Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя
(повне найменування вищого навчального закладу)
Факультет комп'ютерно-інформаційних систем і програмної інженерії
(назва факультету)
Кафедра кібербезпеки
(повна назва кафедри)

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття освітнього ступеня

магістр
(освітній рівень)

на тему: "Аналіз, розробка та впровадження заходів захисту від UDP Flood атак на DNS "

Виконав: студент (ка)
Спеціальності:
125 «Кібербезпека»
(шифр і назва напряму підготовки, спеціальності)
Іваночко Назар Андрійович
підпис (прізвище та ініціали)

Керівник Лечаченко Т. А.
підпис (прізвище та ініціали)
Нормоконтроль Лечаченко Т. А.
підпис (прізвище та ініціали)

Завідувач кафедри Загородна Н.В.
підпис (прізвище та ініціали)

Рецензент
підпис (прізвище та ініціали)

м. Тернопіль – 2023

Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя

Факультет комп'ютерно-інформаційних систем і програмної інженерії
(повна назва факультету)

Кафедра кібербезпеки
(повна назва кафедри)

ЗАТВЕРДЖУЮ
Завідувач кафедри
_____ Загородна Н.В.
(підпис) (прізвище та ініціали)
«__» _____ 2023 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

на здобуття освітнього ступеня Магістр
(назва освітнього ступеня)

за спеціальністю 125 Кібербезпека
(шифр і назва спеціальності)

Студенту Іваночку Назару Андрійовичу
(прізвище, ім'я, по батькові)

1. Тема роботи Аналіз, розробка та впровадження заходів захисту від UDP Flood атак на DNS

Керівник роботи Лечаченко Тарас Анатолійович, доктор філософії., старший викладач кафедри КБ
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

Затверджені наказом ректора від «16» 11 2023 року № 4/7-1061

2. Термін подання студентом завершеної роботи 12.12.2023

3. Вихідні дані до роботи Вимоги до операційної системи OpenBSD, вимоги до системи захисту від UDP Flood атак.

4. Зміст роботи (перелік питань, які потрібно розробити)
Проаналізувати flood атаки на DNS сервери та їх наслідки.
Проаналізувати захист від UDP Flood атак DNS server
Розробити та протестувати тестування програмний модуль від UDP flood атак
Охорона праці та безпека в надзвичайних ситуаціях
Висновки

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень, слайдів)
Тема, мета, задачі. Наукова новизна на практичне значення роботи. Поняття UDP flood атаки Методи захисту від UDP flood атак. UDP flood атаки на DNS. Схема мережі для здійснення UDP flood атаки. Маршрутизатор на базі операційної системи OpenBSD. PF - брандмауер. Операційна система Parrot Security як засіб атаки. DNS сервер на базі операційної системи RedHat Linux. Здійснення DNS flood атаки на DNS –сервер. Здійснення DNS flood атаки на DNS –сервер. Розробка програмного модуля захисту від udp flood атак на DNS –сервер. Налаштування брандмауера PF. Написання програмного модуля. Автоматизація процесу відслідковування атаки та блокування IP зловмисника. Тестування програмного модуля. Висновки.

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Охорона праці	Осухівська Г.М., к.т.н., доцент		
Безпека в надзвичайних ситуаціях	Клепчик В.М., проректор з адміністративно-господарської роботи та будівництва		

7. Дата видачі завдання 20.09.2023 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1.	Ознайомлення з завданням до кваліфікаційної роботи	20.09 – 22.09	Виконано
2.	Підбір джерел для аналізу UDP flood атаки та їх наслідки	23.09 – 03.10	Виконано
3.	Опрацювання джерел в галузі дослідження	04.10 – 12.10	Виконано
4.	Провести аналіз методів захисту від UDP flood атаки	12.10 – 20.10	Виконано
5.	Здійснення UDP flood атаки на DNS -сервер	21.10-25.10	Виконано
6.	Налаштування системи автоматичного блокування та сповіщення UDP flood атаки	26.10 – 30.10	Виконано
7.	Оформлення розділу «Огляд UDP flood атак на сервіси та їх наслідків»	01.11 – 05.11	Виконано
8.	Оформлення розділу «Налаштування лабораторного комплексу для моделювання UDP Flood атаки на DNS»	06.11 – 11.11	Виконано
9.	Оформлення розділу «Розробка програмного модуля захисту від UDP Flood атак на DNS сервер»	12.11-20.11	
10.	Виконання завдання до підрозділу «Охорона праці та безпека в надзвичайних ситуаціях»	21.11 – 25.11	Виконано
11.	Оформлення кваліфікаційної роботи	26.11 – 07.12	Виконано
12.	Нормоконтроль	08.12 – 10.12	Виконано
13.	Перевірка на плагіат	12.12 – 14.12	Виконано
14.	Попередній захист кваліфікаційної роботи	20.12 – 15.12	Виконано
15.	Захист кваліфікаційної роботи	28.12.2023	

Студент _____ Іваночко Н.А.
(підпис) (прізвище та ініціали)

Керівник роботи _____ Лечаченко Т. А.
(підпис) (прізвище та ініціали)

АНОТАЦІЯ

Аналіз, розробка та впровадження заходів захисту від UDP Flood атак на DNS // ОР «Магістр» // Іваночко Назар Андрійович // Тернопільський національний технічний університет імені Івана Пулюя, факультет комп'ютерно-інформаційних систем і програмної інженерії, кафедра кібербезпеки, група СБс-61 // Тернопіль, 2023 // С. 69 , рис. – 27, табл. – - , кресл. – 25 , додат. – 2.

КЛЮЧОВІ СЛОВА: OPENBSD, DNS, PF, UDP, FLOOD, PYTHON.

Кваліфікаційна робота присвячена дослідженню впливу UDP flood атак на DNS та розробці методів захисту. Даний тип атак може призвести до недоступності важливих мережевих служб та фінансових збитків. Забезпечення стабільності та безпеки DNS серверів є критичною для нормального функціонування мережі.

Розроблено автоматизовану систему для виявлення і блокування атакуючих IP-адрес за допомогою операційної системи OpenBSD, системи фільтрації пакетів PF та скрипту на мові програмування Python. Також налаштовано автоматичне повідомлення адміністратору про інцидент.

Робота спрямована на розв'язання практичної проблеми захисту мережевої інфраструктури від потенційно небезпечних атак.

Отримані результати мають велике значення для адміністраторів мережевої інфраструктури, оскільки розроблена система допомагає підвищити безпеку та стійкість DNS серверів, запобігаючи атакам та автоматично повідомляючи адміністратора про події на сервері.

ABSTRACT

Analysis, Development, and Implementation of Countermeasures to Protect Against UDP Flood Attacks on DNS // Thesis of educational level "Master"// Ivanochko Nazar Andriiovych// Ternopil Ivan Puluj National Technical University, Faculty of Computer Information Systems and Software Engineering, Department of Cybersecurity, group СБс-61 // Ternopil, 2023 // P. 69 fig. - 27, tab. - _-_, chair. - 25 , added. – 2.

Keywords: OPENBSD, DNS, PF, UDP, FLOOD, PYTHON.

The qualifying work is devoted to the study of the impact of UDP flood attacks on DNS and the development of protection methods. This type of attack can result in the unavailability of critical network services and financial loss. Ensuring the stability and security of DNS servers is critical to the smooth functioning of the network.

An automated system has been developed to detect and block attacking IP addresses using the OpenBSD operating system, the PF packet filtering system and a script in the Python programming language. An automatic notification to the administrator about the incident is also configured.

The work is aimed at solving the practical problem of protecting network infrastructure from potentially dangerous attacks.

The results obtained are of great importance for network infrastructure administrators, since the developed system helps improve the security and stability of DNS servers by preventing attacks and automatically notifying the administrator about events on the server.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ 8
ВСТУП 9
1 ОГЛЯД UDP FLOOD АТАК НА СЕРВІСИ ТА ЇХ НАСЛІДКІВ 11
1.1 Поняття UDP flood атаки 11
1.2 Мета та наслідки UDP flood атак 12
1.3 Загальні методи захисту від UDP flood атак 13
2 НАЛАШТУВАННЯ ЛАБОРАТОРНОГО КОМПЛЕКСУ ДЛЯ МОДЕЛЮВАННЯ UDP FLOOD АТАКИ НА DNS 20
2.1 DNS сервер як об’єкт атаки 20
2.1.1 Огляд роботи DNS серверів 20
2.2 Схема мережі та програмні засоби для UDP flood атаки 22
2.2.1 Схема мережі для здійснення UDP flood атаки 22
2.2.2 Маршрутизатор на базі операційна система OpenBSD 24
2.2.3 Операційна система Parrot Security як засіб атаки 29
2.2.4 DNS сервер на базі операційна система RedHat Linux 31
2.3 Здійснення DNS flood атаки на DNS -сервер 33
3 РОЗРОБКА ПРОГРАМНОГО МОДУЛЯ ЗАХИСТУ ВІД UDP FLOOD АТАК НА DNS СЕРВЕР 42
3.1 Розробка програмного модуля 42
3.1.1 Створення алгоритму виявлення та блокування UDP flood атаки 42
3.1.2 Налаштування брандмауера PF 43
3.1.3 Вибір середовища розробки 44
3.1.4 Написання програмного модуля 46
3.1.5 Автоматизація процесу відслідковування атаки та блокування IP зловмисника 47
3.2 Тестування програмного модуля 48
4 ОХОРОНА ПРАЦІ ТА БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ 53
4.1 Охорони праці 53

4.2 Шум, вібрація, ультразвук, електромагнітні випромінювання у виробничих приміщеннях для роботи з ВДТ та захист від них 56
ВИСНОВКИ 60
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ 62
Додаток А Публікація 64
Додаток Б Лістинг файлу udpflood_monitor.py 67

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

DoS	—	Denial of Service
UDP	—	User Datagram Protocol
DNS	—	Domain Name System
NAT	—	Network Address Translation
PF	—	Packet Filter
DMZ	—	Demilitarized Zone
BIND	—	Berkeley Internet Name Domain
TSIG	—	Transaction Signature
DNSSEC	—	DNS Security Extensions

ВСТУП

Актуальність теми. З ростом важливості мережевого середовища та збільшенням кількості інтернет-пристроїв актуальність захисту DNS серверів від UDP flood атак надзвичайно висока. Атаки на DNS інфраструктуру можуть призвести до недоступності важливих мережевих служб та великих фінансових втрат. Забезпечення стійкості і безпеки серверів DNS стає терміновою задачею для підтримки нормального функціонування мережі.

Мета і задачі дослідження. Метою кваліфікаційної роботи є розробка програмного модуля захисту DNS серверів від UDP flood атак з використанням операційної системи OpenBSD, системи фільтрації пакетів PF та скрипту на мові програмування Python.

Завданнями дослідження є аналіз сучасних методів захисту, розробка програмного модуля для виявлення і блокування атакуючих IP-адрес, та налаштування автоматичних повідомлень адміністратору.

Об’єкт дослідження. Об’єктом дослідження є мережева інфраструктура та DNS сервери, які піддаються UDP flood атакам.

Предмет дослідження. Предметом дослідження є розробка та імплементація програмного модуля, який забезпечує захист DNS серверів від UDP flood атак та автоматичне сповіщення адміністратора про інциденти.

Наукова новизна одержаних результатів кваліфікаційної роботи. Наукова новизна цієї роботи полягає в розробці та впровадженні комплексного рішення на базі OpenBSD і PF для захисту DNS серверів від UDP flood атак для ефективного виявлення та блокування атакуючих IP-адрес в режимі реального часу. Робота спрямована на вирішення практичної проблеми захисту мережевої інфраструктури від потенційно небезпечних атак.

Практичне значення одержаних результатів. Одержані результати мають велике практичне значення для адміністраторів мережевої інфраструктури, оскільки розроблена система допоможе підвищити безпеку і надійність DNS серверів, запобігаючи атакам і забезпечуючи автоматичне повідомлення адміністратору про події на сервері.

Апробація результатів магістерської роботи. Основні результати проведених досліджень обговорювались на: IV Міжнародній студентській конференції «Концепт науки XXI: стратегії, методи та наукові інструменти» (м.Вінниця, Україна).

Публікації. Основні результати кваліфікаційної роботи опубліковано у працях конференції (див. Додаток А).

1 ОГЛЯД UDP FLOOD АТАК НА СЕРВІСИ ТА ЇХ НАСЛІДКІВ

1.1 Поняття UDP flood атаки

UDP flood атака є однією зі специфічних форм DoS атак, спрямованих на перевантаження мережевих ресурсів та зниження доступності цільового сервісу або сервера. Ця атака використовує протокол UDP і полягає в надсиланні великої кількості UDP-пакетів на цільовий сервер або мережевий об'єкт [1].

Основні риси UDP flood атаки включають:

–Велика кількість пакетів. Зловмисник надсилає велику кількість UDP-пакетів на цільовий об'єкт. Ця кількість пакетів намагається перевищити оброблювальну здатність сервера або мережевого обладнання.

–Брак контролю. UDP являє собою протокол без з’єднання, відсутність механізмів підтвердження доставки пакетів робить його ідеальним для атаки. Зловмисник може надсилати пакети без очікування відповіді, що робить атаку важкою для виявлення та локалізації.

–Закрита обробка. У багатьох випадках атаки на UDP сервери призводять до переповнення буферів і відмови служби, оскільки сервер спробує обробити всі отримані пакети, не перевіряючи їхню легітимність. Це призводить до відмови в обслуговуванні для легітимних користувачів.

–Використання ботнетів. Часто атаки з використанням UDP flood проводяться з використанням ботнетів, тобто мережі компрометованих комп'ютерів або пристроїв, які контролюються зловмисниками. Це дозволяє збільшити потужність атаки та ускладнює виявлення та блокування атакуючих.

UDP flood атаки можуть призвести до значних проблем у доступності мережевих ресурсів та сервісів, що робить їх надзвичайно небезпечними. Захист від таких атак вимагає розробки та впровадження ефективних захисних механізмів, таких як автоматизовані системи для виявлення та блокування атакуючих, щоб забезпечити стійкість та надійність мережевого середовища.

1.2 Мета та наслідки UDP flood атак

Головна мета UDP flood атаки полягає в намаганні перевантажити цільовий сервер або мережевий об'єкт, надсилаючи на нього велику кількість UDP-пакетів протягом короткого періоду часу. Наслідком буде зниження доступності цільового об'єкта для легітимних користувачів і надмірне використання ресурсів, що може призвести до відмови в обслуговуванні [1].

UDP flood атаки можуть мати серйозні наслідки для цільового сервера та мережевого середовища в цілому:

– Недоступність сервісу. Однією з головних наслідків є недоступність цільового сервісу або ресурсу для легітимних користувачів. Сервер перестає відповідати на запити через перевищення оброблювальної здатності. Оскільки атака спрямована на DNS сервер, це може призвести до порушення послуги і неможливості користувачів отримати доступ до вебсайтів за доменними іменами. Це може призвести до великих фінансових втрат для підприємств і організацій.

– Надмірне використання мережевого обладнання. Атаки можуть спричинити надмірне використання мережевого обладнання, такого як маршрутизатори та комутатори, що може призвести до переповнення буферів та зниження продуктивності мережі.

–Втрата даних. У деяких випадках UDP flood атаки можуть призвести до втрати даних, оскільки пакети можуть бути втрачені в мережі без можливості їх відновлення.

–Зниження довіри. Негативні наслідки UDP flood атак можуть призвести до зниження довіри користувачів до обслуговуючих організацій та послуг.

– Споживання мережевого трафіку. Велика кількість надісланих UDP-пакетів може споживати значну частину мережевого трафіку, що може вплинути на продуктивність і доступність інших сервісів в мережі.

– Збій у внутрішній комунікації. UDP flood атака може вплинути на внутрішню комунікацію в організації, особливо якщо сервери, що обслуговують цю комунікацію, стають недоступними.

– Можливість використання для відволікання уваги. UDP flood атака може використовуватися зловмисниками як спосіб відволікти увагу від інших злочинних дій в мережі, таких як вторгнення та крадіжка даних.

З урахуванням серйозних наслідків UDP flood атак, важливо мати вдосконалені захисні механізми та стратегії для виявлення та запобігання цим атакам з метою забезпечення безпеки та доступності мережевих ресурсів і служб.

Важливо бути готовими до їхнього виявлення та запобігання, а також мати плани відновлення, які допоможуть знизити вплив атаки на бізнес-процеси та обслуговування користувачів.

1.3 Загальні методи захисту від UDP flood атак

Захист від UDP flood атак вимагає комплексного підходу та використання різноманітних заходів для виявлення, блокування і пом'якшення наслідків атаки [2]. Нижче розглянуті загальні методи та стратегії для захисту від UDP flood атак:

– Фільтрація на рівні мережевого обладнання. Використання брандмауера або інших мережевих пристроїв для фільтрації UDP-пакетів ще на рівні мережі може допомогти виявити та блокувати атакуючі пакети перед тим, як вони досягнуть цільового сервера. Цей метод може використовувати правила фільтрації, які базуються на джерелі, призначенні, часових інтервалах та інших параметрах.

– Захист на рівні сервера. На цільовому сервері можуть бути налаштовані заходи захисту, такі як обмеження частоти прийому UDP-запитів з одного IP-адреса, відмова у відповіді на недійсні запити.

– Моніторинг трафіку і виявлення аномалій. Використання систем моніторингу та аналізу трафіку дозволяє виявляти аномальну активність, включаючи збільшення обсягу UDP-пакетів на певний порт або від певного джерела. Системи моніторингу можуть виявити атаку та сповістити адміністраторів.

– Захист від ботнетів. Посилення заходів захисту від ботнетів є критичним для боротьби з UDP flood атаками. Це може включати в себе виявлення та відключення скомпрометованих пристроїв, вдосконалення систем виявлення вторгнень.

– Доступність резервних серверів. Наявність резервних серверів та механізмів перенаправлення трафіку може допомогти забезпечити доступність сервісу під час атаки, перенаправляючи трафік на менш навантажені сервери.

– Виявлення та відповідь на атаку. Важливо мати план виявлення та відповіді на UDP flood атаку, який включає в себе миттєве виявлення атаки, блокування атакуючих IP-адрес, і моніторинг стану сервера під час та після атаки.

– Організаційні заходи. Залучення адміністраторів мережі, розробка стратегій і планів відновлення, а також тренування персоналу щодо виявлення та реагування на атаки важливо для успішного захисту від UDP flood атак.

Брандмауери є ключовими компонентами мережевого обладнання, які можуть використовуватися для фільтрації трафіку. Вони дозволяють налаштовувати правила фільтрації на основі різних параметрів, таких як IP-адреси джерела, порти, протоколи та зразки трафіку. Брандмауери можуть використовувати списки блокування IP-адрес для автоматичного блокування атакуючих IP-адрес, визначених як джерела атаки. Це може включати IP-адреси, які відправляють надмірну кількість UDP-пакетів протягом короткого часу. Брандмауери можуть встановлювати пороги для кількості UDP-пакетів, які можуть надсилати один IP-адрес за певний період часу. Якщо IP-адрес перевищує ці обмеження, його трафік може бути автоматично заблокований. Вони можуть мати спеціалізовані правила фільтрації, спрямовані на виявлення типових атак на протокол UDP. Наприклад, вони можуть виявляти атаки на UDP DNS-сервери за специфічними патернами запитів. Великі мережі можуть використовувати розподілені брандмауери, розташовані на різних вузлах мережі, щоб забезпечити захист від UDP flood атак на більшому масштабі та з більшою пропускною здатністю. Брандмауери можуть здійснювати моніторинг трафіку та

вести журнал подій. Це допомагає адміністраторам виявляти аномальну активність і аналізувати атаки для подальшого вдосконалення захисних заходів.

