трьох цілей атакуючого і один з чотирьох рівнів скритності (описані нижче).

Процедура синтезу атаки: З огляду на конкретний сценарій атаки і початкову «опорну» машину, яку зловмисник вже скомпрометував, наша система вибирає випадкове час початку атаки (протягом періоду часу, коли «опорна» машина залишається активною в наших даних). Далі наша система генерує набір логінів атаки, поки не досягне умови завершення, заданого метою сценарію атаки.

Під час кожної ітерації наша система визначає набір всіх можливих входів «в наступний стрибок»; цей набір відповідає всім комбінаціям внутрішніх машин, які атакуючий зламав (потенційні вихідні машини), всім обліковим даним, які атакуючий зламав (потенційні імена цільових користувачів), і всім машинам (потенційні місця призначення), до яких зламані облікові дані атакуючого успішно отримали доступ при будь-якому вході в систему в усьому нашому наборі даних. Моделюючи потужного противника, ми припускаємо, що атакуючий має можливість визначити, до яких пунктів призначення може отримати доступ з скомпрометованих облікових даних (наприклад, за допомогою операцій виявлення поза зоною дії або внутрішньої розвідки, таких як перерахування Active Directory[103]). На початку атаки набір скомпрометованих машин і повноважень відповідає тільки машині атакуючого і обліковим даним власника машини.

Потім наша система відсіває цей список потенційних наступних хопів, грунтуючись на скритності і цілі сценарію атаки. Після обрізки цього списку наша система випадковим чином вибирає один з цих логінів в якості наступного

атакуючого логіна і встановлює час цього логіна, додаючи випадковим чином зсув між 0-12 годинами після попереднього атакуючого логіна. Наша система позначає місце призначення нового входу як скомпрометований, а також облікові дані будь-якого користувача, який недавно увійшов в це місце (тобто імітує атакуючого, який «скомпрометував» паролі / облікові дані всіх імен користувачів, які увійшли в нову машину призначення в протягом останніх семи днів). Атака завершується (припиняється генерація додаткових входів), якщо цей новий вхід відповідає меті сценарію (описаної нижче), або якщо атака відвідала всі можливі місця призначення,

Сценарії атак: Дванадцять сценаріїв латерального переміщення в нашому фреймворку виникають в результаті сполучення цілі атаки з рівнем «скритності» атаки. Мета атаки визначає, коли атака завершується успішно, тобто коли наша система синтезу припиняє генерувати логіни атаки. Рівень скритності атаки визначає фазу обрізки нашої системи генерації атак, яка відсіває набір потенційних наступних переходів тільки до тих, які відповідають певному рівню скритності.

1.Мета атаки

а) Агресивне поширення: ця мета моделює атакуючого, який намагається скомпрометувати якомога більше машин на підприємстві (наприклад, атака ransomware). При синтезі цих атак наша система генерує логіни в режимі обходу по широті, перебираючи всі скомпрометовані облікові дані, отримані атакою, і переходячи до всіх пунктів призначення, доступні за кожним обліковим даними; якщо наша система вже отримала доступ до машини призначення на попередньому етапі атаки, з будь-яким набором облікових даних, вона не генерує додатковий вхід до системи. Атака завершується, як тільки наша система або згенерувала вхід до всіх можливих пунктів призначення, до яких скомпрометований набір облікових даних може отримати доступ (відповідно до заданого сценарію скритності), або коли вона перевищить 50 логінів атаки.

б) Цілеспрямована компрометація: наша система імітує цілеспрямовану атаку, синтезуючи переходи атаки до тих пір, поки не буде отримано вхід до однієї з приблизно двадцяти п'яти "високоцінних машин", які ми вибрали; ці машини відображають репрезентативний діапазон машин на які спрямовані багато реальних атак, такі як сервери, що управляють обліковими даними користувачів та дозволами (наприклад, контролери домену Windows) та критична інфраструктура (наприклад, DNS-сервери організації).

За такої мети атаки наша система попередньо обчислює набір найкоротших шляхів від початкового плацдарму до будь-якої машини з обліковими даними, що дозволяють отримати доступ до однієї з цих високоцінних машин (наприклад, машин, до яких увійшли системні адміністратори). Потім наша система обчислює найкоротший шлях від будь-якої зі скомпрометованих машин атаки до будь-якої з високоцінних машин. У міру того, як наша система синтезує логіни для атак, вона підбирає набір кандидатів на переходи, щоб переконатися, що атака йде найкоротшим шляхом до цих двох цілей.