Використання фільтрації на рівні мережевого обладнання є ефективним методом захисту від UDP flood атак, оскільки це дозволяє виявляти і блокувати атакуючий трафік на рівні самої мережі, зменшуючи навантаження на цільовий сервер та мережу в цілому. Комбінування фільтрації на рівні мережі з іншими захисними стратегіями робить мережеву інфраструктуру більш стійкою до UDP flood атак і забезпечує надійність служб та ресурсів

Захист на рівні сервера є ще однією важливою складовою стратегії захисту від UDP flood атак. Цей підхід включає в себе налаштування захисних заходів безпеки безпосередньо на цільових серверах, що піддаваються атаці, і спрямований на обмеження впливу атаки на їхню роботу. Сервери можуть бути налаштовані для обмеження частоти прийому UDP-запитів з одного IP-адреса. Це допомагає уникнути перевантаження сервера великою кількістю запитів від одного джерела. Якщо певна IP-адреса надсилає надмірну кількість запитів, сервер може відмовити у відповіді або відхилити деякі запити. Сервери можуть бути налаштовані для відмови у відповіді на недійсні або аномальні UDP-запити. Це включає в себе відхилення запитів з невірними або невідомими полями, які можуть бути характерними для UDP flood атак. Це допомагає запобігти використанню сервера для ампліфікації атаки. Сервери можуть бути налаштовані для обмеження доступу до певних ресурсів чи послуг з певних IP-адрес або діапазонів IP-адрес. Це дозволяє відмовити у доступі атакуючим IP-адресам. Сервери повинні мати системи моніторингу та логування, які дозволяють виявляти аномальну активність та атаки. Це може включати в себе аналіз журналів сервера та моніторинг використання ресурсів. Регулярне оновлення програмного забезпечення сервера, операційної системи та додатків є критичним для зменшення вразливостей, які можуть бути використані для запуску UDP flood атак.

Захист на рівні сервера є важливою складовою загальної стратегії захисту від UDP flood атак і допомагає забезпечити надійну роботу окремих серверів та послуг. Комбінування захисту на рівні мережевого обладнання та захисту на

рівні сервера робить мережеву інфраструктуру більш стійкою до подібних атак і забезпечує безпеку та доступність ресурсів.

Моніторинг трафіку і виявлення аномалій є важливими компонентами захисту від UDP flood атак. Цей підхід полягає у нагляді за мережевим трафіком та виявленні будь-яких аномальних патернів чи активності, які можуть вказувати на атаку або недоречну поведінку. Моніторинг починається з збору даних про мережевий трафік. Це може включати в себе аналіз заголовків UDP-пакетів, які містять інформацію про джерело, призначення, порти та інші параметри. Переважна більшість UDP пакетів повинна мати нормальну, очікувану структуру та розподіл. Моніторинг дозволяє збирати статистику та порівнювати активність зі стандартами для виявлення аномалій. Однією з основних ознак UDP flood атак є раптове та значуще збільшення кількості UDP-пакетів, які надсилаються на цільовий сервер чи мережу. Моніторинг дозволяє виявити такі стрибки в активності. Моніторинг може виявити атаки, які використовують ампліфікацію, де атакуючі використовують сервери або послуги, щоб збільшити обсяг трафіку. Виявлення цього типу атак допомагає швидко вжити заходів для їхнього припинення. Виявлення аномалій може базуватися на аналізі поведінки трафіку. Наприклад, сервери можуть спостерігати за змінами у частоті запитів, кількості IP-адрес, які надсилають запити, та іншими факторами. Моніторинг систем можуть бути інтегровані з системами сповіщення, що автоматично повідомляють адміністраторів про виявлення аномалій або атак. Це дозволяє адміністраторам реагувати швидко на потенційно небезпечні ситуації.

Моніторинг трафіку і виявлення аномалій дозволяють реагувати на UDP flood атаки швидко та ефективно, виявляючи їх на ранніх стадіях і запускаючи захисні заходи. Цей підхід є важливою частиною загальної стратегії захисту мережі і дозволяє забезпечити безпеку та доступність ресурсів та послуг.

Захист від ботнетів є критично важливим елементом стратегії захисту від UDP flood атак і інших видів мережевих атак. Ботнети представляють собою мережу комп'ютерів, які були заражені шкідливим програмним забезпеченням. Щоб запобігти комп'ютерам у вашій мережі стати частиною ботнету, важливо встановити і регулярно оновлювати антивірусне програмне забезпечення на всіх

комп'ютерах та серверах. Це допомагає виявляти та видаляти шкідливі програми. Регулярно оновлюйте операційні системи і програмне забезпечення на всіх комп'ютерах. Виробники випускають патчі для виправлення вразливостей, які можуть бути використані ботнетами для зараження. Спам та фішингові листи можуть містити посилання на шкідливий код або віруси, які можуть заражати комп'ютери. Важливо використовувати ефективний антиспам та антифішинг захист. Обмеження доступу до мережі та ресурсів на рівні користувача або пристрою може допомогти запобігти небажаним підключенням до ботнетів.

Проведення навчання користувачів щодо безпеки в мережі. Вміння розпізнавати підозрілі повідомлення та призначення файлів, а також проявляти обережність під час завантаження та відкриття веб-сторінок є важливим фактором боротьби з ботнетами..

Захист від ботнетів вимагає комплексного підходу, який включає в себе технічні, організаційні та психологічні заходи. Важливо регулярно переглядати та підтримувати заходи захисту, оскільки загрози від ботнетів постійно розвиваються.

Забезпечення доступності резервних серверів є важливою складовою стратегії захисту від UDP flood атак та інших мережевих атак. Резервні сервери - це альтернативні сервери, готові приймати навантаження у разі відмови основного сервера. Основна мета резервних серверів полягає в забезпеченні безперервності роботи служб та додатків навіть під час атак або інших негативних подій. Архітектура високої доступності передбачає наявність не тільки одного резервного сервера, але і цілого кластера серверів, які можуть брати на себе навантаження. Це забезпечує високий рівень доступності і зменшує вплив атак на окремі сервери. Використання систем балансування навантаження (Load Balancing) дозволяє розподілити трафік між резервними серверами рівномірно, забезпечуючи оптимальне використання ресурсів і запобігаючи перевантаженню окремих серверів. Резервні сервери повинні бути готові швидко приймати навантаження і відновлювати роботу служб після виявлення відмови основного сервера. Це може включати в себе резервне копіювання даних і налаштувань. Резервні сервери можуть бути розташовані в різних географічних

регіонах для забезпечення надійності в умовах надзвичайних ситуацій або при глобальних атаках. Потрібно проводити регулярні тести на відновлення для перевірки, наскільки швидко і ефективно резервні сервери можуть приймати навантаження після відмови основного сервера. Мають бути створені документовані процедури відновлення в разі відмови основного сервера, які включають в себе кроки для активації резервних серверів та перенесення трафіку.

Забезпечення доступності резервних серверів є важливим запобіжним заходом проти UDP flood атак та інших типів атак. Це гарантує, що ваша інфраструктура залишиться надійною та доступною, навіть під час негативних сценаріїв.

Виявлення та відповідь на атаку є важливим етапом в захисті від UDP flood атак та інших мережевих атак. Під час цього етапу ви спрямовуєте зусилля на виявлення атак та вживання заходів для обмеження їхнього впливу на ваші системи та мережу. Потрібно використовувати системи моніторингу мережі та журналізації, щоб виявити незвичайну активність, таку як збільшення кількості UDP-пакетів, які надходять на сервери або мережу. Моніторинг повинен бути постійним і систематичним. Також слід моніторити використання ресурсів, таких як процесор, пам'ять та мережева пропускна здатність, для виявлення аномалій, які можуть бути спричинені атаками. Системи сповіщення повинні автоматично повідомляти адміністраторів про виявлення атаки. Це допоможе реагувати на атаку швидко. Після виявлення атаки слід вживати заходів для обмеження її впливу на систему. Це може включати в себе блокування IP-адрес атакуючих, перенесення навантаження на резервні сервери, а також впровадження заходів безпеки на рівні мережі та серверів. Після того, як атака була відповідно припинена або обмежена, потрібно провести детальний аналіз атаки, щоб з'ясувати, яким чином вона відбувалася і які уразливості були використані. Це допоможе вдосконалити стратегії захисту для запобігання майбутнім атакам. Також мають бути розроблені плани відновлення, які включають в себе кроки для відновлення роботи систем та служб після атаки.

Виявлення та відповідь на атаку є критичними етапами в захисті від UDP flood атак та інших мережевих загроз. Швидка реакція і використання відповідних інструментів та стратегій може значно зменшити вплив атаки на інфраструктуру та забезпечити надійність та доступність послуг і ресурсів.

Захист від UDP flood атак - це постійний процес, оскільки зловмисники намагаються постійно вдосконалювати свої методи. Ефективна стратегія захисту включає в себе поєднання усіх ресурсів для забезпечення безпеки та доступності мережевих ресурсів та послуг.

2 НАЛАШТУВАННЯ ЛАБОРАТОРНОГО КОМПЛЕКСУ ДЛЯ МОДЕЛЮВАННЯ UDP FLOOD АТАКИ НА DNS

2.1 DNS сервер як об’єкт атаки

2.1.1 Огляд роботи DNS серверів

DNS сервери відповідають за перетворення доменних імен (наприклад, www.meta.ua) в IP-адреси, які використовуються для ідентифікації серверів у мережі [3].

DNS сервери можуть бути налаштовані для обслуговування конкретних доменних зон (наприклад, .com, .org, .net тощо). Якщо DNS сервер не має інформації про запит, він може виконати рекурсивний запит до іншого DNS сервера, який може мати необхідну інформацію. DNS сервери можуть кешувати результати запитів, щоб зменшити навантаження на мережу і прискорити обробку подібних запитів у майбутньому. Кеш може зберігати результати запитів на певний період часу. DNS сервери зберігають різні типи записів, такі як A-записи (для IPv4 адрес), AAAA-записи (для IPv6 адрес), MX-записи (для поштових серверів), CNAME-записи (для псевдонімів), і багато інших. Кожен тип запису використовується для конкретного завдання в системі DNS. Багато DNS серверів використовують механізми реплікації для забезпечення доступності та надійності. Резервні DNS сервери можуть мати копії зон даних основного сервера і можуть автоматично відповідати на запити у разі відмови основного сервера. DNSSEC - це набір розширень для DNS, які забезпечують підписи та перевірку цілісності даних DNS. Вони допомагають запобігти DNS-атакам, які спрямовані на злам DNS трафіку. DNS сервери можуть бути цільовими об'єктами різних атак, включаючи UDP flood атаки, DNS ампліфікація (коли зловмисник використовує DNS запити для підсилення своїх атак), і отруєння кешу DNS. Захист від цих атак включає в себе моніторинг трафіку, фільтрацію небажаних запитів та використання DNSSEC.

DNS використовує протоколи, такі як UDP (для запитів) і TCP (для великих відповідей або запитів, що потребують рекурсивного пошуку). DNS сервери стандартно слухають на 53 UDP і 53 TCP портах.

Існують різні реалізації DNS серверів, такі як BIND, Microsoft DNS Server, PowerDNS, Unbound і багато інших. Кожен з них має свої особливості та конфігураційні параметри.

Для захисту DNS серверів важливо використовувати системи моніторингу, щоб виявляти незвичайну активність та атаки, а також для відстеження використання ресурсів сервера.

Для ефективного захисту DNS серверів від UDP flood атак і інших загроз важливо належно налаштувати інфраструктуру DNS, використовувати методи захисту, такі як фільтрація і обмеження запитів, і регулярно оновлювати програмне забезпечення та при потребі конфігурацію сервера.

2.1.2 UDP flood атаки на DNS

UDP flood атаки на DNS є однією зі специфічних форм мережевих атак, які спрямовані на DNS сервери. Ці атаки спробують перевантажити цільовий DNS сервер великою кількістю надмірних UDP запитів, заважаючи нормальному функціонуванню DNS і викликаючи недоступність сервісу [4].

DNS використовує протокол UDP для передачі запитів і відповідей. UDP - це протокол без з'єднання, що не передбачає підтвердження доставки пакетів. Це робить його ідеальним для атак типу flood, де атакуючий може надсилати велику кількість UDP-пакетів на цільовий сервер.

У цьому типі атаки цільовим об'єктом є DNS сервер, який відповідає за розпізнавання доменних імен та повернення відповідей. Атака спрямована на перевантаження ресурсів цього сервера, щоб зробити його недоступним для легітимних користувачів.

Атакуючі можуть використовувати ботнети або інші мережеві ресурси для надсилання паралельних запитів на DNS сервер (Рисунок 2.1). Це дозволяє збільшити інтенсивність атаки і здатність до перевантаження сервера.

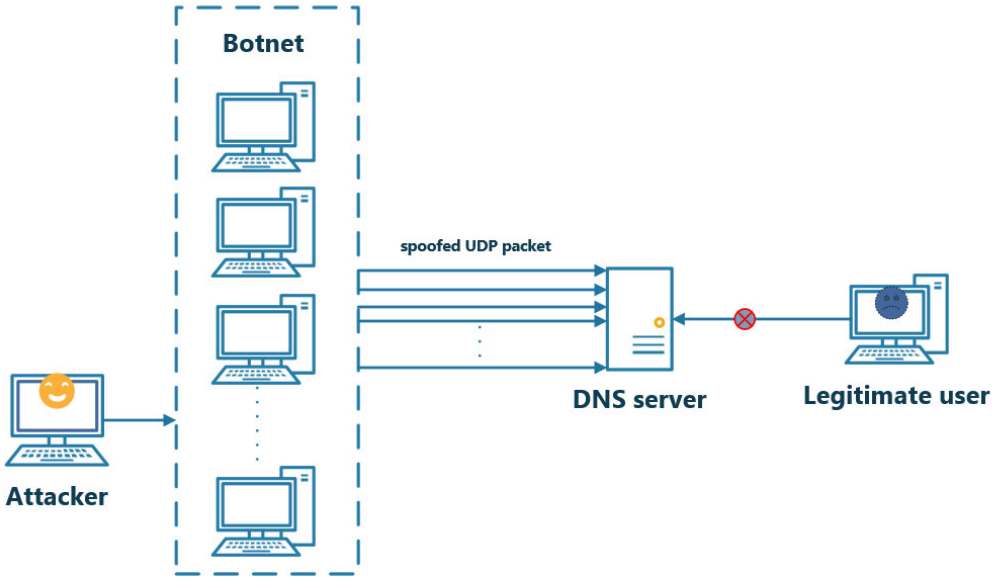


Рисунок 2.1 – Загальний принци здійснення UDP flood атаки

Добре налаштовані системи моніторингу можуть виявити незвичайну активність, яка вказує на UDP flood атаку на DNS. Адміністратори повинні отримувати сповіщення про такі атаки, щоб реагувати на них якнайшвидше.

Заходи захисту включають в себе використання брандмауера для фільтрації надмірних запитів DNS, обмеження доступу до DNS серверів лише для довірених джерел, використання DNSSEC для перевірки цілісності запитів та відповідей, а також моніторинг та логування активності DNS серверів.

2.2 Схема мережі та програмні засоби для UDP flood атаки

2.2.1 Схема мережі для здійснення UDP flood атаки

Для проведення UDP flood атаки буде використана схема мережі показана на рисунку 2.2.

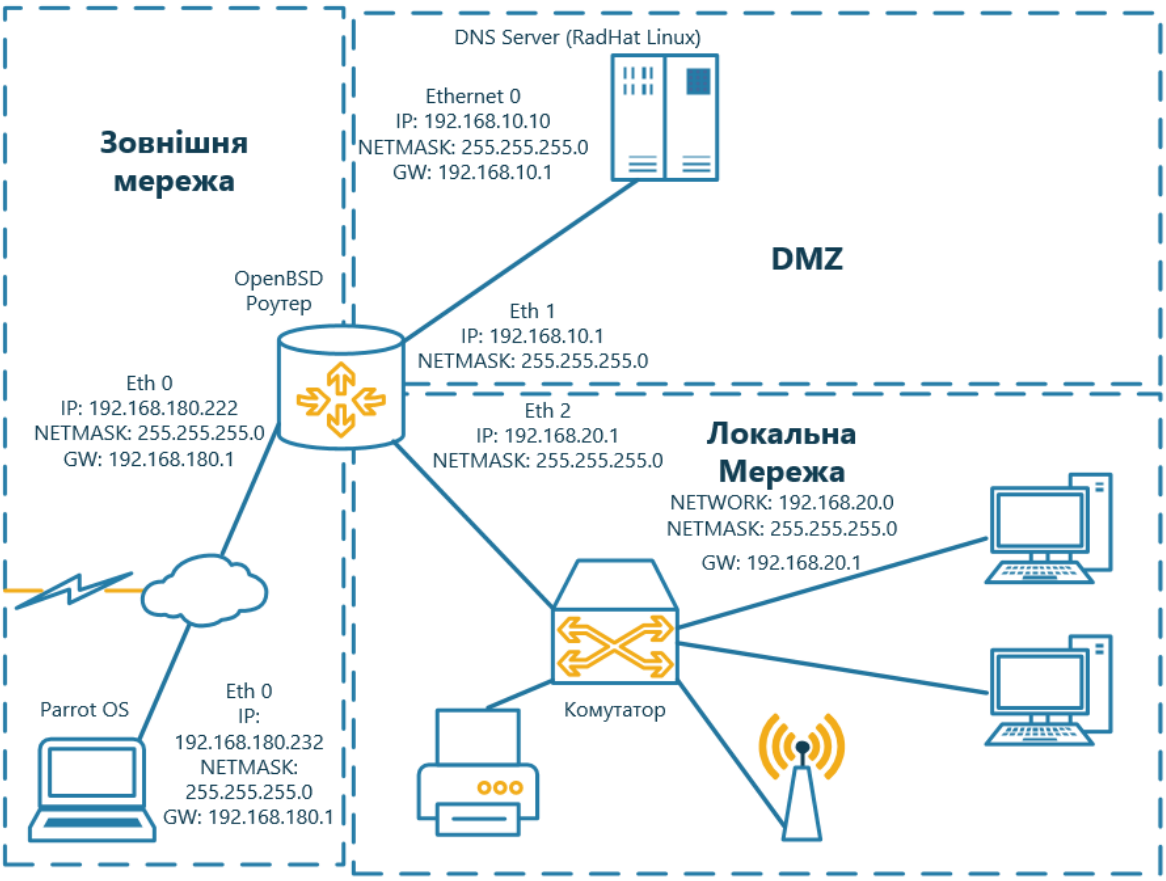


Рисунок 2.2 – Схема мережі для здійснення UDP flood атаки

Схема мережі для демонстрації UDP flood атаки на DNS складається з наступних елементів:

1) Маршрутизатор на базі операційної системи OpenBSD [5]. OpenBSD буде використовувати як шлюз з NAT та брандмауером, використовуючи брандмауер PF [6]. Така конфігурація дозволяє контролювати трафік між локальною мережею і зовнішньою мережею та DMZ, а також приховувати локальні IP-адреси від інтернету через NAT.