с) Оппортуністична розвідка: в цій атаці наша система синтезує логіни атак доти, доки не буде отримано логін, що дає доступ до нового місця призначення, до якого початкова жертва не має доступу. На кожному кроці ця атака перераховує набір життєздатних переміщень по кордону. Потім наша система обрізає цей набір, щоб включити в нього тільки ті ребра, які звертаються до нової машини призначення (ще не відвіданої під час атаки), і випадковим чином вибирає одне з ребер, що залишилися для наступного входу в систему на шляху атаки. Цей сценарій допомагає оцінити Hopper проти атак, які можуть скомпрометувати цікаву чи чутливу машину, яку ми не вказали у наборі високоцінних машин для нашої мети «Цільової компрометація».

2. Прихована атака

а) Відсутність скритності: обрізання не провадиться. Для вибору наступного входу в атаку наша система випадково обирає один з кандидатів на наступний

перехід.

b) Активне використання облікових даних: обрізання набору кандидатів на наступний перехід тільки для тих логінів, які використовують облікові дані початкової жертви або облікові дані початкової жертви або переходи, які використовують новий набір облікових даних, якщо легітимний користувач нещодавно (протягом останніх 24 годин) увійшов у вихідну машину стрибка; тобто, зловмисник виконує вхід з новим набором облікових даних тільки в тому випадку, якщо він створює шляхи з неясним причинно-наслідковим зв'язком. Цей сценарій імітує потайливого зловмисника, який використовує облікові дані тільки тих машин, де нещодавно був активний легітимний користувач, що дозволяє йому потенційно вислизнути від виявлення, оскільки справжній користувач, ймовірно, міг увійти до системи.

c) Рух по відомими ребрами: обрізає набір кандидатів на наступні переходи лише до тих, що перетинають ребра графа з попередньою історією. Така скритність імітує зловмисника, який намагається уникнути виявлення, переміщаючись лише між машинами з попередніми доброякісними входами в систему. (Тобто створює тільки неаномальні ребра атаки).

d) Комбінована скритність: застосовує критерії "Активне використання облікових даних" та "Відомий кордон" критерії скритності для відсіювання набору логінів, які наша система вибирає як наступний стрибок.

Хоча ця схема не є абсолютно точною, вона забезпечує хорошу апроксимацію потужного та реалістичного супротивника: коли зловмисник переміщається в нове місце призначення, наша схема передбачає, що зловмисник може використовувати вразливість локальних привілегій для отримання повних адміністративних привілегій на машині і, таким чином, скомпрометувати або перехопити облікові дані будь-якого користувача, який нещодавно увійшов до системи. Оскільки кешовані облікові дані мають різний час життя залежно від протоколу входу та локальних конфігурацій цільової машини, наше семиденне

вікно дає консервативну оцінку того, як облікові дані інших користувачів залишаються вразливими на машині. Більше того, хоча в реальному світі зловмисник може не знати всіх машин, до яких може отримати доступ до облікових даних довільного користувача, ми припускаємо, що зловмисник може визначити всі машини, до яких може успішно отримати доступ до облікових даних, при генерації набору потенційних місць призначення наступного стрибка. В ідеалі наша система повинна точно знати, до яких машини може отримати доступ цей користувач, але в нашому дослідженні немає легкого доступу до цих даних, тому ми наблизили цей набір, обчисливши всі машини, до яких ім'я користувача коли-небудь мало доступ у нашому логіні в нашій базі даних.

Синтез атак: Ми випадково вибрали набір з 50 користувачів з усіх співробітників, які не є системними адміністраторами, які мали хоча б один внутрішній вхід у систему в нашому остаточному, постфільтрованому наборі даних. (Розділ 4). Для кожного користувача з цього набору 50 "початкових жертв" ми запустили нашу систему, щоб синтезувати синтетичні атаки зловмисника, що компрометує ноутбук випадкового співробітника, де кожен із дванадцяти сценаріїв атаки на початкову жертву імітує різні типи скритності та мети зловмисника. Якщо бічний рух атаки не вдавався (тобто не вдавалося отримати і використовувати новий набір облікових даних), ми запускали нашу систему повторно кожного дня нашого вікна оцінки (коли плацдарм атаки та початкова жертва все ще існували в Dropbox). Якщо в якийсь із цих днів наша система проводила успішну атаку, ми заміняли невдале латеральне переміщення випадковим набором успішних логінів.

Загалом ця процедура дозволила створити 326 синтетичних атак, які успішно провели бічне переміщення. Для 9 з початкових жертв атакуючий не мав можливості переміщатися вбік, тому що користувач мав доступ тільки до невеликого набору серверів, до яких мали доступ тільки він або члени команди з ідентичним дозволом доступу; Таким чином, у атакуючого, що починає з машин

цих початкових жертв, не було шляху до інших машин. Крім того, для 36 початкових жертв зловмисники не мали прихованого шляху, який дозволив би їм здійснити бічне переміщення. Навіть якщо одна з цих початкових жертв могла зустріти облікові дані іншого користувача, які б дозволили отримати доступ до нової машини, інший співробітник ніколи не використовував свої облікові дані для входу на нову машину із взаємодоступного сервера(ів). У таких ситуаціях зловмисник не міг використовувати ці нові облікові дані без генерації ніколи раніше не баченої події входу в систему; тому наша система синтезу не може генерувати прихованих атак. імітації атак у цих ситуаціях.

Загалом цей процес згенерував 326 успішних атак бічного переміщення, де кожна атака успішно досягла своєї мети при заданому рівні скритності. Для сценаріїв, що належать до двох останніх рівнів скритності, ми виявили, що багато користувачів просто не мали жодного шляху до нових облікових даних/машин, які також переміщалися між машинами виключно за допомогою попередніх логінів. Наприклад, у сценаріях цільової компрометації жодна з початкових жертв не мала шляху, який би вів до набору підвищених облікових даних адміністратора, а також проходив прихованим (раніше протореним) шляхом входу в систему на важливу машину; Типові шляхи до цих критичних серверів часто включають машини, на яких працюють лише кілька користувачів (зазвичай системних адміністраторів) які мають законний доступ.

B.3 Додаткові відомості про сповіщення

З 2399 попереджень, зазначених як явні шляхи перемикання облікових даних (Розділ 4), 1326 попереджень відображають шляхи переміщення, де перемикання облікових даних відбулося при вході з клієнта на сервер. Ці попередження загалом поділяються на дві категорії помилкових спрацьовувань. По-перше, приблизно 10% цих входів відповідають клієнтським машинам, які виконують вхід на сервер конкретної команди з обліковим записом служби, що рідко використовується. Ці службові імена користувачів не збігаються з ім'ям користувача власника клієнтської машини, що спричиняє попередження від детектора явного перемикання облікових даних Hopper; через рідкісне використання цих службових облікових записів процедура Hopper для виведення та обрізання облікових даних службових облікових записів не фільтрує ці попередження. Другий клас хибних спрацьовувань (приблизно 70% хибних попереджень про перехід із клієнта на сервер) виникає через те, що системні адміністратори в невеликих офісах виконують повторну візуалізацію чи повторне надання існуючої клієнтської машини. Цей процес повторного надання машини запускає сценарій, який створює безліч входів на сервери керування та аутентифікації для правильного налаштування машини (наприклад, завантаження та встановлення різного програмного забезпечення з внутрішніх репозиторіїв, налаштування політик та дозволів машини на всіх контролерах домену організації тощо) . Кожен із цих входів видає попередження про несподіване використання облікових даних системного адміністратора на клієнтському пристрої, що належить іншому користувачеві (тобто шлях, де, на думку Hopper, старий власник машини раптово почав створювати шляхи входу під обліковими даними системного адміністратора). Хоча наша обрізка за сприятливим сценарієм (Розділ 4 та Додаток B.1) видаляє багато явних входів з заміною облікових даних, які відбуваються в результаті (пере)надання машин у великих офісах Dropbox, деякі з невеликих офісів не мають виділених підмереж для надання машин. Отже, логіни, зроблені в рамках

цих операцій, не підпадають під фільтрацію Hopper та не викликають попереджень.

Таблиця В.2 - Порівняння базового рівня: кількість атак, які SAL [71] виявляє за різних комбінацій параметрів.

	1	2	3	4	5	6	10	15	16	18
0.01	386	596	717	764	808	834	965	1,05	1,0	1,1
0.02	722	1,1	1,36	1,5	1,65	1,77	2,21	2,65	2,7	2,8
0.1	1,809	2,72	3,41	3,97	4,41	4,79	6,09	7,52	7,7	8,2
0.2	2,34	3,55	4,50	5,3	5,94	6,5	8,24	10,17	10	11
0.5	8,02	11,3	13,9	15,8	17,4	18,7	23,1	27,92	28	30
0.75	11,97	16,8	20,5	23,1	25,4	27,5	33,9	40,40	41	43