2) Атакуючий комп'ютер на базі операційної системи Parrot Security [7]. Цей комп'ютер відіграє роль атакуючого. Він генерує і надсилає велику кількість UDP-пакетів на цільовий DNS сервер.

3) Цільовий DNS сервер на базі операційної системи RedHat Linux [8]. Цей сервер є об'єктом атаки. Атакуючий комп'ютер надсилає велику кількість UDP-пакетів на цей сервер з метою перевантажити його мережеві ресурси. DNS сервер розташований в зоні DMZ.

Зона DMZ - це сегмент мережі, який відділений від внутрішньої мережі і призначений для розміщення ресурсів, які повинні бути доступними з Інтернету або зовнішньої мережі, але при цьому ізольовані від основної внутрішньої мережі для забезпечення безпеки.

4) Локальна мережа. В локальній мережі розміщено клієнтську комп’ютери та мережеве обладнання, яке забезпечує зв'язок між всіма комп'ютерами та серверами в мережі.

Дана схема мережі допоможе навчитися розпізнавати, захищати та реагувати на UDP flood атаки у контрольованому середовищі.

2.2.2 Маршрутизатор на базі операційної системи OpenBSD.

OpenBSD - це відкрите операційне середовище, яке є однією з реалізацій сімейства UNIX-подібних операційних систем [5]. Ця система славиться своєю спрямованістю на безпеку та стабільність. OpenBSD розпочала свою історію як форк NetBSD у 1995 році, відтоді вона еволюціонувала самостійно і стала популярною в спільноті, яка цінує безпеку і відкритий код. Розробниками OpenBSD є висококваліфіковані фахівці з безпеки. Ця ОС активно видаляє потенційні уразливості, відмовляється від ввімкнених за замовчуванням сервісів, та має ряд заходів безпеки, які роблять її важкою мішенню для атак. OpenBSD поставляється з широким набором інструментів для розробки, включаючи компілятори, відладчики, систему контролю версій і інші ресурси для програмістів. Підтримує багато різних архітектур, включаючи x86, x64, ARM, SPARC, PowerPC, і багато інших. OpenBSD славиться своєю підтримкою мережевих технологій, включаючи брандмауери, віртуальні приватні мережі, IPsec, IPv6, і багато інших. Система також має вбудований аналізатор мережевого трафіку, який допомагає виявляти проблеми. Всі компоненти OpenBSD поширюються під ліцензією, яка дозволяє вільне використання, модифікацію та розповсюдження. Це робить систему популярною серед вільної та відкритої спільноти. OpenBSD включає в себе інструменти, призначені для тестування та захоплення мережевого трафіку, такі як tcpdump.

OpenBSD має власну систему управління пакетами, відому як "pkg". Вона дозволяє користувачам легко встановлювати, оновлювати та видаляти пакети програмного забезпечення. OpenBSD співпрацює зі спільнотою безпеки і змагається за створення безпечних та надійних рішень у сфері комп'ютерної безпеки. Відомий проект OpenSSH, який широко використовується для безпечних з'єднань SSH, виник в рамках OpenBSD.

OpenBSD - це потужна операційна система, яка відзначається своєю акцентом на безпеку, стабільність і відкритий код. Вона широко використовується як операційна система для серверів та мережевих пристроїв у приватних і комерційних проектах. Це включає в себе брандмауери та маршрутизатори.

Маршрутизатор є невід'ємною частиною будь-якої мережі. Операційна система OpenBSD, спроектована з врахуванням безпеки, стала популярним вибором для створення маршрутизаторів з функцією NAT та брандмауера PF.

Файл rc.conf - це конфігураційний файл, який використовується в операційній системі OpenBSD для налаштування різних параметрів та служб під час завантаження системи. Цей файл містить змінні та їхні значення, які визначають різні параметри системи. На рисунку 2.3 показано конфігураційний файл rc.conf.

```
#!/bin/sh
hostname="router.cstntu.lan"
#LAN
ifconfig_em2="inet 192.168.20.1 netmask 255.255.255.0"
#DMZ
ifconfig_em1="inet 192.168.10.1 netmask 255.255.255.0"
#WAN
ifconfig_em0="inet 192.168.180.222 netmask 255.255.255.0"
defaultrouter="192.168.180.1"
#
gateway_enable="YES"
#
sshd_enable="YES"
moused_enable="YES"
ntpdate_enable="YES"
ntpd_enable="YES"
#
pf_enable="yes"
pf_rules="/etc/pf.conf"
pflog_enable="yes"
pflog_logfile="/var/log/pflog"
#
```

Рисунок 2.3 – Конфігураційний файл rc.conf

Розглянемо докладно кожен параметр у файлі rc.conf:

–hostname="router.cstntu.lan". Цей параметр визначає ім'я хоста (назву комп'ютера) системи. У цьому випадку, ім'я хоста встановлене як "router.cstntu.lan";

–ifconfig_em2="inet 192.168.20.1 netmask 255.255.255.0". Цей рядок налаштовує інтерфейс em2 для LAN мережі з IP-адресою "192.168.20.1" і маскою підмережі "255.255.255.0";

–ifconfig_em1="inet 192.168.10.1 netmask 255.255.255.0". Цей рядок вказує налаштування для інтерфейсу em2, який використовується для зони DMZ (Demilitarized Zone) нашої мережі. Вказана IP-адреса - "192.168.10.1", а маска підмережі - "255.255.255.0";

–ifconfig_em0="inet 192.168.180.222 netmask 255.255.255.0". Цей параметр встановлює IP-адресу та маску підмережі для інтерфейсу em0, який є зовнішнім інтерфейсом маршрутизатора. IP-адреса в цьому прикладі - "192.168.180.222", а маска підмережі - "255.255.255.0";

–defaultrouter="192.168.180.1". Ця стрічка вказує IP-адресу нашого шлюза за замовчуванням для вихідного трафіку;

–gateway_enable="YES". Цей параметр вмикає функцію маршрутизації на нашій системі, дозволяючи їй маршрутизувати пакети між різними мережами (інтерфейсами) ;

–sshd_enable="YES". Вмикає службу SSH (Secure Shell), щоб дозволити віддалений доступ до системи через SSH-протокол;

–pf_enable="yes". Вмикає брандмауер Packet Filter (PF) на нашій системі, що дозволяє контролювати трафік на рівні пакетів;

–pf_rules="/etc/pf.conf". Вказує шлях до конфігураційного файлу PF, який буде використовуватися для налаштування правил брандмауера;

–pflog_enable="yes". Вмикає журналювання брандмауера PF для запису журналів трафіку.

NAT - це технологія, яка дозволяє приховати приватні IP-адреси внутрішньої мережі за однією або кількома публічними IP-адресами. Коли трафік виходить до мережі Інтернет через зовнішній інтерфейс, NAT перекладає джерело та призначення пакетів, забезпечуючи приховування приватних IP-адрес від зовнішнього світу. OpenBSD підтримує NAT через інструменти, такі як pfctl і конфігураційний файл pf.conf.

PF - це брандмауер, розроблений для OpenBSD, який надає потужні можливості фільтрації та контролю трафіку на рівні пакетів. Він дозволяє налаштовувати правила для переадресації, блокування, фільтрації та контролю трафіку між різними мережевими інтерфейсами [6].

Файл pf.conf - це конфігураційний файл брандмауера PF в операційній системі OpenBSD. В цьому файлі визначаються правила фільтрації та керування трафіком на рівні пакетів. На рисунку 2.4 наведений конфігураційний файлу pf.conf нашого маршрутизатора.

```
ext_if = "em0"
dmz_if = "em1"
int_if = "em2"
dmznet = $dmz_if:network
localnet = $int_if:network
#
set skip on lo0
set limit { states 4000000, frags 4000000, src-nodes 400000 }
set optimization aggressive
#
scrub in all
#
nat on $ext_if from $localnet to any -> ($ext_if)
#
block all
#VPN
pass in inet proto udp to $ext_if port { 1701,500,4500 }
pass in inet proto tcp to $ext_if port { 1723 } keep state
#
pass from { lo0, $localnet, $dmznet } to any keep state
#
block in log quick on $ext_if from { 127.0.0.0/8, 192.168.0.0/16, 172.16.0.0/12,
 10.0.0.0/8, 169.254.0.0/16, 192.0.2.0/24, 0.0.0.0/8, 240.0.0.0/4 }
```

Рисунок 2.4 – Конфігураційний файл pf.conf

Розглянемо докладно кожен параметр у файлі pf.conf:

–`ext_if = "em0", dmz_if = "em1", int_if = "em2"`. Ці рядки визначають імена мережевих інтерфейсів маршрутизатора, включаючи зовнішній інтерфейс (ext_if), інтерфейс DMZ (dmz_if) і локальний інтерфейс (int_if);

–`dmznet = $dmz_if:network, localnet = $int_if:network`. Ці рядки визначають IP-адреси і маски підмереж для інтерфейсів DMZ і локальної мережі, використовуючи значення змінних dmz_if і int_if;

–`set skip on lo0`. Ця команда вказує брандмауеру пропускати (не аналізувати) весь трафік, що проходить через інтерфейс lo0 (локальний інтерфейс петлі);

–`set limit { states 4000000, frags 4000000, src-nodes 400000 }`. Визначає обмеження на кількість станів (states), фрагментів (frags) та джерел (src-nodes). Це допомагає уникнути перевантаження брандмауера та витрати ресурсів;

–`set optimization aggressive`. Встановлює агресивну оптимізацію для більшої продуктивності брандмауера;

–`scrub in all`. Ця команда включає функцію "прибирання" (scrubbing), яка обрізає та очищає пакети, щоб запобігти можливим атакам, особливо тим, які використовують пакети з некоректними заголовками;

–`nat on $ext_if from $localnet to any -> ($ext_if)`. Встановлює правило для NAT. Це дозволяє приватній локальній мережі використовувати зовнішню IP-адресу (зовнішній інтерфейс) при виході в Інтернет. Весь трафік з локальної мережі буде маршрутизуватися через зовнішній інтерфейс;

–`block all`. Забороняє весь трафік за замовчуванням, якщо він не відповідає жодному із наступних правил;

–`pass in inet proto udp to $ext_if port { 1701, 500, 4500 }`. Ці правила дозволяють UDP-трафік на певні порти на зовнішньому інтерфейсі, що використовується для VPN-підключень;

–`pass in inet proto tcp to $ext_if port { 1723 } keep state`. Це правило дозволяє TCP-трафіку на певний порт на зовнішньому інтерфейсі. Порт 1723 використовується для PPTP VPN;

—`pass from { lo0, $localnet, $dmznet } to any keep state`. Це правило дозволяє трафіку з інтерфейсів lo0, em2 (локальна мережа) і em1 (DMZ) проходити через будь-який інтерфейс та зберігає стан з'єднання;

—`block in log quick on $ext_if from { 127.0.0.0/8, 192.168.0.0/16, 172.16.0.0/12, 10.0.0.0/8, 169.254.0.0/16, 192.0.2.0/24, 0.0.0.0/8, 240.0.0.0/4 }`. Це правило блокує трафік з резервованих IP-діапазонів та недійсних IP-адрес, таких як IP-адреси локальної мережі та інші.

2.2.3 Операційна система Parrot Security як засіб атаки

Parrot Security - це дистрибутив Linux, який базується на Debian і розробляється спеціально з огляду на кібербезпеку [7]. Parrot Security містить широкий набір інструментів для кібербезпеки, включаючи сканери портів, інструменти для аналізу мережевого трафіку, програми для тестування на проникнення, засоби аналізу вразливостей та інші. Він базується на стабільному ядрі Debian Linux, що забезпечує стабільність та надійність операційної системи.

Parrot Security містить інструменти для забезпечення анонімності та приватності в Інтернеті, такі як Tor і анонімні мережі.

На рисунку 2.5 показано мережеві налаштування операційної системи Parrot Security.

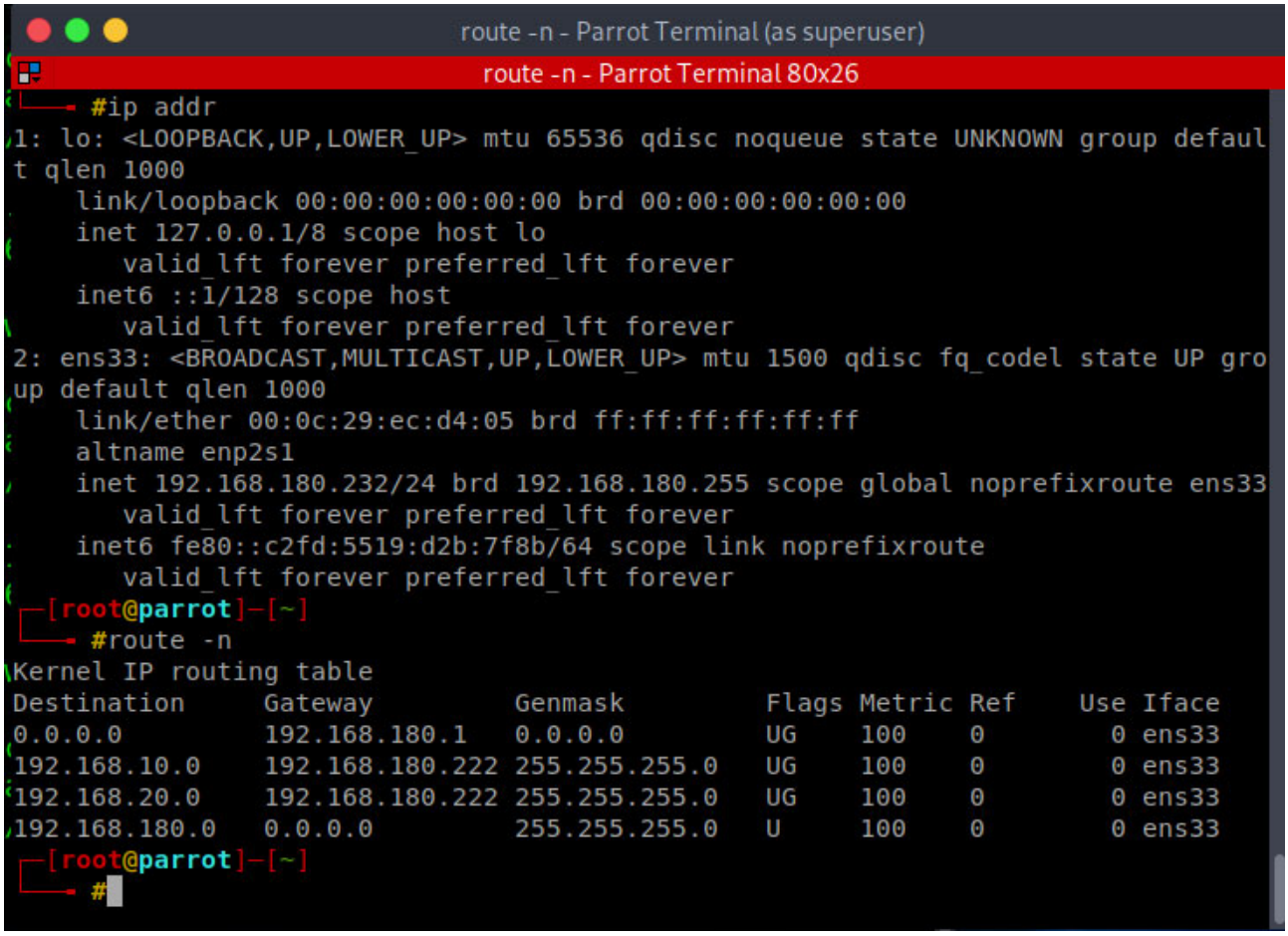


Рисунок 2.5 – Мережеві налаштування операційної системи Parrot Security

На атакуючому комп'ютері буде використано hping3 утиліту, для здійснення UDP flood атак [9]. hping3 - це інструмент для маніпулювання мережевим трафіком як в лабораторних умовах для тестування мережі так і в реальних у мовах. Програма підтримує роботу з різними мережевими протоколами, включаючи ICMP, TCP, UDP і деякі інші. Можна вибирати протокол, використовуючи відповідні параметри командного рядка.

Утиліта може генерувати UDP-пакети для надсилання на вказаний хост і порт. Це дозволяє симулювати UDP-трафік до цільового сервера.

Також hping3 дозволяє підробити адресу джерела, що дозволяє здійснити атаку з іншої IP-адреси. hping3 дозволяє налаштувати інтервали між надсиланням пакетів, що може впливати на інтенсивність атаки.

2.2.4 DNS сервер на базі операційна система RedHat Linux

Red Hat Linux - це комерційний дистрибутив операційної системи Linux, який розробляється і підтримується американською компанією Red Hat, Inc. Red Hat Linux відомий своєю стабільністю, безпекою та підтримкою.

Red Hat Linux надає користувачам комерційну підтримку, включаючи доступ до оновлень безпеки, підтримки та консультаційних послуг. Це робить Red Hat Linux популярним в корпоративних середовищах, де безпека і стабільність важливі. Red Hat Linux часто використовується як серверна операційна система для вебсерверів, баз даних, DNS, електронної пошти та інших серверних завдань. Red Hat пропонує різні рішення для цих сценаріїв, включаючи Red Hat Enterprise Linux Server.

На рисунку 2.6 показано мережеві налаштування операційної системи Red Hat Linux.

```
[root@redhat9srv network]# ifconfig
lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 286  bytes 30790 (30.0 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 286  bytes 30790 (30.0 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

torealworld1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.10.10  netmask 255.255.255.0  broadcast 192.168.10.255
        inet6 fe80::c11a:2d92:8f57:e7c7  prefixlen 64  scopeid 0x20<link>
        ether 00:0c:29:36:9b:17  txqueuelen 1000  (Ethernet)
        RX packets 68110730  bytes 4360928649 (4.0 GiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 2118  bytes 179063 (174.8 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

[root@redhat9srv network]# route -n
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
0.0.0.0         192.168.10.1    0.0.0.0         UG    100    0        0 torealworld1
192.168.10.0    0.0.0.0         255.255.255.0   U     100    0        0 torealworld1
[root@redhat9srv network]#
```

Рисунок 2.6 – Мережеві налаштування операційної системи Red Hat Linux

BIND - це одна з найпопулярніших програм для реалізації DNS-сервера в операційних системах Linux і UNIX [10]. BIND дозволяє зберігати інформацію про доменні імен у розподілених сховищах даних. Це означає, що інформація про

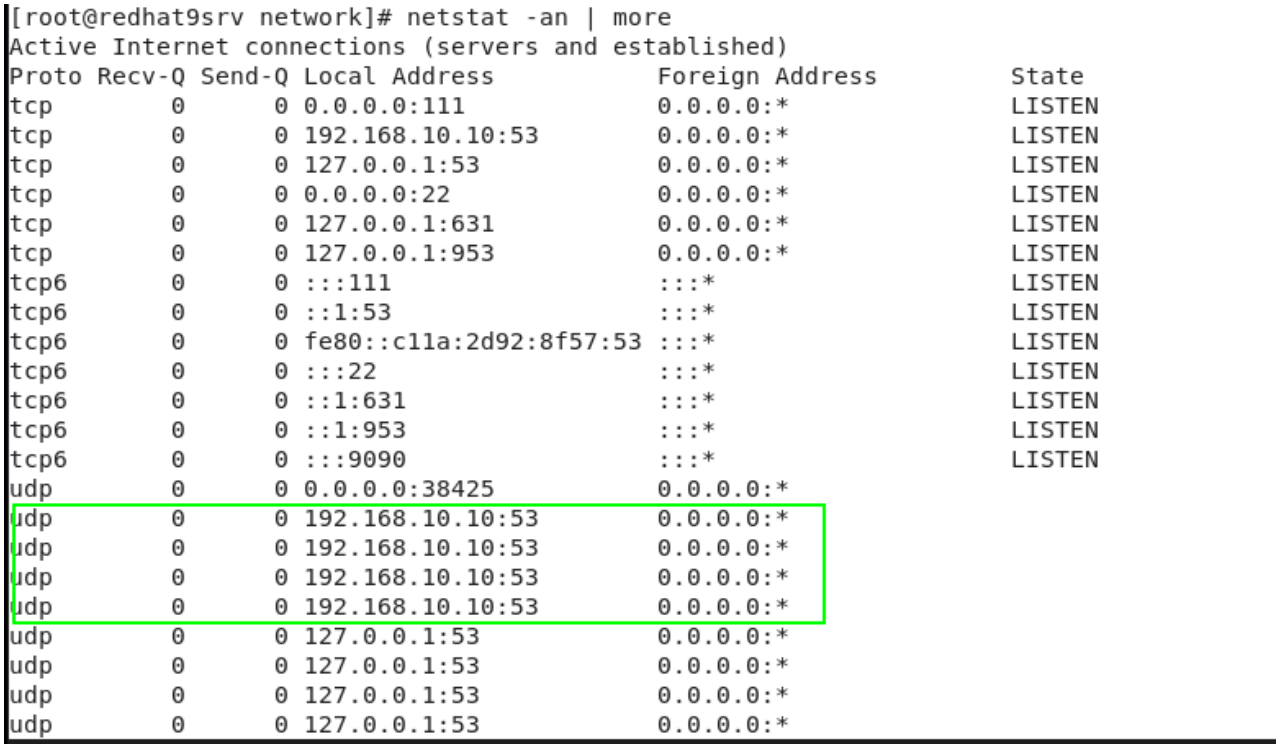
доменні імена може бути розділена на декілька серверів для забезпечення високої доступності та надійності.

DNS-сервер BIND підтримує рекурсивний та ітеративний режими розподілення запитів. Рекурсивний режим означає, що сервер спробує знайти відповідь на запит клієнта, навіть якщо це означає звернення до інших DNS-серверів. Ітеративний режим передбачає, що сервер надсилає клієнту список імен інших DNS-серверів, які можуть знайти відповідь.

BIND дозволяє адміністраторам конфігурувати зони, в яких зберігається інформація про доменні імена. Кожна зона може бути налаштована окремо з власними записами. Сервер має різні механізми безпеки, включаючи TSIG для автентифікації та DNSSEC для захисту від підробки DNS-відповідей. BIND підтримує журналювання подій та логування діяльності сервера. Це дозволяє адміністраторам відслідковувати проблеми та аналізувати роботу сервера. DNS сервер дозволяє внесення динамічних змін у DNS-записи, наприклад, при використанні DHCP для присвоєння IP-адрес комп'ютерам. BIND кешує DNS-запити, що дозволяє прискорити відповіді на повторні запити.

DNS сервер BIND запускається як сервіс у фоновому режимі, і його конфігураційний файл named.conf знаходяться в /etc/ каталозі та файли зон в /var/named/.

При перевірці командою netstat – an можна побачити що DNS сервер працює та очікує з’єднання на 53 UDP порт (Рисунок.2.7).



```
[root@redhat9srv network]# netstat -an | more
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 0.0.0.0:111             0.0.0.0:*               LISTEN
tcp        0      0 192.168.10.10:53        0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.1:53            0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:22              0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.1:631           0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.1:953           0.0.0.0:*               LISTEN
tcp6       0      0 :::111                  :::*                    LISTEN
tcp6       0      0 ::1:53                  :::*                    LISTEN
tcp6       0      0 fe80::c11a:2d92:8f57:53 :::*                    LISTEN
tcp6       0      0 :::22                   :::*                    LISTEN
tcp6       0      0 ::1:631                 :::*                    LISTEN
tcp6       0      0 ::1:953                 :::*                    LISTEN
tcp6       0      0 :::9090                 :::*                    LISTEN
udp        0      0 0.0.0.0:38425           0.0.0.0:*
udp        0      0 192.168.10.10:53        0.0.0.0:*
udp        0      0 192.168.10.10:53        0.0.0.0:*
udp        0      0 192.168.10.10:53        0.0.0.0:*
udp        0      0 192.168.10.10:53        0.0.0.0:*
udp        0      0 127.0.0.1:53            0.0.0.0:*
udp        0      0 127.0.0.1:53            0.0.0.0:*
udp        0      0 127.0.0.1:53            0.0.0.0:*
udp        0      0 127.0.0.1:53            0.0.0.0:*
```

Рисунок 2.7 – Вивід команди `netstat -an`

BIND є одним з найбільш популярним DNS-сервером. Він широко використовується для надання DNS-послуг в Інтернеті та корпоративних мережах.

2.3 Здійснення DNS flood атаки на DNS -сервер

На початковому етапі здійснимо перевірку працездатності DNS сервера з операційної системи Windows, яка розміщена в локальній мережі (див. рисунок 2.2).

На рисунку 2.8 показано мережеві налаштування операційної системи Windows.

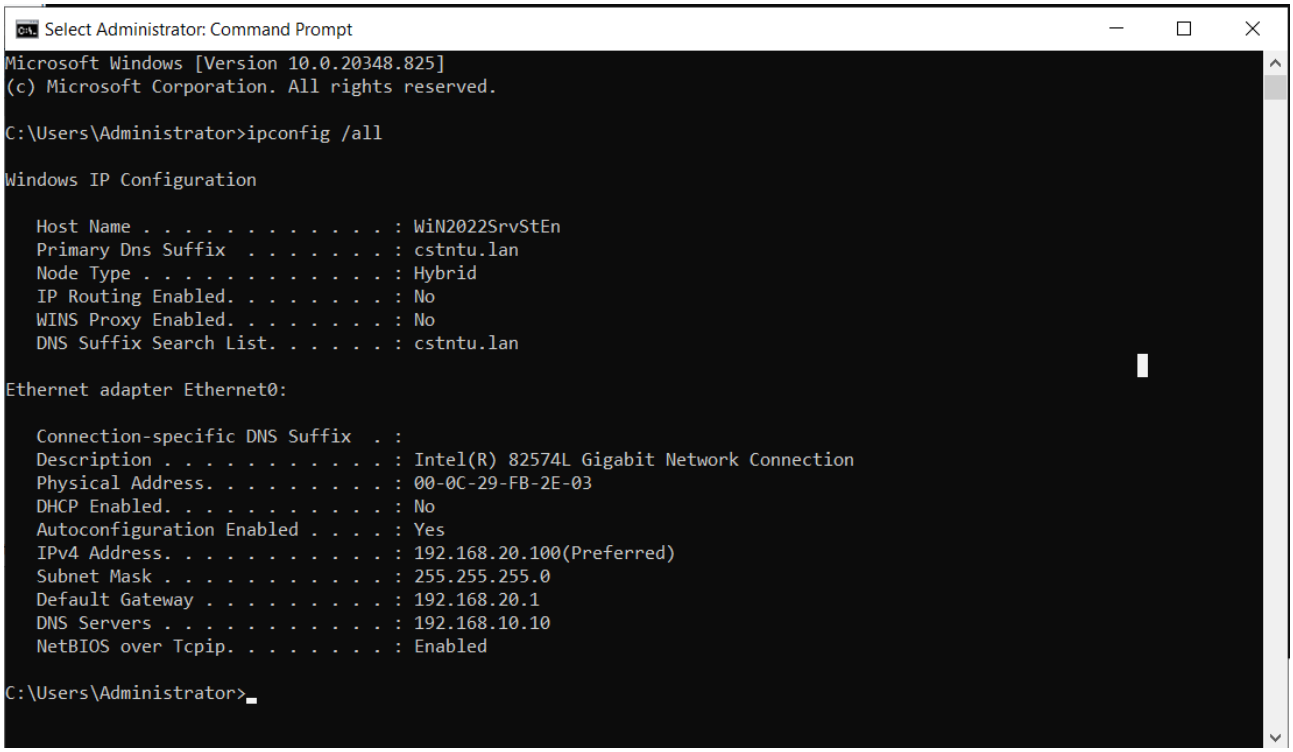


Рисунок 2.8 – Вивід команди `ipconfig /all`

На рисунку 2.9 показано перевірку доступності IP адреси DNS сервера та сервісу DNS.

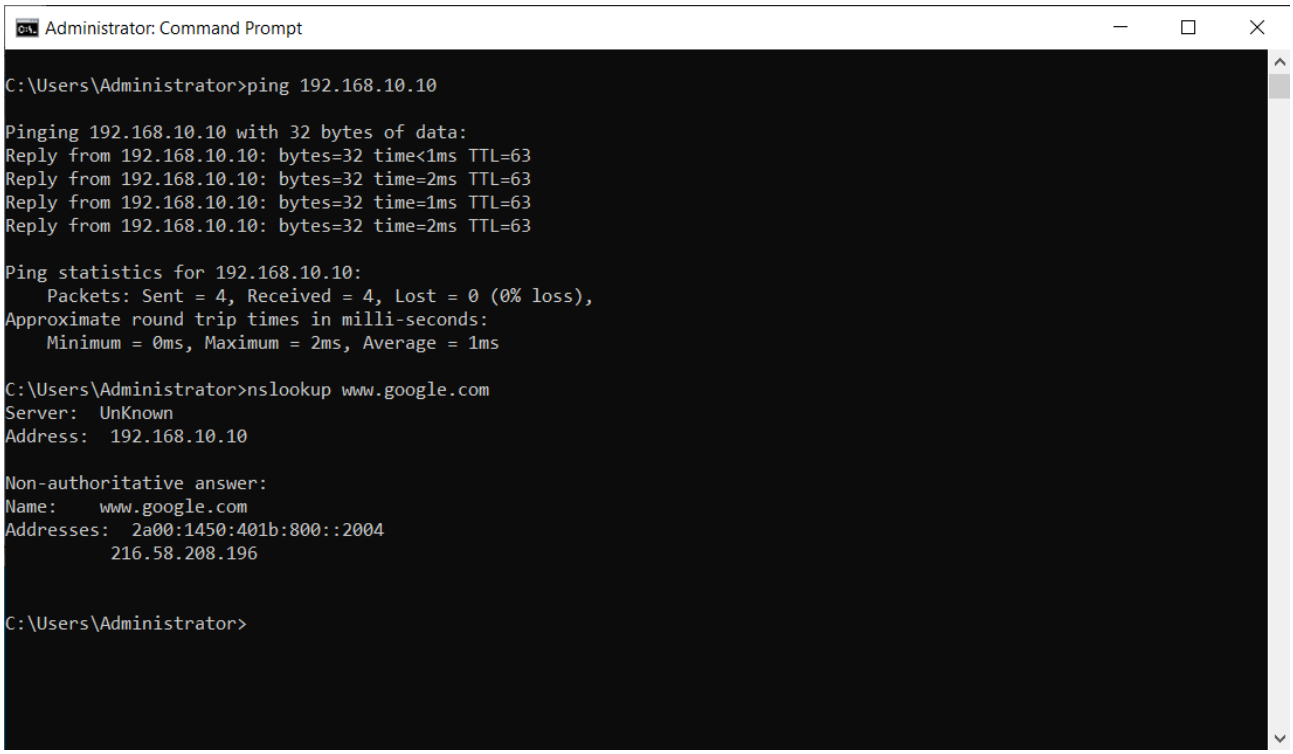


Рисунок 2.9 – Перевірка доступності IP адреси DNS сервера та сервісу DNS

Для показу методики проведення UDP flood використаємо операційну систему для тестування безпеки Parrot Security та інструмент hping3.

Відкриємо в Parrot Security термінали та ведемо наступну команду.

```
$sudo hping3 --flood -a 111.111.111.10 -2 -p 53 192.168.10.10
```

Команда hping3 запускає UDP flood атаку на сервер з IP-адресою 192.168.10.10 на порт 53 (DNS). Давайте розглянемо параметри цієї команди:

1) `--flood`. Цей параметр вказує hping3 надсилати UDP-пакети на цільовий сервер з максимальною швидкістю. Це робить атаку надзвичайно інтенсивною і може завдати значної шкоди мережі;

2) `-a 111.111.111.10`. Цей параметр використовується для підробки адреси джерела (spoof source address), що дозволяє здійснити атаку з іншої IP-адреси, вказуючи, що IP-адреса джерела пакетів є 111.111.111.10;

3) `-2`. Цей параметр вказує використовувати протокол UDP;

4) `-p 53`. Цей параметр вказує порт 53, що відповідає DNS-запитам;

5) `192.168.10.10`. Це цільова IP-адреса для атаки, в даному випадку, DNS сервера з IP-адресою 192.168.10.10.

Ця команда генерує велику кількість UDP-пакетів і надсилає їх на порт 53 UDP цільового сервера з вказаною IP-адресою джерела 111.111.111.10. Для збільшення інтенсивності атаки відкриємо ще два термінали в Parrot Security та використавши підроблені IP адреси джерела 111.111.111.11 та 111.111.111.12 запустимо додаткових дві атаки (Рисунок 2.10).

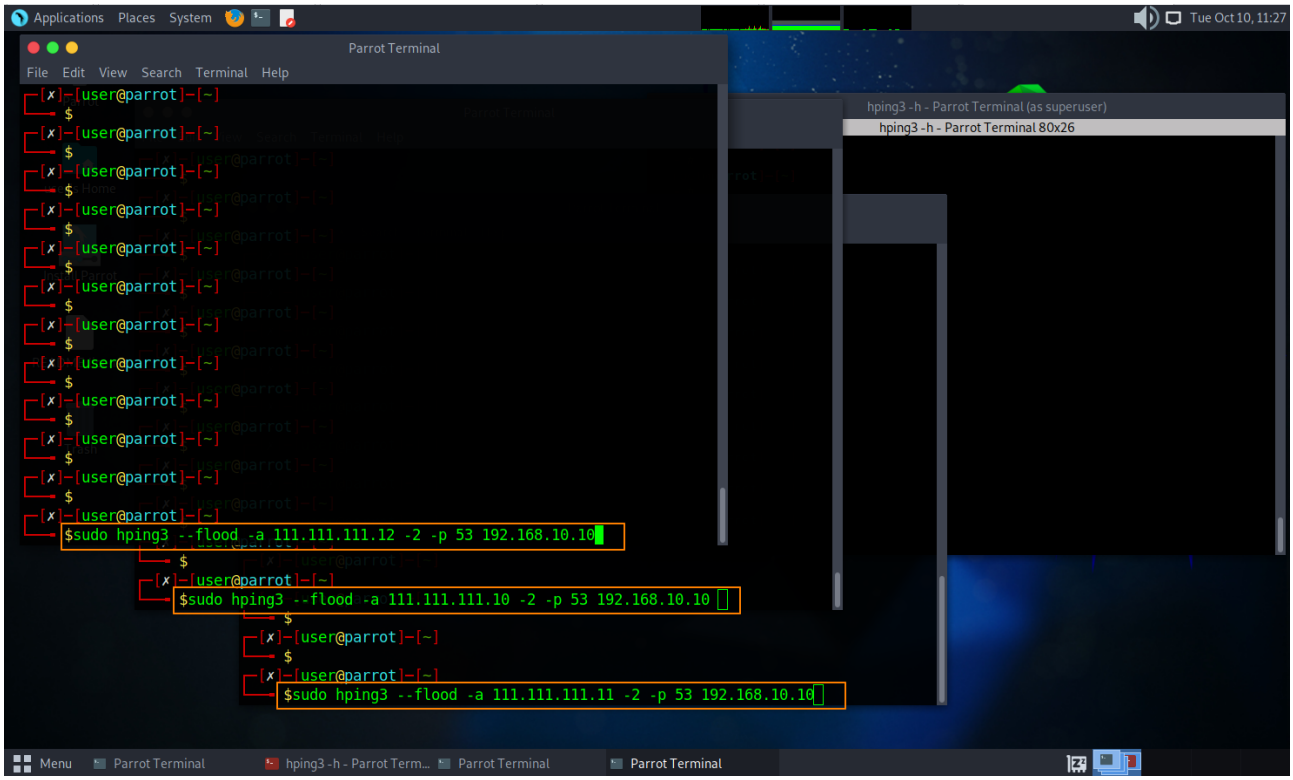


Рисунок 2.10 – Здійснення UDP flood атаки на DNS сервер

Вивід команди `top` (Рисунок 2.11) надає інформацію про поточну роботу системи та використання ресурсів.

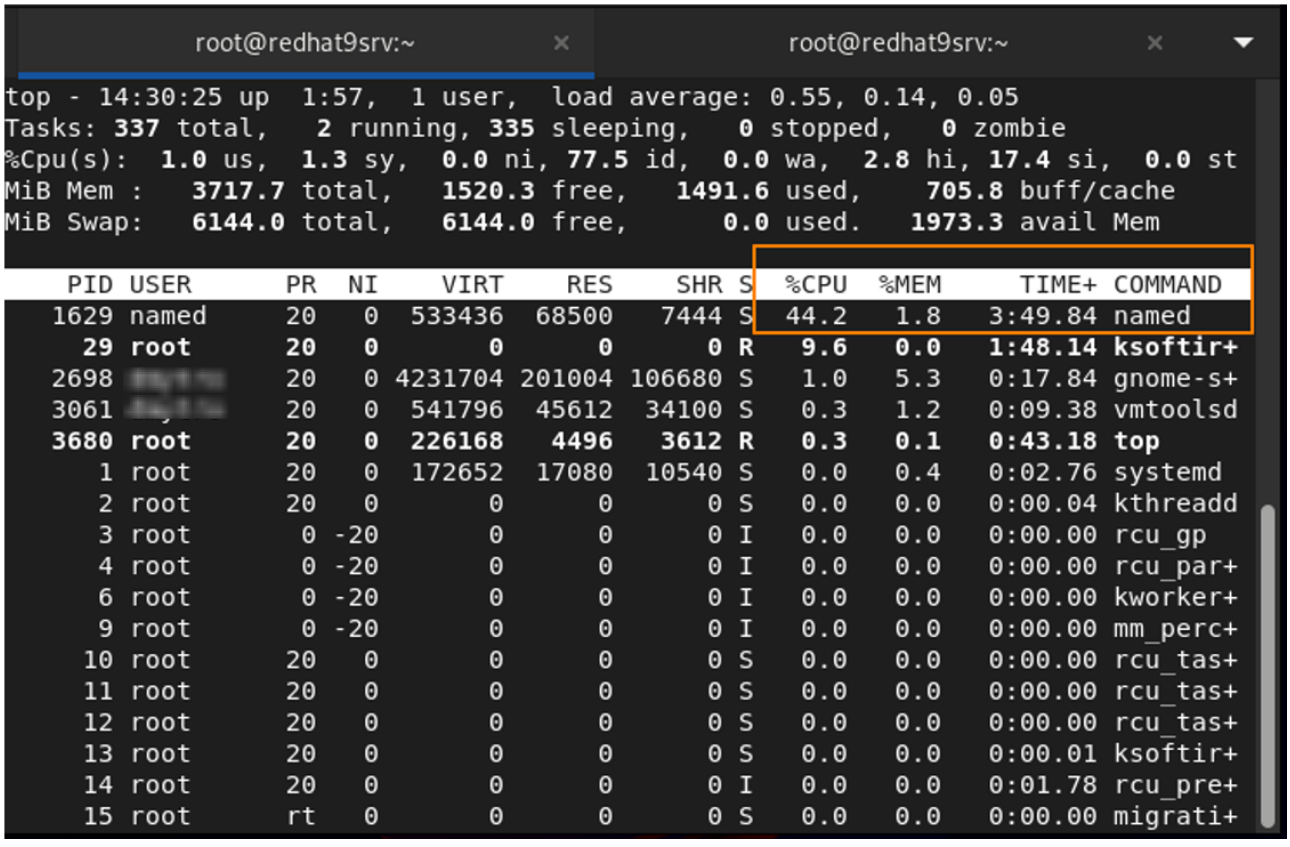


Рисунок 2.11 – Вивід команди `top`

Відсоток використання процесора процесом named (PID 1629) становить 42.2%, що означає, що процес використовує близько половини потужності процесора. Це є наслідком атаки на UDP порт 53.