B.4 Деталі базової оцінки

SAL складається із двох етапів. По-перше, SAL бере набір історичних входів у систему та використовує ці "навчальні дані" для побудови графа входів у систему, аналогічного графу Hopper. Враховуючи партію нових входів, SAL генерує набір попереджень-кандидатів (логінів), визначаючи всі логіни, які перетинають ребро графа, що мало місце за N > днів у навчальних даних, де N - поріг, що задається користувачем, який визначає мінімальну кількість (або відсоток) днів для "доброякісного" ребра. По-друге, SAL обрізає ці кандидати у попередження до остаточного списку попереджень, видаляючи будь-який вхід до системи, що відповідає "доброякісному шаблону входу". SAL використовує свої історичні навчальні дані вивчення набору доброякісних шаблонів шляхом зіставлення кожного входу до системи зі списком "шаблонів входу до системи", де кожен "шаблон входу" складається з триплету (атрибути вихідної машини, атрибути цільової машини, атрибути цільового користувача). Наприклад, враховуючи вхід у систему машина A, dest = машина B, користувач = Аліса), (джерело = Нью-Йорк, пункт призначення = Сан-Франциско, користувач = Sales) буде одним шаблоном входу, якщо машина A знаходиться у Нью-Йорку, машина B – у Сан-Франциско, а Аліса працює у відділі продажів. Потім SAL агрегує всі шаблони входу в систему за всіма своїми навчальними і формує набір "доброякісних шаблонів", визначаючи будь-який шаблон входу, в якому досить велика частка вихідних машин, цільових машин та/або користувачів має хоча б один історичний, наприклад, при заданому користувачем порозі в 33%, SAL додасть шаблон входу до системи у свій набір доброякісних шаблонів, якщо принаймні 33% машин підприємства мають логін, що відповідає шаблону(як вихідна або кінцева машина), або якщо більше 33% користувачів мають хоча б один логін, що відповідає шаблону.

Таблиця В.3 - Порівняння базового рівня: кількість попереджень, згенерованих SAL [71] на оціночних даних при різних комбінаціях параметрів.

	1	2	3	4	5	6	10	15	16	18	20
0,01	30	33	34	34	34	34	34	34	34	34	34
0,025	55	72	73	73	74	76	82	83	83	83	83
0,1	98	126	130	139	150	155	174	186	186	188	191
0.2	117	156	165	175	190	201	225	244	249	251	254
0,5	188	231	245	251	260	274	302	309	313	315	316
0,75	222	256	267	271	279	289	315	320	323	326	326

На основі доступних нам даних ми використовуємо наступний набір атрибутів зі статті SAL: кожен користувач має два атрибути: (команда користувача та тип користувача: системний адміністратор, звичайний користувач або службовий обліковий запис), і кожна машина має два атрибути: (тип машини: клієнтська або географічне розташування машини).

Оскільки для SAL потрібні входи, що надаються двома користувачами, N - мінімальна кількість днів для доброякісного краю входу і P мінімальна частка машин або користувачів для доброякісного шаблону, ми здійснили пошук по сітці в діапазоні параметрів та вибрали найкращі параметри для нашої оцінки: тобто, комбінацію параметрів, яка виявляла всі атаки з найменшою кількістю помилкових спрацьовувань. У таблиці В.3 та Таблиця В.2 показують загальний обсяг попереджень та кількість атак, виявлених при різних комбінаціях цих параметрів. комбінацій параметрів.

ДОДАТОК С - ПРЕЗЕНТАЦІЯ РОБОТИ

- **Мета:**

  створення нового методу виявлення вдосконалених атак для перешкоджання викрадення облікових даних.

- **Завдання:**

  1. Проаналізувати методи виявлення вдосконалених атак на корпоративні мережі і їх проблеми з перевагами та недоліками, щоб визначити найпоширеніший метод виявлення.

  2. Розглянути найпопулярніші методи атак і способи зменшення помилкових спрацювань.

  3. Запропонувати метод захисту для пом'якшення та виявлення атак в корпоративній мережі.

  4. Розробити ментальні моделі, які покращать розуміння і здатність виявляти фішингові атаки.

  5. Розробити новий набір емпіричних результатів і алгоритмів виявлення.