Для моніторингу трафіку в операційній системі Red Hat в реальному часі використаємо утиліту iptraf-ng. Вона надає інформацію про обсяги та напрямки мережевого трафіку в системі і дозволяє аналізувати різні аспекти мережевої активності. iptraf-ng може показувати в реальному часі обсяги передачі та отримання даних через мережеві інтерфейси системи. Можна відстежувати як загальний обсяг трафіку, так і окремі пакети за допомогою різних фільтрів.

Утиліта надає інформацію про обсяг трафіку за різними мережевими протоколами, такими як TCP, UDP, ICMP і т. д. iptraf-ng дозволяє відстежувати широкомовний трафік в мережі, що може бути корисним при роботі з мультимедійними інформаційними потоками. Можна переглядати інформацію про активні мережеві з'єднання, включаючи джерело та призначення, порти та статуси. Утиліта надає графічний інтерфейс, який полегшує відстеження та аналіз мережевого трафіку.

На рисунку 2.12 можна побачити що за проміжок часу до 60 секунд (Time: 0:00) є велика кількість вхідних пакетів UDP, але вихідні пакети UDP відсутні.

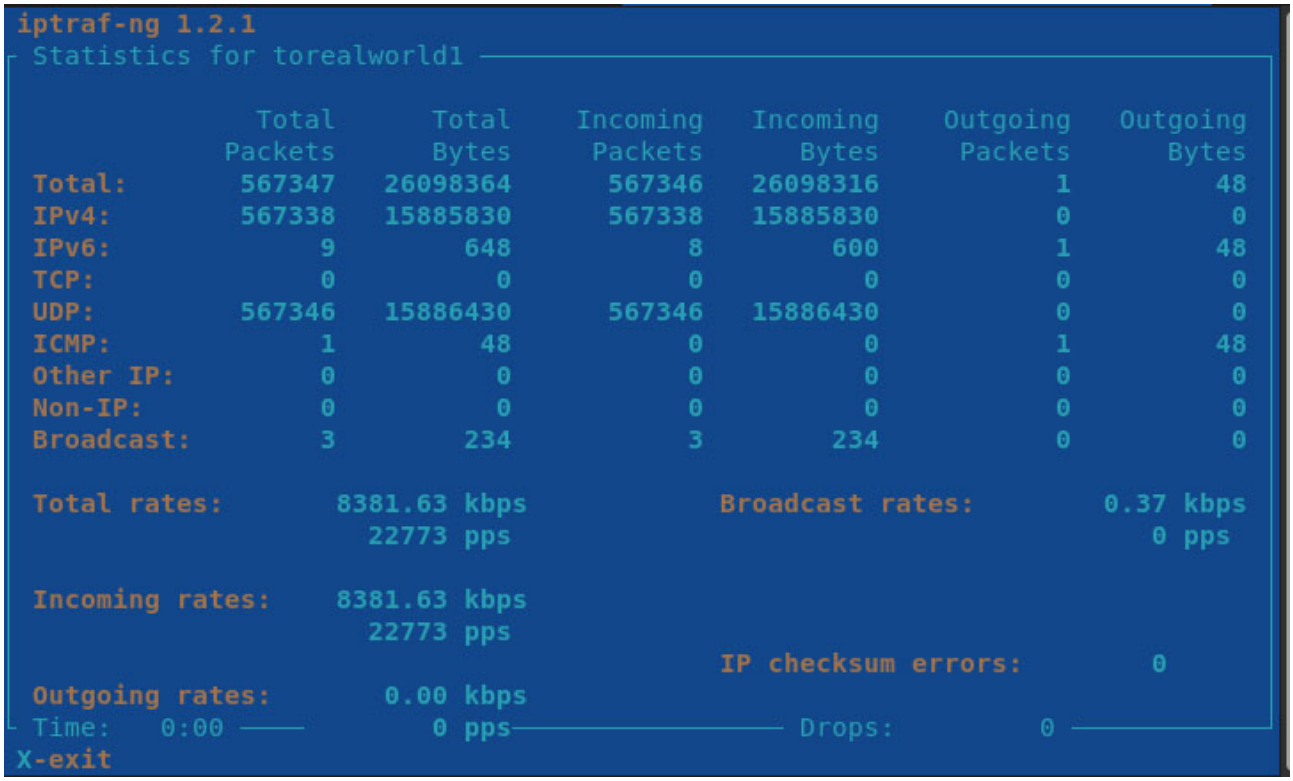


Рисунок 2.12 – Вивід загальної статистики команди `iptraf-ng`

На рисунку 2.13 можна побачити що за проміжок часу до 60 секунд (Time: 0:00) є велика кількість вхідних пакетів UDP на порт 53, але вихідні пакети UDP відсутні.

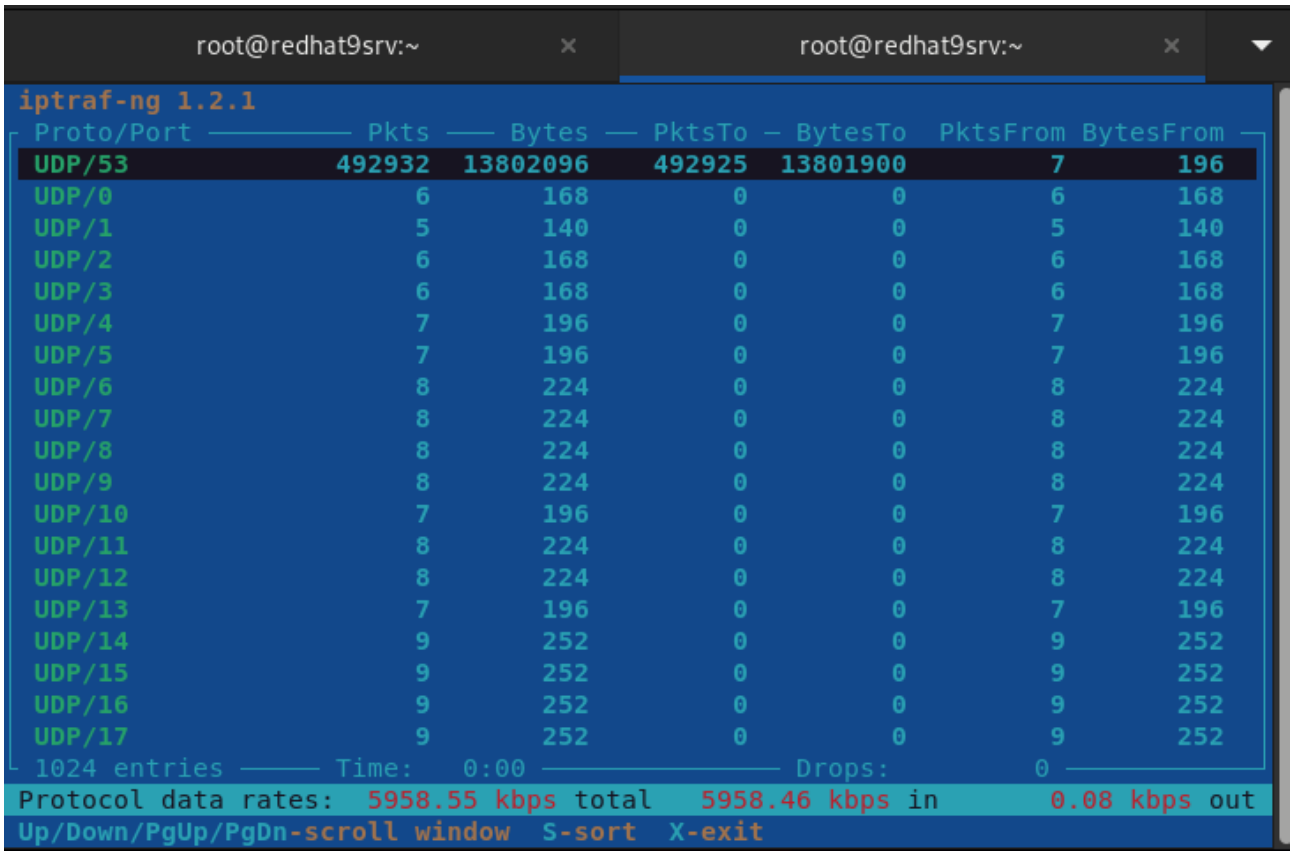


Рисунок 2.13 – Вивід статистики по портах команди `iptraf-ng`

Все це свідчить про те що здійснюється UDP flood атака.

Моніторинг атаки в реальному часі можна також здійснити за допомогою `iptraf-ng`. На рисунку 2.14 та 2.15 можна побачити що за проміжок часу до 60 секунд (Time: 0:00) є лише вхідний UDP трафік на порт 53 з однаковим розміром пакету 46 байт.

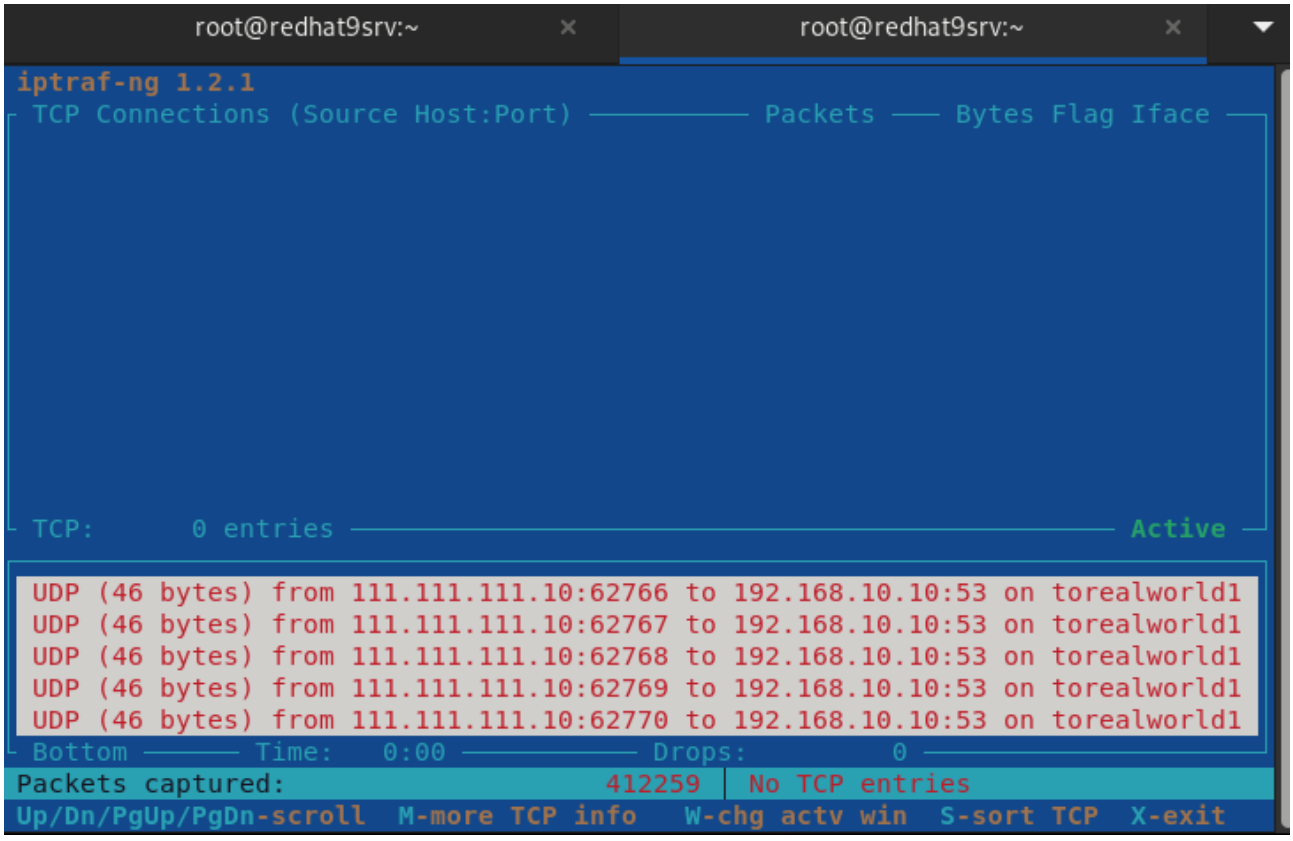


Рисунок 2.14 –Моніторинг атаки в реальному часі

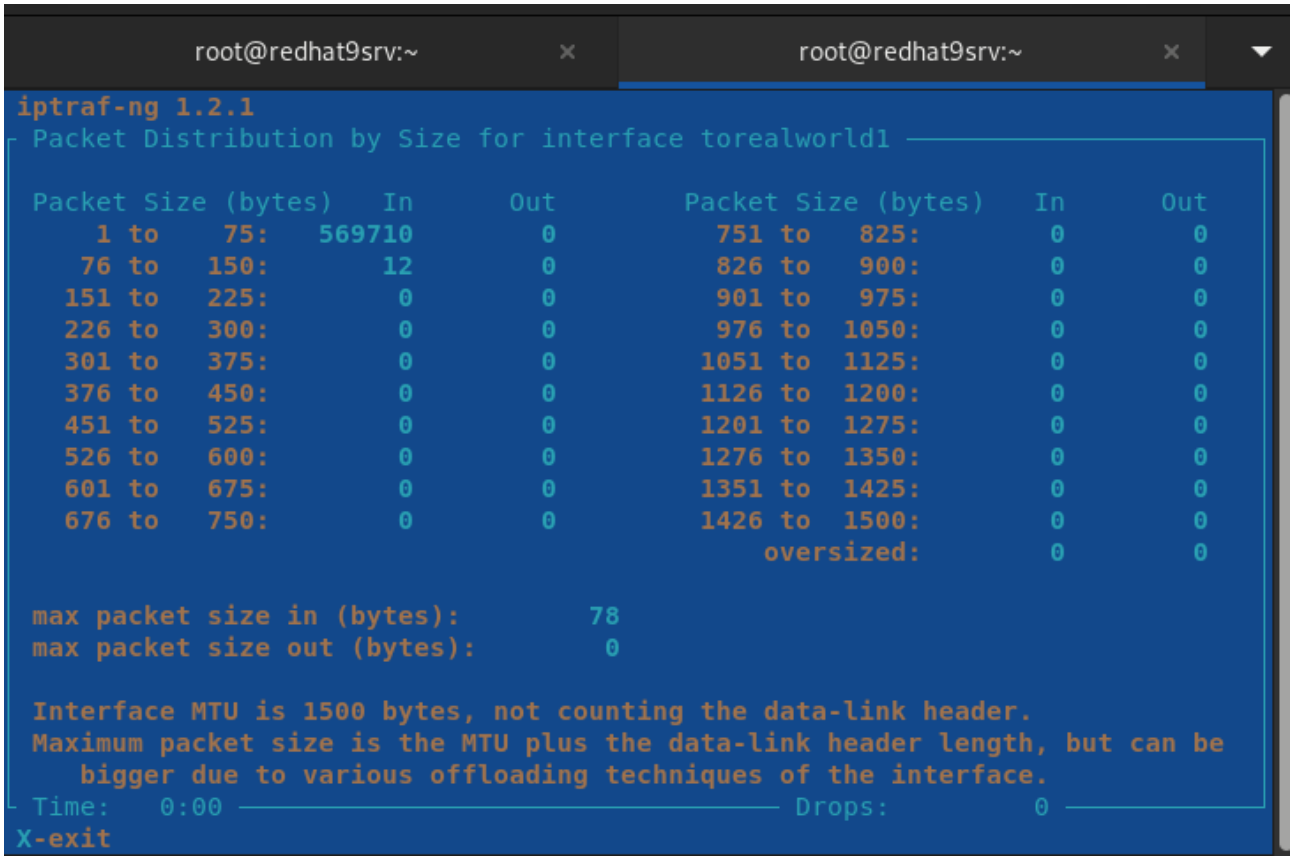


Рисунок 2.15 –Статистика про розміру пакетів

При даній інтенсивності атаки DNS сервіс перестає відповідати на запити дійсних користувачів. В кращому варіанті відповіді будуть приходити зі значною затримкою. В чому можна переконатись виконавши команду `nslookup` з операційної системи Windows, яка знаходиться в локальній мережі. На рисунках 2.16-2.18 показано вивід команди nslookup під час UDP flood атаки.