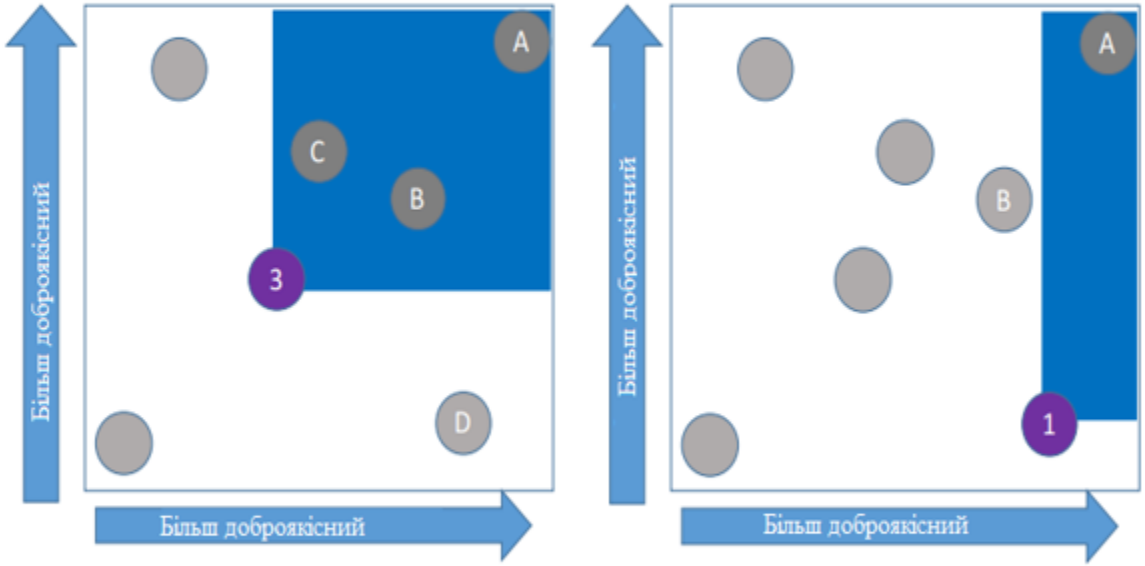
  6. Протестувати розроблену систему виявлення.

- **Об'єктом та предметом дослідження в дипломній роботі є:**
  система та методи виявлення основані на даних для виявлення ускладнених атак.

- **Актуальність:**

  протягом останнього десятиліття безліч організацій по усьому світі стали жертвами великої кількості атак, що призводять не тільки до великих фінансових трат, але й завдають фізичної шкоди важливим інфраструктурам, які вони виводять з ладу під час атаки.

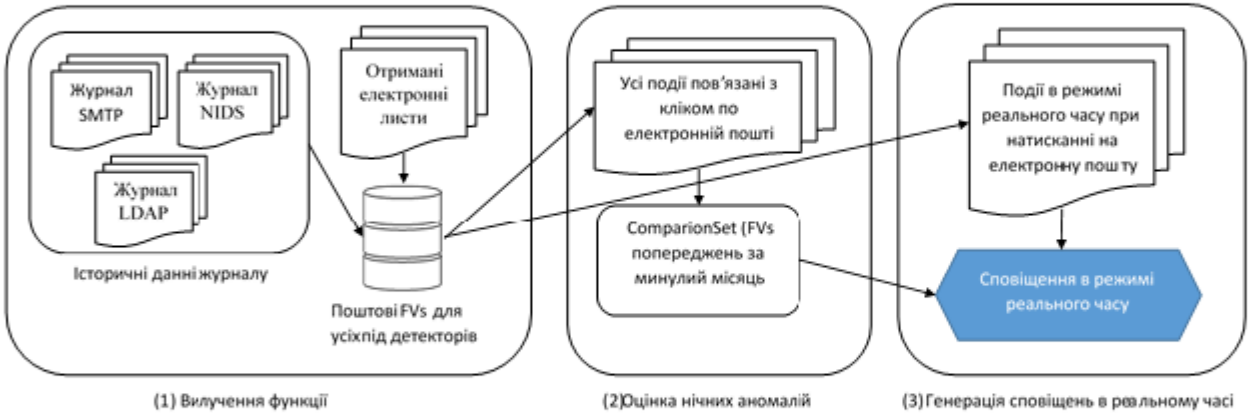
Приклад оцінки DAS для подій в двовимірному просторі ознак