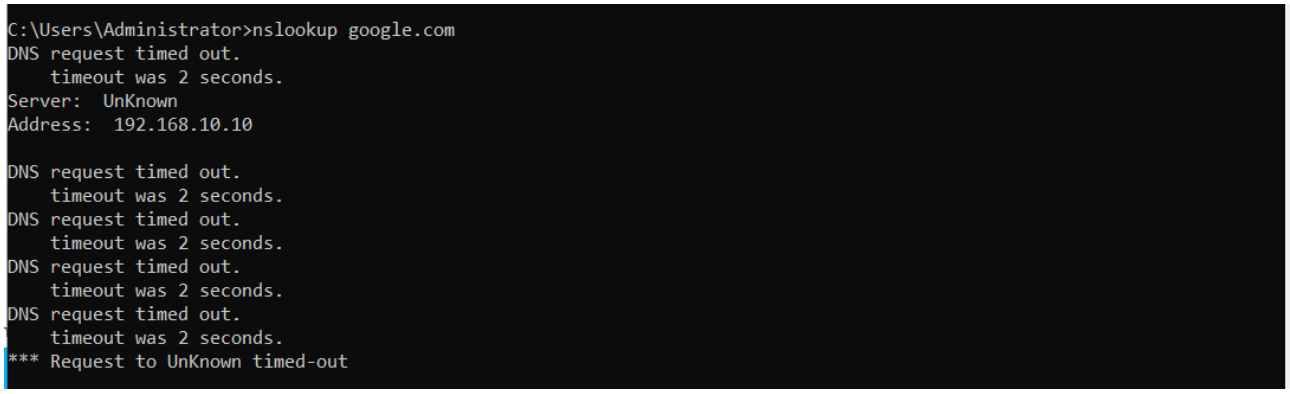


Рисунок 2.16 – Вивід команди `nslookup google.com`

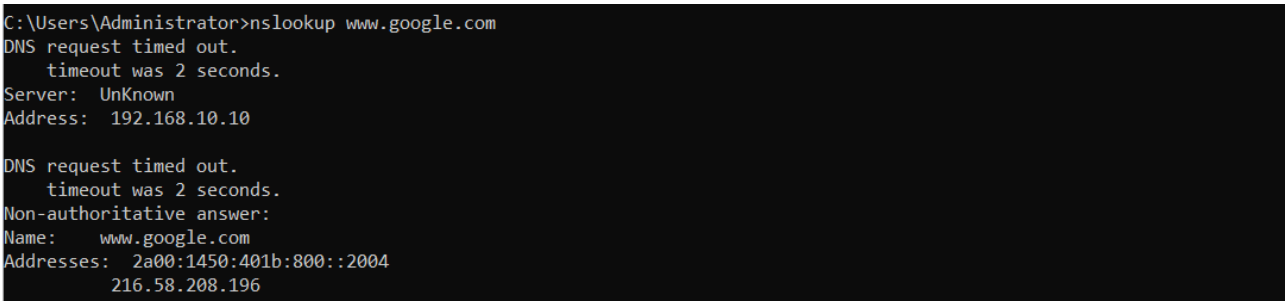


Рисунок 2.17 – Повторення команди `nslookup google.com`

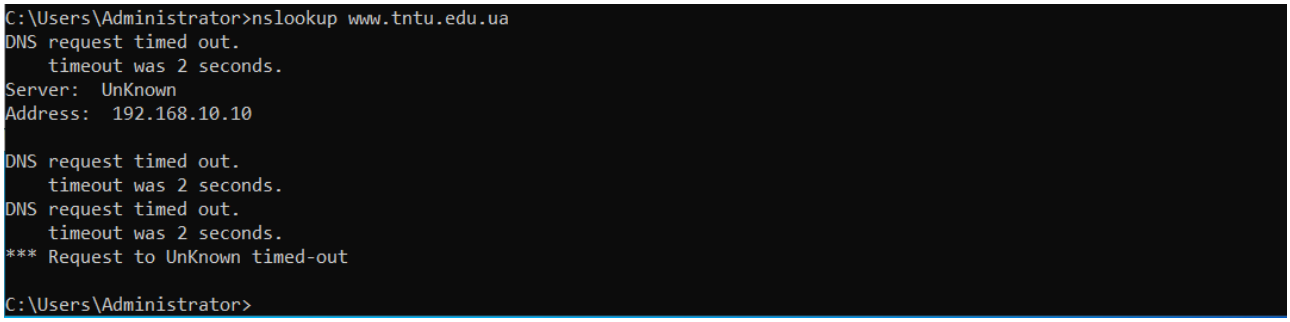


Рисунок 2.18 – Вивід команди `nslookup aol.com`

Дані виводи показують що атака пройшла успішно. DNS сервіс працює не коректно і не відповідає на запити користувачів локальної мережі.

На рисунку 2.19 показано статистику виконання атаки на DNS сервіс.

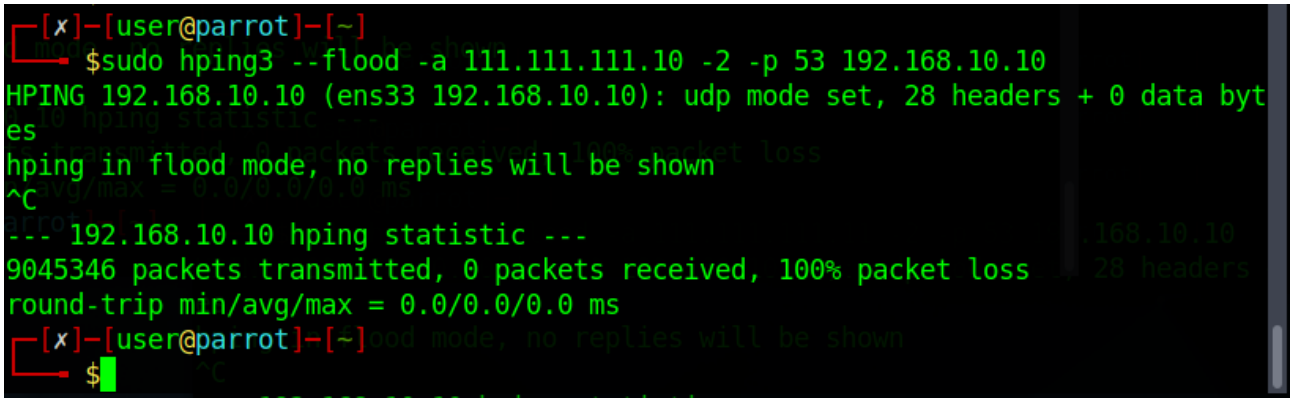


Рисунок 2.19 – Статистика виконання атаки на DNS сервіс

Цей вивід команди hping3 показує, що була виконана UDP flood атака на IP-адресу 192.168.10.10 на порті 53.

Ось розшифровка виводу:

1) `HPING 192.168.10.10 (ens33 192.168.10.10): udp mode set, 28 headers + 0 data bytes.` hping3 налаштований на виконання UDP flood атаки

на IP-адресу 192.168.10.10 через інтерфейс ens33, додано 28 байтів заголовка UDP, але без даних;

2) `hping in flood mode, no replies will be shown` - hping3 вказує, що вона працює в режимі flood (затоплення), і відповіді не будуть показані;

3) `--- 192.168.10.10 hping statistic ---.` Це роздільник, який вказує на початок статистики від hping3;

4) `9045346 packets transmitted, 0 packets received, 100% packet loss.` Після атаки вказано кількість переданих пакетів (9045346), кількість отриманих пакетів (0) і відсоток втрати пакетів (100%). У цьому випадку всі пакети були втрачені.

5) `round-trip min/avg/max = 0.0/0.0/0.0 ms.` Ця частина показує статистику щодо часу затримки. У цьому випадку час затримки дорівнює нулю, оскільки атака flood означає надсилання пакетів без очікування відповідей.

3 РОЗРОБКА ПРОГРАМНОГО МОДУЛЯ ЗАХИСТУ ВІД UDP FLOOD АТАК НА DNS СЕРВЕР

3.1 Розробка програмного модуля

Розробка програмного модуля для захисту від UDP flood атак включає в себе ряд кроків та заходів з метою виявлення та обмеження таких атак. Модуль повинен моніторити вхідний мережевий трафік на маршрутизаторі. Для цього буде використано інструменти, такі як брандмауер PF.

Модуль повинен виявляти аномальний трафік на UDP порт 53, який може вказувати на потенційну flood атаку. Це може бути визначено за допомогою порівняння обсягів трафіку зі певними параметрами.

Якщо виявлена атака, модуль повинен за допомогою брандмауера PF заблокувати трафік від вказаних джерел атаки та повідомити про UDP flood атаку електронною поштою адміністраторів системи.

3.1.1 Створення алгоритму виявлення та блокування UDP flood атаки

Алгоритм виявлення та блокування UDP flood атаки за допомогою PF буде реалізований наступним чином:

1) Створення правила PF. До конфігураційного файлу pf.conf добавимо правило для відслідковування з’єднань до IP-адресі нашого DNS сервера (192.168.10.10) і 53 порту UDP.

2) Аналіз станів PF. Використаємо команду /sbin/pfctl -s state для отримання списку станів, які обробляє PF. У нашому випадку, ми будемо шукати стани, які відповідають IP-адресі нашого DNS сервера (192.168.10.10) і порту 53.

3) Виявлення атаки. Перевіряємо кількість станів зі статусом "NO_TRAFFIC:SINGLE". Цей стан вказує на те, що дане з'єднання має стан "NO_TRAFFIC " і в даний момент не передає жодного трафіку. "SINGLE" вказує на те, що дане з'єднання має одиничний (індивідуальний) стан, тобто це окреме з'єднання, яке не має спільних атрибутів з іншими з'єднаннями.

Це означає, що брандмауер встановив стан для цього з'єднання, але відсутній обмін даними між вихідним і вхідним IP-адресами та портами.

4) Створення PF таблиці. Створимо в конфігураційному файлі pf.conf спеціальну PF таблицю для зберігання IP-адрес, які беруть участь у атаці.

5) Блокування трафіку з таблиці. Після того, як створили таблицю блокування напишемо правило блокування IP адрес з таблиці.

5) Додавання IP адрес до PF таблиці. Якщо виявили атаку то використовуємо команду pfctl для додавання IP атакуючого до таблиці PF.

6) Логування та сповіщення. Важливо логувати всі виявлені атаки та вживати заходів щодо сповіщення адміністратора про можливі атаки. Розробимо систему сповіщення на електронну пошту адміністратору про UDP flood атаку на DNS сервер.

3.1.2 Налаштування брандмауера PF

Для можливості виконання алгоритму описаного в пункті 3.1.1 зробимо зміни в конфігураційному файлі pf.conf. Добавимо наступні конфігураційні рядки:

1) `table <udpflooddns> persist`. Визначає PF-таблицю з назвою `<udpflooddns>` та вказує, що вона повинна бути постійною (може не містити даних).

2) `pass in on $ext_if proto {udp } from any to 192.168.10.10 port {53}`. Дозволяє UDP пакети з будь-якого джерела до IP-адреси 192.168.10.10 на порту 53 (DNS).

3) `block drop in quick on $ext_if from <udpflooddns> to any`. Блокує і відкидає всі пакети, які надходять з IP-адрес, які знаходяться в таблиці `<udpflooddns>`.

На рисунку 3.1 показано вмісти конфігураційного файлу pf.conf з внесеними змінами.

```
ext_if = "em0"
dmz_if = "em1"
int_if = "em2"
dmznet = $dmz_if:network
localnet = $int_if:network
#
set skip on lo0
set limit { states 4000000, frags 4000000, src-nodes 400000 }
set optimization aggressive
#
scrub in all
#
nat on $ext_if from $localnet to any -> ($ext_if)
#
block all
#VPN
pass in inet proto udp to $ext_if port { 1701,500,4500 }
pass in inet proto tcp to $ext_if port { 1723 } keep state
#
pass from { lo0, $localnet, $dmznet } to any keep state
#
block in log quick on $ext_if from { 127.0.0.0/8, 192.168.0.0/16, 172.16.0.0/12,
 10.0.0.0/8, 169.254.0.0/16, 192.0.2.0/24, 0.0.0.0/8, 240.0.0.0/4 }
#FOR UDPflood DNS
table <udpflooddns> persist
pass in  on $ext_if proto {udp} from any to 192.168.10.10 port {53}
block drop in quick on $ext_if from <udpflooddns> to any
```

Рисунок 3.1 – Конфігураційний файл pf.conf

Цей файл pf.conf налаштовує брандмауер PF для захисту мережі від різних видів трафіку та включає заходи безпеки для захисту від UDP flood атак на DNS сервер.

3.1.3 Вибір середовища розробки

Для написання програмного модуля автоматичного блокування IP-адрес зловмисників буде використано мову програмування Python [11].

Python - це високорівнева мова програмування, яка відома своєю лаконічністю і простотою синтаксису, що робить її відмінним вибором для початківців і досвідчених розробників. Високий рівень абстракції дозволяє виразно виражати ідеї в коді, що полегшує розробку та зрозумілість програмного коду. Python має велику кількість вбудованих бібліотек та модулів для різних завдань, включаючи роботу з мережами, базами даних, обробку тексту, наукове обчислення і багато інших.

Код, написаний на Python, може бути запущений на різних операційних системах без змін. Python підтримується на багатьох платформах, включаючи

Unix, Linux, Windows і macOS. Python, завдяки об'єктно-орієнтованому підходу до програмування, дозволяє створювати модульні програми. Також дозволяє легко працювати з вебсервісами та API.. Python розповсюджується під ліцензією Python Software Foundation License. Це дає можливість використовувати мову в різних проектах безкоштовно.

3.1.4 Написання програмного модуля

Програмний модуль написаний на мові Python і призначений для виявлення та реагування на UDP flood атаки на DNS сервер. Лістинг даного сценарію наведено в додатку Б.

Давайте розглянемо, як відбувається процес виявлення та блокування UDP flood:

1) Визначення поточного часу. Скрипт спочатку отримує поточний час та дату та формує тему для повідомлення електронної пошти, що включає ім'я хоста (router.cstntu.lan) і поточний час.

2) Виконання команди pfctl. Скрипт виконує команду /sbin/pfctl -s state | grep 192.168.10.10:53, щоб отримати стан брандмауера PF, який містить інформацію про підключення до порта 53 DNS сервера (192.168.10.10). Результат цієї команди зберігається у змінній output.

3) Аналіз результатів. Результат виконаної команди розбивається на окремі рядки, і скрипт аналізує кожен рядок для пошуку IP-адрес, які можуть бути пов'язані із UDP flood атакою. Кожен рядок, в якому зустрічається фраза "NO_TRAFFIC:SINGLE", інтерпретується як можлива IP-адреса, що виконує атаку. Знайдені IP-адреси додаються до списку udpflood_ips.

4) Перевірка кількості IP-адрес. Скрипт перевіряє кількість знайдених IP-адрес. Якщо не виявлено атак, виводиться повідомлення "No UDP DNS flood attacks detected."

5) Обробка атак з більш ніж 100 IP-адресами. Якщо кількість виявлених IP-адрес з станом "NO_TRAFFIC:SINGLE" перевищує 100, то це вважається UDP flood атакою. У цьому випадку, скрипт видаляє дубльовані IP-адреси зі списку і записує оригінальні у файл /root/udpflooddns/udpflooddnsip. Потім скрипт

виконує команду pfctl -t udpflooddns -T add -f /root/udpflooddns/udpflooddnsip, щоб додати ці IP-адреси до таблиці PF udpflooddns.

6) Відправка повідомлення по електронній пошті. Скрипт генерує повідомлення, яке містить кількість виявлених IP-адрес і список унікальних IP-адрес, а також інформацію про дату і час виявлення атаки. Потім він використовує бібліотеку smtplib для відправки цього повідомлення на вказану адресу електронної пошти.

7) Обробка винятків. Скрипт також включає обробку можливих винятків, таких як помилки виконання команди PF або відсутність атак.

Цей скрипт дозволяє виявляти та реагувати на масові UDP flood атаки на DNS сервер за допомогою інтеграції зі системою PF брандмауера та відправки сповіщень по електронній пошті.

3.1.5 Автоматизація процесу відслідковування атаки та блокування IP зловмисника

Для автоматичного запуску відслідковування атаки та блокування IP зловмисника використаємо cron. Cron - це програмне забезпечення для планування виконання завдань на операційній системі Unix і подібних до Unix. Його можна використовувати для автоматизації різних завдань, таких як виконання скриптів, відправлення звітів по електронній пошті, очищення тимчасових файлів тощо.

Добавимо наступні рядки у файл crontab:

```
* * * * * sleep 30 &&  python3
 /root/udpflooddns/udpflood_monitor.py
* * * * * python3 /root/udpflooddns/udpflood_monitor.py
```

Рядок означає наступне:

1) `* * * * *`. Запускати кожну хвилину.

2) `sleep 30`. Зачекати 30 секунд перед запуском скрипта.

Тепер скрипт буде автоматично запускатися кожні 30 секунд та здійснювати перевірку на UDP flood атаку.

3.2 Тестування програмного модуля

Для тестування працездатності розробленого програмного модуля захисту від UDP flood атак здійснимо тестову атаку ідентичну як в пункті 2.3 (Рисунок 3.2).

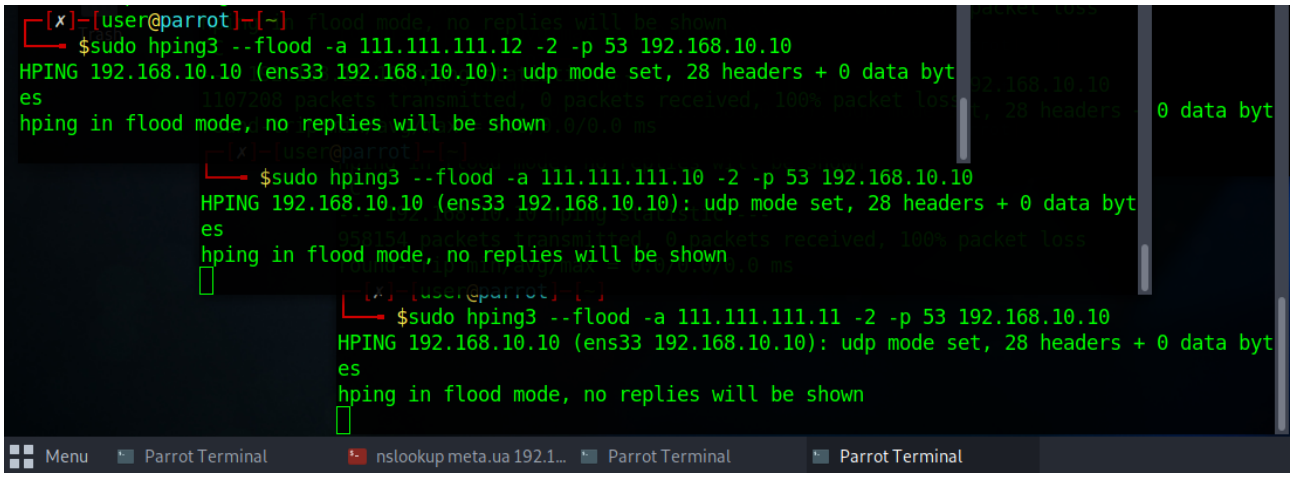


Рисунок 3.2 – Здійснення тестової UDP flood атаки

На першому етапі тестування відключимо автоматичне відслідковування атаки та проаналізуємо поведінку маршрутизатора на базі OpenBSD під час атаки.

На рисунку 3.3 можна побачити що маршрутизатор з новою конфігурацією PF коректно відслідковує з’єднання за станом "NO_TRAFFIC:SINGLE".

```
all udp 192.168.10.10:53 <- 111.111.111.12:30754       NO_TRAFFIC:SINGLE
all udp 192.168.10.10:53 <- 111.111.111.11:29084       NO_TRAFFIC:SINGLE
all udp 192.168.10.10:53 <- 111.111.111.12:30755       NO_TRAFFIC:SINGLE
all udp 192.168.10.10:53 <- 111.111.111.11:29085       NO_TRAFFIC:SINGLE
all udp 192.168.10.10:53 <- 111.111.111.12:30756       NO_TRAFFIC:SINGLE
all udp 192.168.10.10:53 <- 111.111.111.11:29086       NO_TRAFFIC:SINGLE
all udp 192.168.10.10:53 <- 111.111.111.12:30757       NO_TRAFFIC:SINGLE
all udp 192.168.10.10:53 <- 111.111.111.11:29087       NO_TRAFFIC:SINGLE
all udp 192.168.10.10:53 <- 111.111.111.12:30758       NO_TRAFFIC:SINGLE
all udp 192.168.10.10:53 <- 111.111.111.11:29088       NO_TRAFFIC:SINGLE
all udp 192.168.10.10:53 <- 111.111.111.12:30759       NO_TRAFFIC:SINGLE
all udp 192.168.10.10:53 <- 111.111.111.11:29089       NO_TRAFFIC:SINGLE
all udp 192.168.10.10:53 <- 111.111.111.12:30760       NO_TRAFFIC:SINGLE
all udp 192.168.10.10:53 <- 111.111.111.11:29090       NO_TRAFFIC:SINGLE
all udp 192.168.10.10:53 <- 111.111.111.12:30761       NO_TRAFFIC:SINGLE
all udp 192.168.10.10:53 <- 111.111.111.11:29091       NO_TRAFFIC:SINGLE
all udp 192.168.10.10:53 <- 111.111.111.12:30762       NO_TRAFFIC:SINGLE
all udp 192.168.10.10:53 <- 111.111.111.11:29092       NO_TRAFFIC:SINGLE
all udp 192.168.10.10:53 <- 111.111.111.12:30763       NO_TRAFFIC:SINGLE
all udp 192.168.10.10:53 <- 111.111.111.11:29093       NO_TRAFFIC:SINGLE
all udp 192.168.10.10:53 <- 111.111.111.12:30764       NO_TRAFFIC:SINGLE
```

Рисунок 3.3 – Вивід команди `/sbin/pfctl -s state`

Оскільки система виявлення та блокування атаки відключена то відсоток використання процесора процесом named (PID 4527) становить 61.8%, що означає, що процес використовує більше половини потужності процесора (Рисунок 3.4). Це є наслідком атаки на UDP порт 53.

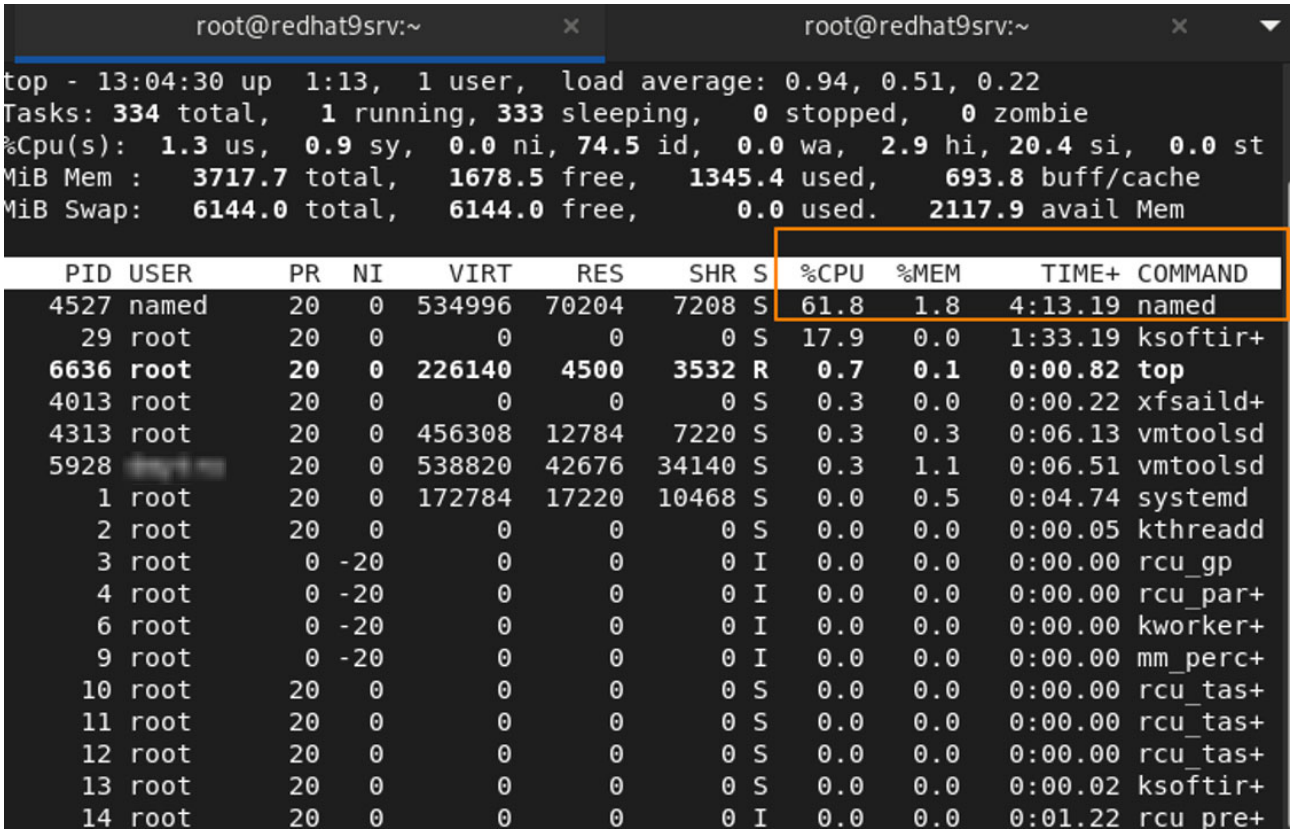


Рисунок 3.4 – Вивід команди `top` на DNS сервері

При даній інтенсивності атаки DNS сервіс знову перестає відповідати на запити користувачів. Що підтверджується виводом команди nslookup в операційної системи Windows, яка знаходиться в локальній мережі (Рисунок 3.5).

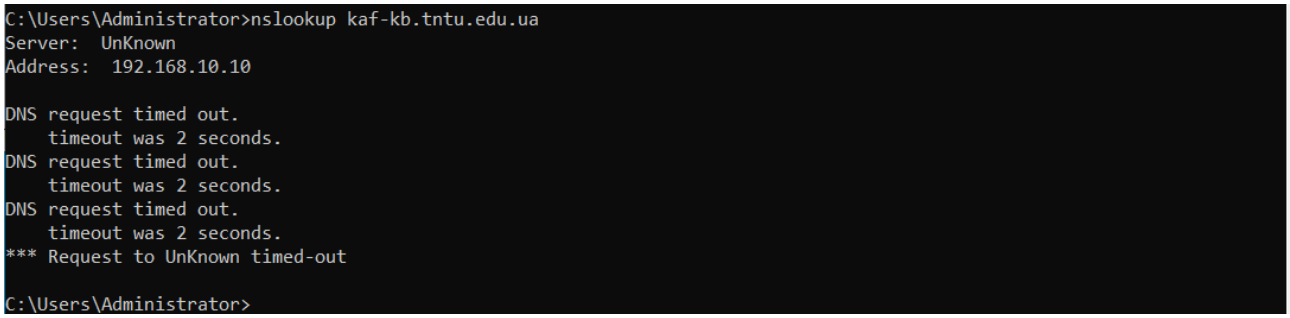


Рисунок 3.5 – Вивід команди `nslookup kaf-kb.tntu.edu.ua`

Ввімкнемо автоматичне відслідковування та блокування UDP flood атаки. Після ввімкнення даної системи протягом декількох секунд ми отримуємо лист з повідомлення про атаку (Рисунок 3.6).

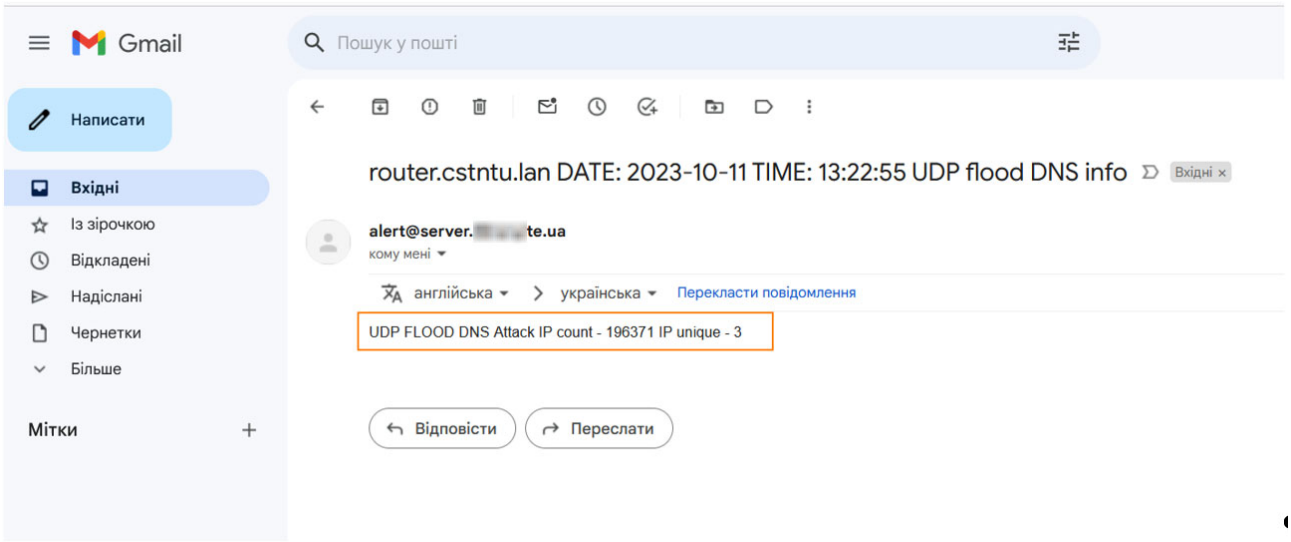


Рисунок 3.6 – Лист з повідомленням про UDP flood атаку

В тілі листа можна побачити що було виявлено одночасних 196371 з’єднання за станом "NO_TRAFFIC:SINGLE" на порт UPD 53 і ці з’єднання відбулись з 3 унікальних IP. В нашому випадку це 111.111.111.10, 111.111.111.11 та 111.111.111.12.

Після блокування IP атакуючих вивід команди `top` на DNS сервері показує що система вернулась до нормального функціонування (Рисунок 3.7).

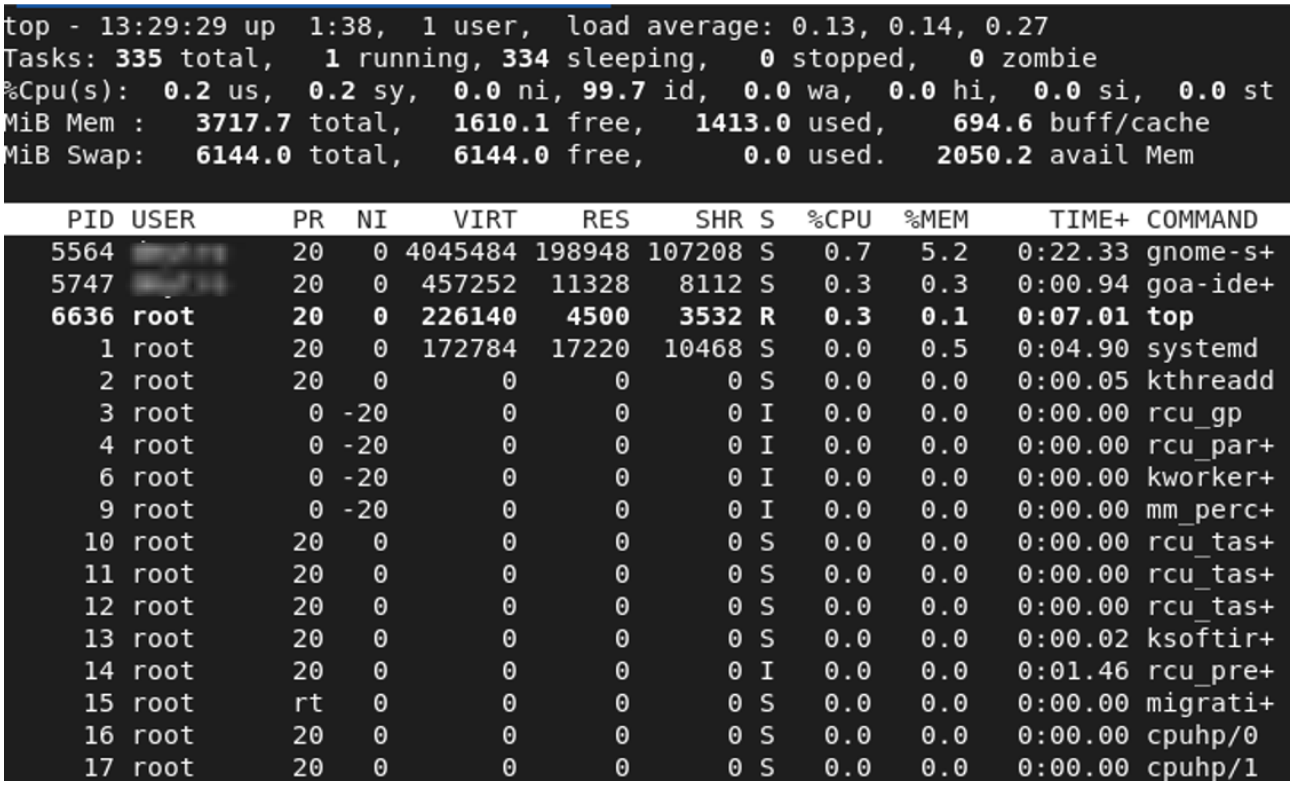


Рисунок 3.7 – Вивід команди `top` на DNS сервері після блокування атакуючих IP адрес

Також можна переконатись що після запуску системи відслідковування та блокування UDP flood атаки наш DNS сервіс знову почав відповідати на запити користувачів. Що підтверджується повторним виводом команди nslookup в операційної системи Windows, яка знаходиться в локальній мережі та завантаженням тестового сайту (Рисунок 3.8).

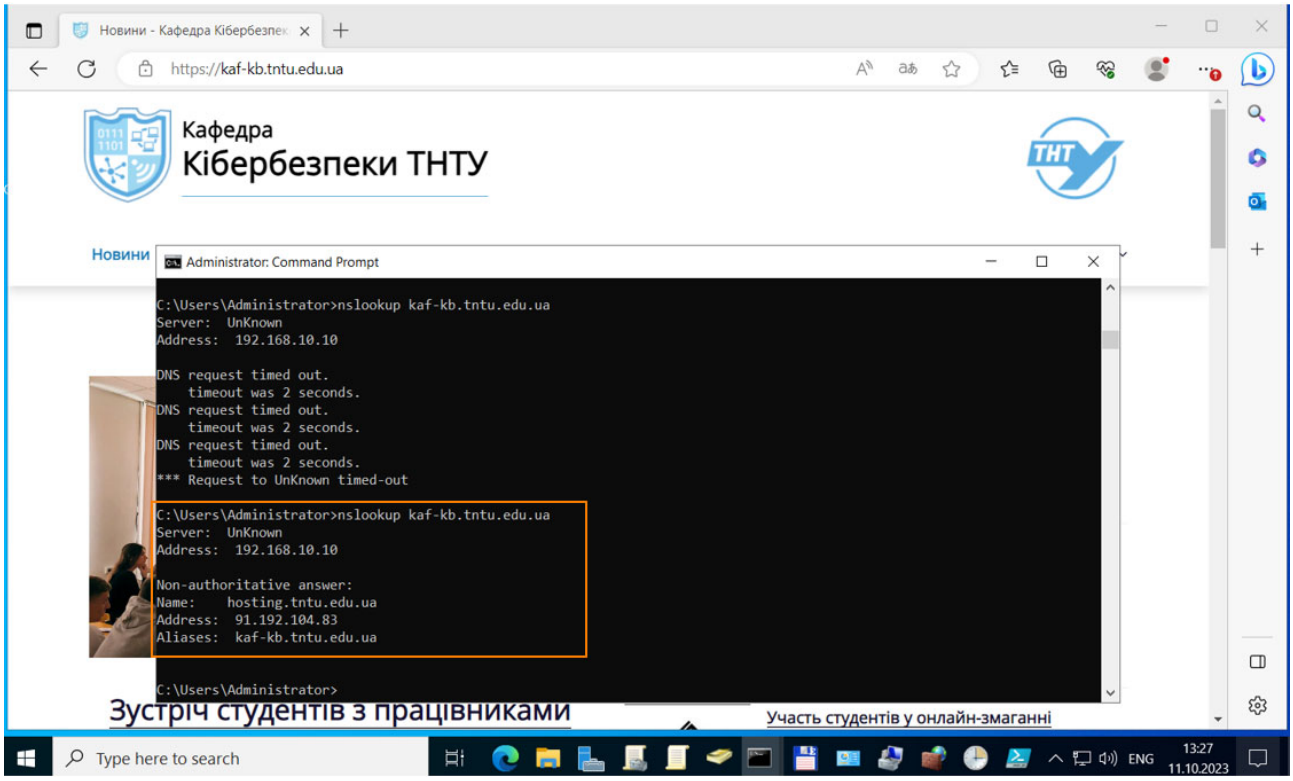


Рисунок 3.8 – Перевірка працездатності DNS сервера з операційної системи Windows

Це доводить те що система відслідковування та блокування UDP flood атаки на DNS сервіс працює коректно та надійно.

4 ОХОРОНА ПРАЦІ ТА БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ

4.1 Охорони праці

Метою даної кваліфікаційної роботи є розробка програмного модуля захисту DNS серверів від UDP flood атак з використанням операційної системи OpenBSD. Дані системи встановлюються на серверному обладнанні, яке розміщене в приміщення спеціального типу. При роботі з даними системами потрібно забезпечити дотримання вимог з охорони праці, техніки безпеки та протипожежної безпеки при використанні ПК.

Основними регламентуючими нормативними документами охорони праці користувачів комп'ютерів є:

– НПАОП 0.00-7.15-18 «Вимоги щодо безпеки та захисту здоров'я працівників під час роботи з екранними пристроями»;

– ДСанПіН 3.3-2.007-98 «Державні санітарні правила і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин»;

– НАПБ А.01.001-2004 «Правила пожежної безпеки в Україні».

Вимоги до приміщень, згідно з [12, 13], щодо розташування робочого місця передбачають виконання наступних вимог:

– мінімальна площа, яка виділяється на одне робоче місце повинна становити мінімум 6,0 м2, при об'ємі – мінімум 20,0 м3;

– розташування робочих місць користувачів ПК заборонено у цокольних або підвальних приміщеннях.

При організації робочих місць у НПАОП 0.00-7.15-18 передбачано наявність природного і штучного освітлення. Зазвичай, природне освітлення поступає у приміщення через вікна та світлові прорізи і забезпечує коефіцієнт освітленості на рівні не менше 1,5%. Орієнтація вікон – на північ або північний схід. Штучне освітлення забезпечують відповідні джерела, наприклад, люмінесцентні лампи. Приміщення з комп'ютерною технікою не повинні межувати з будівлями, де рівень шуму чи вібрації перевищує визначені допустимі значення. Покриття підлоги повинне бути матовим з коефіцієнтом

відбиття 0,3-0,5. Для внутрішнього оздоблення приміщень слід використовувати дифузно-відбивні матеріали з коефіцієнтами відбиття для стелі 0,7-0,8, для стін 0,5-0,6 [12].