Порівняння класичних методів виявлення аномалій

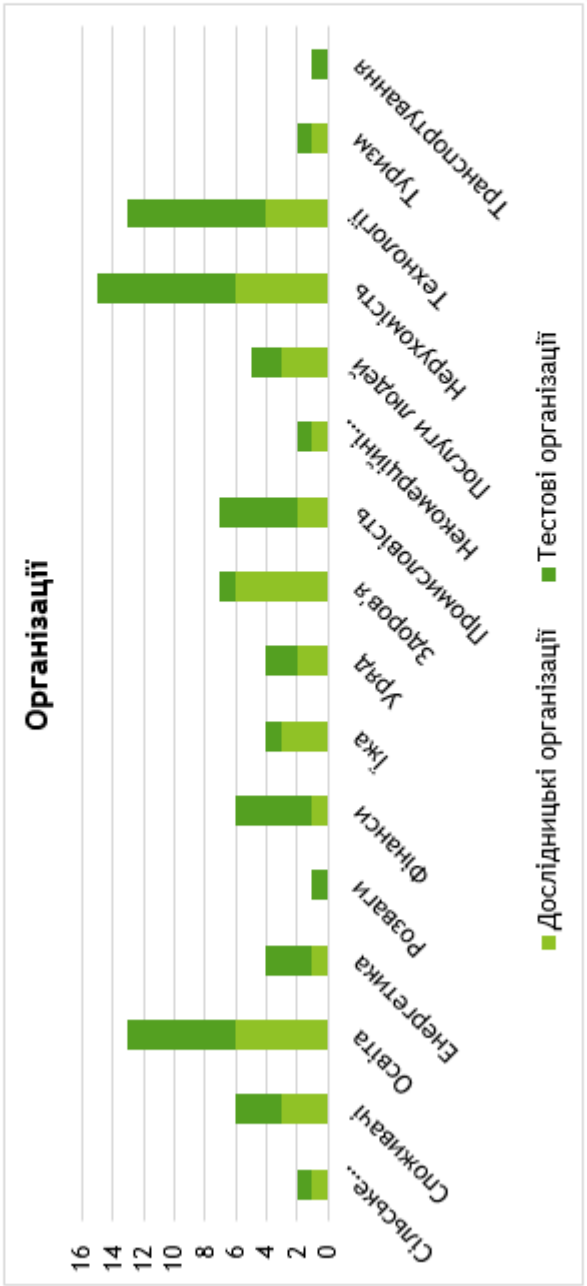
Алгоритм	Виявлено	Щоденний б'юджет
kNN	3/19	10
	17/19	2,455
GMM	4/19	10
	17/19	147
KDE	4/19	10
	17/19	91
DAS	17/19	10

Огляд детектора реального часу



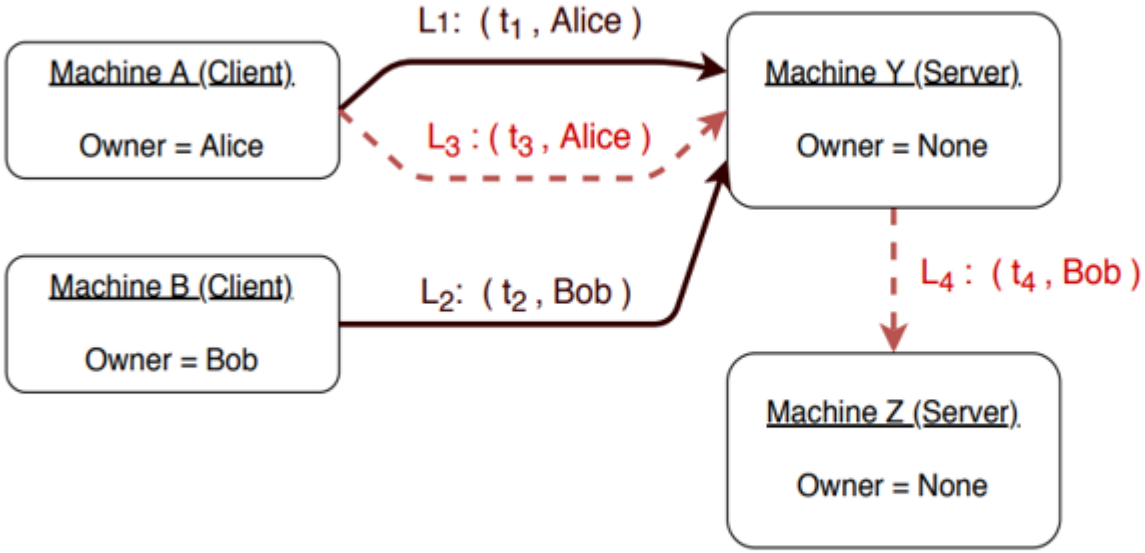
Розподіл обсягів організації

Розбивка економічних секторів



Частка активних інцидентів протягом 30 днів

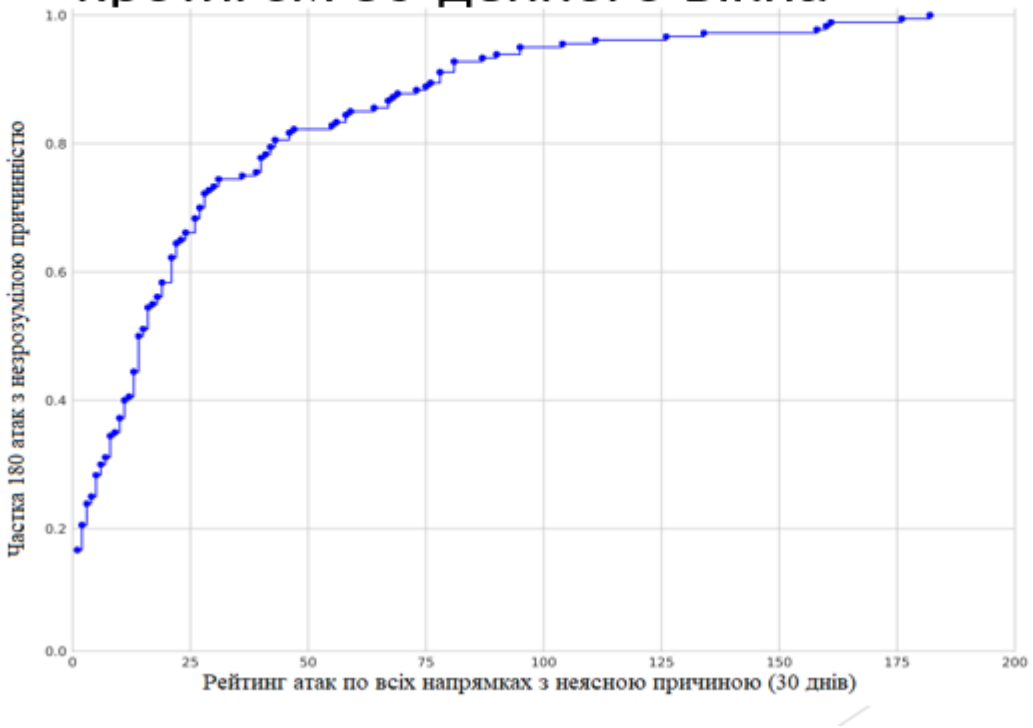
Приклад простого графа входу в систему



Інформація, що міститься в передбачуваному причинно-наслідковому шляху, створеному механізмом причинності Hopper

Компонент шляху	Опис
Фокальний стрибок	Остаточний стрибок шляху, або шлях перемикання облікових даних
Список стрибків	Список входів на шляху
Причинний користувач	Ім'я користувача, працівника, чия машина ініціювала шлях
Імовірність шляху	Частка причинних шляхів, щоб Hopper зробив висновок про фокальний стрибок

Рейтинг шляхів атак з неясною причинно-наслідковим зв'язком, щодо всіх попереджень, згенерованих Hopper протягом 30-денного вікна



Порівняння ефективності детекторів Hopper та SAL

Детектор	Швидкість виявлення	Мін. загальна кількість сповіщень
SAL	156/327	3556
	312/327	28771
Hopper	312/327	3544

Висновки

- Оскільки організації регулярно стаються жертвами складних атак, це покладає багатообіцяючий шлях вперед через новий набір заснований на даних і методах, які можуть пом’якшити ці руйнівні загрози.
- Для подальшого розвитку даний метод має великі перспективи, оскільки представлені моделі допоможуть покращити розуміння та здатність виявлення фішингових атак.

UNICHECK

Ім'я користувача:
Кафедра кібербезпеки

ID перевірки:
1009454504

Дата перевірки:
01.12.2021 13:57:31 EET

Тип перевірки:
Doc vs Internet + Library

Дата звіту:
01.12.2021 13:59:58 EET

ID користувача:
100008300

Назва документа: **диплом_Вишневий**

Кількість сторінок: **92** Кількість слів: **20005** Кількість символів: **144511** Розмір файлу: **2.13 MB** ID файлу: **1009469030**

2.13%
Схожість

Найбільша схожість: **1.19%** з джерелом з Бібліотеки (ID файлу: **1005681005**)

1.07% Джерела з Інтернету	73	Сторінка 94
1.34% Джерела з Бібліотеки	59	Сторінка 94

0% Цитат

Вилучення цитат вимкнене

Вилучення списку бібліографічних посилань вимкнене

0%
Вилучень

Немає вилучених джерел

Модифікації

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

Замінені символи	56

Wed Dec 01 10:00:09 EET 2021, Мостовий Сергій Володимирович, Хмельницький національний університет, ХНУ

Anti-Plagiarism v-15.257

Максимальное совпадение с одним документом 1.0%

Словари проверки: en_US, ru_RU, ua_UA. **Ошибок в документах: 10%**

ID: 97701 Название: Метод виявлення вдосконалених атак на корпоративні мережі Добавлено в БД: 2021-12-01 Авторы: Вишневий В.В. Руководители: Тітова В.Ю. Консультанты: Оппоненты:	Документ		Суммарное совпадение по Базе Данных	
	Символы	Лексемы	Символы	Лексемы
	141783	876	1252 (1%)	19 (2%)

Источник плагиата

ID	Описание	Наличие плагиата в документе	
		Символы	Лексемы

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

РЕЦЕНЗІЯ НА ДИПЛОМНУ РОБОТУ

освітньо-кваліфікаційного рівня «магістр»

Магістр Вишневий Віктор Володимирович

Тема Метод виявлення вдосконалених атак на корпоративні мережі

Спеціальність 123 – Комп'ютерна інженерія

Обсяг дипломної роботи освітньо-кваліфікаційного рівня «магістр»:

кількість листів креслень 11 ; кількість сторінок записки 92

1. Короткий зміст ДР та прийнятих рішень Дана кваліфікаційна робота присвячена для удосконалення методу виявлення вдосконалених атак на основі використання підходу основаного на даних

2. Висновок про відповідність ДР дипломному завданню Дипломна робота у повній мірі відповідає поставленому завданню як в теоретичній, так і в практичній частині дипломної роботи

3. Характеристика виконання кожного розділу роботи, ступінь використання останніх досягнень науки і техніки і передових методів роботи: У вступі подана загальна характеристика поставленої задачі, чітко визначено об'єкт, предмет та методи дослідження, сформульована актуальність. Визначені задачі, які необхідно вирішити для досягнення поставленої мети, практична цінність отриманих результатів, їхня новизна та наведені відомості про публікації. У першому розділі проведено огляд атак на периметрі підприємства, виконане обгрунтування актуальності теми дослідження і виконана постановка задачі. В другому розділі виконана розробка математичної моделі для реалізації методу, яка базується на методах машинного навчання а інтелектуального аналізу даних. В третьому розділі визначено основні положення методу та розроблено алгоритми його реалізації. Четвертий розділ присвячено апробації методу та алгоритмів його реалізації моделюванням.

4. Позитивні сторони проекту Дипломна робота має комплексну наукову і практичну цінність. Наукова цінність полягає у вдосконаленні методу виявлення вторгнень у корпоративну мережу, представленні нового підходу до виявлення рідкісних, немаркованих атак у великих, складних наборах даних. Практична цінність результатів дослідження полягає у обгрунтуванні можливості підвищення ефективності розробленої системи для запобігання вдосконалених атак на підприємствах.

5. Негативні сторони проекту немає.

6. Оцінка графічного оформлення та пояснювальної записки роботи Графічне оформлення виконане відповідно до теми дипломної роботи з дотриманням стандартів. В загальному графічне оформлення виконане якісно, пояснювальна записка відповідає нормам щодо її оформлення.

7. Відгук про роботу в цілому В загальному дипломна робота заслуговує позитивної оцінки. Весь матеріал дипломної роботи структурований, чіткий та послідовний. Усі розділи роботи послідовні та логічні, що дозволяє чітко розуміти викладений матеріал в рамках тематики дипломної роботи. Графічний матеріал дозволяє наочно побачити доцільність та ефективність рішень, які були прийняті за основу для досягнення поставленої мети.

8. Інші зауваження немає

9. Оцінка дипломної роботи Враховуючи всі позитивні та негативні сторони представленої дипломної роботи, можна зробити висновок, що вона заслуговує оцінку «відмінно».

РЕЦЕНЗЕНТ (прізвище, ім'я, по батькові, посада, місце роботи)
Гуриан Іван Васильович
к.т.н. доц. каф ІТС

« 3 » 12 2021. (підпис)

ХНУ Зам. 54, тир. 6000. 2007.

РІШЕННЯ ЕКСПЕРТНОЇ КОМІСІЇ

КАФЕДРИ КІБЕРБЕЗПЕКИ

ПРО ДОПУСК КВАЛІФІКАЦІЙНОЇ РОБОТИ ДО ЗАХИСТУ

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

Назва: Метод виявлення вдосконалених атак на корпоративні мережі

Автор: Вишневий В.В.

Спеціальність: 123 – комп'ютерна інженерія

Освітня програма: освітньо-професійна

Науковий керівник: Тітова В. Ю., к.т.н доцент

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.	+
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 законними і не є плагіатом, оскільки:

1) запозичення розміщені в розділах, які не описують безпосередньо авторське дослідження і не стосуються результатів роботи;
2) усі запозичення фрагментарні, або мають належним чином оформленні посилання;
3) більшість джерел запозичення дублюють одне одного;

Дата

Підпис (Тітова В.Ю.)

Підпис (Клюєв О.А.)