У приміщеннях, де організовано робочі місця користувачів ПК, повинні бути забезпечені аптечками першої медичної допомоги. Вологе прибирання у таких приміщеннях є обов'язковим кожного дня.

Щодо ергономічної організації робочого місця, то воно також повинно відповідати вимогам, наведеним у [12, 13]. Конструкція робочого місця повинна забезпечити підтримання оптимальної робочої пози. У відповідності до НПАОП 0.00-7.15-18, обладнання і організація робочого місця працюючих з ЕОМ мають забезпечувати відповідність конструкції всіх елементів робочого місця та їх взаємного, розташування ергономічним вимогам з урахуванням характеру і особливостей трудової діяльності.

Висота робочого столу з ПК повинна бути виконана в діапазоні 680...800 мм, а ширина і глибина – 600...1400 мм і 800..1000 мм відповідно. Стіл також повинен мати достатній простір для ніг, що забезпечить зручну осанку користувача. Стілець на робочому місці користувача ПК повинен бути підйомно-поворотним, регульованим за висотою, за кутом і за нахилом сидіння та спинки [12].

Екран комп'ютера повинен бути розміщений на відстані 600...700 мм від очей користувача. Розташування монітору має забезпечувати зручність зорового спостереження у вертикальній площині під кутом +30 градусів до нормальної лінії погляду працівника [12].

Електромережі штепсельних з'єднань та електророзеток для живлення ПК потрібно виконувати за магістральною схемою. При організації робочих місць електромережу штепсельних розеток для живлення ПК у центрі приміщення прокладають у каналах або під знімною підлогою в металевих трубах або гнучких металевих рукавах [12].

Щодо безпеки при роботі з ПК, щодня перед початком роботи необхідно очищати монітор від пилу та інших забруднень. Після закінчення роботи з ПК, він та периферійні пристрої повинні бути відключені від електричної мережі. У разі виникнення певної аварійної ситуації необхідно негайно відключити ПК від

електричної мережі. Не допускається виконувати обслуговування, ремонт та налагодження ПК безпосередньо на робочому місці [12].

Основні вимоги до пожежної безпеки вказані в НАПБ А.01.001-2004 «Правила пожежної безпеки в Україні». Згідно з [12], на та під приміщеннями, в яких розміщені ЕОМ, а також у суміжних із ними приміщеннях не дозволяється розташування приміщень категорій А та Б за вибухопожежною небезпекою.

Фальшпідлога у приміщеннях з ЕОМ має бути з негорючих матеріалів або матеріалів груп горючості Г1, Г2 з межею вогнестійкості не менше 0,5 години. Простір під нею слід розділяти негорючими діафрагмами на відсіки площею не більше 250 м2. Діафрагми повинні мати межу вогнестійкості не менше 0,75 год. Звукопоглинаюче облицювання стін та стель цих приміщень слід виготовляти з негорючих матеріалів або матеріалів груп горючості Г1, Г2. Персональні комп'ютери після закінчення роботи повинні відключатися від мережі. Не рідше одного разу на квартал необхідно очищати від пилу агрегати та вузли, кабельні канали та простір між підлогами [13].

Приміщення повинні бути забезпечені первинними засобами пожежогасіння, а саме вогнегасниками, що використовуються для локалізації і ліквідації пожеж у їх початковій стадії розвитку.

Вогнегасники слід встановлювати у легкодоступних та помітних місцях (коридорах, біля входів або виходів з приміщень тощо), а також у пожежонебезпечних місцях, де найбільш вірогідна поява осередків пожежі. При цьому необхідно забезпечити їх захист від попадання прямих сонячних променів та безпосередньої (без загороджувальних щитків) дії опалювальних та нагрівальних приладів.

Вибір типу та необхідна кількість вогнегасників визначається відповідно до Типових норм належності вогнегасників, затверджених наказом Міністерства України з питань надзвичайних ситуацій та у справах захисту населення від наслідків Чорнобильської катастрофи від 02.04.2004 № 151.

У кваліфікаційній роботі розроблено метод захисту серверів від атак. Дослідження в роботі вимагали взаємодії людини з серверним обладнанням, тому важливим та актуальним було провести аналіз основних вимог до

приміщень та робочих місць з ПК, що дозволило забезпечити комфортні і безпечні умови праці адміністраторів систем.

4.2. Шум, вібрація, ультразвук, електромагнітні випромінювання у виробничих приміщеннях для роботи з ВДТ та захист від них

Під шумом розуміють набір багаточисельних звуків, які швидко змінюються за частотою, силою і складаються з ряду гармонік [14]. З фізичної точки зору звуки є механічними коливальними рухами частинок пружного середовища в діапазоні частот, що чує людина. Звукові гармоніки розповсюджуються у вигляді хвиль.

Шум є загально-біологічним подразником, діє не тільки на органи слуху, але може викликати порушення роботи серцево-судинної і нервової систем, зумовлювати професійні захворювання [14]. Основними характеристиками звукових коливань є інтенсивність (сила), частота і форма звукової хвилі. Інтенсивність визначається енергією, що переноситься за 1 с звуковою хвилею через поверхню площею 1 м2, яка перпендикулярна напрямку розповсюдження звукової хвилі. Діапазон тисків, що сприймає вухо людини, дуже широкий (10-12Вт/м2 – поріг больового відчуття, верхня межа) [14].

З розвитком промисловості все більший контингент людей підпадає під вплив вібрацій, які являють собою механічні коливання, що передаються тілу людини. Основні параметри вібрацій – частота та амплітуда коливань, але на відміну від шуму, при якому енергія механічних коливань передається через повітряне середовище, при дії вібрацій вона розповсюджується по тканинах і викликає їх коливання або тіла людини в цілому [14]. Найбільш небезпечна вібрація частотою 16-250 Гц, дія якої призводить до вібраційної хвороби.

Нормування шуму здійснюється згідно з “Санітарними нормами допустимих рівнів шуму на робочих місцях”. В Україні застосовується принцип нормування шуму на основі граничних спектрів (гранично допустимих рівнів звукового тиску) в октавних смугах частот та еквівалентних рівнів звуку.

Гранично-допустимі рівні шумів санітарними нормами встановлені для кожного класу [14]:

– для високочастотних шумів (вище 800 Гц) – 75-85 дБ;

– для середньо частотних шумів (300-800 Гц) – 85-90 дБ;

– для низькочастотних шумів (до 300 Гц) – 90-100 дБ.

Шумові явища мають якість кумуляції, накопичуючись в організмі, вони все більше і більше пригнічують нервову систему. Відомо, що після шумової дії інтенсивністю 120 дБ протягом однієї години потрібно 5 годин, щоб гострота слуху повернулась до норми. Стабільні широкосмугові шуми, які перевищують граничний рівень, викликають зниження темпу, ефективності й якості роботи операторів [14].

Ультразвук широко застосовують у технологічних процесах виготовлення радіоелектронної апаратури (промивка деталей, зварювання мініатюрних вузлів тощо) з частотою вище 2220 кГц. При цьому густина енергії ультразвукових коливань у мільйони разів більша густини енергії звуків, які ми чуємо. Тому під його дією відбувається нагрівання тіла, а при дії коливань через рідкі і тверді середовища відбувається розривання і руйнування тканин [14]. Захист від ультразвуку, який діє через повітряне середовище, досягається шляхом звукоізоляції установок (листова сталь, дюралюміній, що обклеєні гумою або руберойдом, гетинакс) або розміщення їх в окремій звукоізолюючій кабіні. Ультразвукові установки повинні мати блокування, яке відключає генератор ультразвукових коливань в момент відкривання кришок або кожухів [14]. Для запобігання шкідливої дії шуму і вібрації на організм працюючих проводяться технічні, організаційні і медико-профілактичні заходи. Одним з основних технічних заходів є зменшення при експлуатації та на стадії проектування, конструювання обладнання причин шуму і вібрації в самому джерелі утворення. Досягають цього завдяки використанню раціональної конструкції обладнання, заміни ударної дії деталей і машин коливальною, з’єднання елементів гнучкими зв’язками, врівноважування обертових частин механізмів, заміни металевих деталей пластмасовими, забезпечення різних власних частот коливань механізму

з частотою збуджуючої сили [14]. Якщо неможливо ізолювати чи знизити шум і вібрацію самого джерела, потрібно:

– ізолювати джерело шуму або вібрації від навколишнього середовища засобами вібро- та звукоізоляції;

– раціонально планувати виробничі приміщення, що мають інтенсивні джерела шуму;

– збільшувати звукопоглинання внутрішніх поверхонь приміщення шляхом звукопоглинальних покриттів.

Якщо не вдається зменшити рівень шуму і вібрації на робочому місці до нормативних значень та необхідно використовувати засоби індивідуального захисту: рукавиці, взуття, навушники, м’які шоломи, які зменшують рівень звукового тиску на 40-50 дБ.

У процесі виробництв, експлуатації і зберігання комп’ютерної і радіоелектронної апаратури можуть виникати механічні і динамічні дії, що характеризуються широким діапазоном частот коливань, а також амплітудою, прискоренням і часом дії [14].

При експлуатації високочастотного обладнання всередині виробничих приміщень зниження напруженості електромагнітного випромінювання досягається такими методами:

– захист часом – обмеження часу перебування людини в електромагнітному полі, що залежить від інтенсивності опромінення або напруженості ЕМП.

– захист відстанню застосовується при неможливості послабити інтенсивність опромінення в заданій зоні іншими методами: збільшують відстань між джерелом випромінювання і обслуговуючим персоналом;

– добре виконане екранування джерела і усунення нещільності у фланцевих з’єднаннях, фідерів, зазорів у обшивці корпусів, нещільних електричних контактів;

– проведення дистанційного контролю й управління роботою передавачів з екранованого приміщення;

– засобами індивідуального захисту.

В залежності від типу джерела випромінювання, його потужності, характеру технологічного процесу може застосовуватись один з вказаних методів або будь-яка їх комбінація.

ВИСНОВКИ

В ході виконання кваліфікаційної роботи освітнього рівня "Магістр" було проведено комплексне дослідження аспектів захисту DNS серверів від UDP flood атак. Робота включала в себе аналіз актуальності теми, вивчення методів захисту, розробку програмного модуля для виявлення та блокування атак, а також експериментальне тестування розробленого заходу захисту.

У ході дослідження було ретельно розглянуто поняття UDP flood атак та їх наслідки для DNS серверів. Були визначені загальні методи захисту, включаючи фільтрацію на рівні мережевого обладнання, захист на рівні сервера, моніторинг трафіку та виявлення аномалій, захист від ботнетів, доступність резервних серверів та виявлення та відповідь на атаку.

Окремо було проведено огляд роботи DNS серверів та операційної системи OpenBSD, які використовувались у дослідженні. Детально були наведені конфігурації брандмауера PF та операційної системи OpenBSD для забезпечення захисту DNS серверу від UDP flood атак.

Одним із ключових результатів дослідження є розробка Python-скрипта, який автоматично виявляє та блокує UDP flood атаки на DNS сервері. Скрипт використовує дані з брандмауера PF та інші інструменти для реагування на атаки.

Всі розроблені заходи захисту та методи були піддані практичному тестуванню, що дозволило підтвердити їх ефективність у реальних умовах. Результати тестування свідчать про успішну реалізацію заходів захисту та їх здатність ефективно захищати DNS сервери від UDP flood атак.

Отже, дана кваліфікаційна робота має важливе значення для підвищення рівня безпеки та стабільності роботи DNS серверів. Розроблені методи та програмний модуль можуть бути використані адміністраторами систем для захисту своїх DNS серверів від потенційних атак та забезпечення безперебійної роботи служби.

Крім того, результати цієї кваліфікаційної роботи можуть бути корисні в освітньому процесі. Розроблені методи та скрипти можуть стати цінним ресурсом для навчання. Вони надають студентам можливість покращити своє

розуміння концепцій атак на безпеку мережі, їх наслідків і методів захисту від них.

Студенти зможуть вивчити, як працюють UDP flood атаки, як вони можуть загрожувати системі та як їх можна відвернути. Це сприятиме підвищенню обізнаності студентів у галузі кібербезпеки та підготовці їх до вирішення сучасних викликів у цій сфері."

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. UDP Based Denial-of-Service (DoS) Attack [Електронний ресурс]. — URL: https://www.ncsc.gov.ie/emailsfrom/Resources/UDP-DoS-Attack/ (дата звернення: 29.10.2023).
2. DDoS QUICK GUIDE [Електронний ресурс]. — URL: https://www.cisa.gov/sites/default/files/publications/DDoS%20Quick%20Guide.pdf (дата звернення: 29.10.2023).
3. What is DNS? [Електронний ресурс]. — URL: https://aws.amazon.com/route53/what-is-dns/ (дата звернення: 29.10.2023).
4. What is a DNS flood? [Електронний ресурс]. — URL: https://www.cloudflare.com/learning/ddos/dns-flood-ddos-attack/ (дата звернення: 29.10.2023).
5. OpenBSD Frequently Asked Questions [Електронний ресурс]. — URL: https://www.openbsd.org/faq/index.html (дата звернення: 29.10.2023).
6. OpenBSD PF - User's Guide [Електронний ресурс]. — URL: https://www.openbsd.org/faq/pf/ (дата звернення: 29.10.2023).
7. Parrot Security Edition [Електронний ресурс]. — URL: https://parrotsec.org/ (дата звернення: 29.10.2023).
8. Red Hat Enterprise Linux [Електронний ресурс]. — URL: https://access.redhat.com/products/red-hat-enterprise-linux/ (дата звернення: 29.10.2023).
9. UDP and TCP Packet Crafting Techniques using hping3 [Електронний ресурс]. — URL: https://github.com/Samsar4/Ethical-Hacking-Labs/blob/master/2-Scanning-Networks/1-hping3.md (дата звернення: 29.10.2023).
10. BIND 9 Administrator Reference Manual [Електронний ресурс]. — URL: https://bind9.readthedocs.io/en/latest/# (дата звернення: 29.10.2023).
11. Python documentation [Електронний ресурс]. — URL: https://www.python.org/doc/ (дата звернення: 29.10.2023).

12. НПАОП 0.00-7.15-18 Вимоги щодо безпеки та захисту здоров’я працівників під час роботи з екранними пристроями. Київ. 2018.
13. Катренко Л.А., Катренко А.В. Охорона праці в галузі комп’ютинґу. Львів: Магнолія-2006. 2012. 544 с.
14. Желібо Е.Н. Безпека життєдіяльності: Навчальний посібник. Київ: Каравела, Львів: Новий світ - 2000. 2001. 320с.

Додаток А – Публікація

Іваночко Назар Андрійович, здобувач другого (магістерського) рівня вищої освіти факультету комп'ютерно-інформаційних систем і програмної інженерії
Тернопільський національний технічний університет імені Івана Пулюя, Україна

Тимощук Віталій Дмитрович, здобувач першого (бакалаврського) рівня вищої освіти факультету прикладних інформаційних технологій та електроінженерії
Тернопільський національний технічний університет імені Івана Пулюя, Україна

Букатка Соломія Романівна, здобувачка першого (бакалаврського) рівня вищої освіти факультету комп'ютерно-інформаційних систем і програмної інженерії
Тернопільський національний технічний університет імені Івана Пулюя, Україна

Науковий керівник: Тимощук Дмитро Іванович, старший викладач кафедри кібербезпеки
Тернопільський національний технічний університет імені Івана Пулюя, Україна

РОЗРОБКА ТА ВПРОВАДЖЕННЯ ЗАХОДІВ ЗАХИСТУ ВІД UDP FLOOD АТАК НА DNS СЕРВЕР

У сучасному світі, де цифрові технології стали неодмінною частиною нашого життя, безпека в мережі є критичним аспектом функціонування будь-якої системи. Особливу увагу привертають атаки типу UDP Flood на DNS (Domain Name System) сервери. Ці види атак можуть створити загрозу для інформаційної безпеки, спричиняючи перебої у обслуговуванні. Тому виникає необхідність у вдосконаленні методів захисту від цих атак.

UDP flood атаки на сервери DNS є одними з найбільш розповсюджених та ефективних методів атаки на цю службу. Ця атака використовує протокол UDP та порт 53. Даний протокол є протоколом без збереження стану та гарантії доставки, що дозволяє відправляти безперервні запити (UDP-пакети) до сервера без підтвердження доставки.

Ця атака є особливо ефективною через свою простоту в реалізації. Зловмисники можуть легко запустити UDP Flood атаку за допомогою ботнетів (мережі скомпрометованих комп'ютерів), які надсилають велику кількість UDP-пакетів до DNS сервера.

DNS сервери використовуються для перетворення доменних імен в IP-адреси, що дозволяє користувачам зручно користуватись послугами у мережі Інтернет. У разі UDP Flood атаки на DNS, сервер отримує велику кількість фальшивих запитів, що перевантажують його ресурси та призводять до тимчасової недоступності сервісу для легітимних користувачів [1].

Дослідницька робота була спрямована на вивчення та аналіз проблеми UDP Flood атак на DNS сервери та розробку ефективних заходів мінімізації їх впливу.

У результаті дослідження був розроблений програмний модуль для захисту DNS серверів від UDP flood атак, який базується на операційній системі OpenBSD, системі фільтрації пакетів PF [2] та мові програмування Python. Цей модуль в режимі реального часу аналізує дані, отримані через PF, та має здатність автоматично блокувати IP-адреси зловмисників у випадку виявлення атаки, сповіщаючи адміністратора про цей інцидент.

Всі розроблені методи та заходи захисту були піддані практичним випробуванням, які підтвердили їх ефективність у реальних умовах. Результати тестування підтверджують успішну інтеграцію заходів захисту та їх здатність ефективно зменшувати вплив UDP flood атаки на роботу DNS сервера.

Результат цього дослідження є кроком у напрямку покращення безпеки мережі та захисту інфраструктури DNS від потенційних загроз, що допоможе забезпечити стабільну та безперебійну роботу мережевої інфраструктури у цифровому світі.

Список використаних джерел:

1. UDP Based Denial-of-Service (DoS) Attack [Електронний ресурс]. — URL: https://www.ncsc.gov.ie/emailsfrom/Resources/UDP-DoS-Attack/ (дата звернення: 29.10.2023).
2. OpenBSD PF - User's Guide [Електронний ресурс]. — URL: https://www.openbsd.org/faq/pf/ (дата звернення: 29.10.2023).

Додаток Б – Лістинг файлу udpflood_monitor.py

```
import subprocess
import socket
import datetime
import smtplib
from email.mime.text import MIMEText

# Get current time
current_time = datetime.datetime.now().strftime("DATE: %Y-%m-%d
TIME: %H:%M:%S")
tema = f"router.cstntu.lan {current_time} UDP flood DNS info"

try:
    # Execute the pfctl command to retrieve the firewall state
    output = subprocess.check_output("/sbin/pfctl -s state | grep
192.168.10.10:53", shell=True)
    output = output.decode().strip().split('\n')  # Split the output
into lines

    udpflood_ips = []

    # Iterate through each line of the output
    for line in output:
        if "NO_TRAFFIC:SINGLE" in line:
            parts = line.split()  # Split the line into words
            if len(parts) >= 6:
                src_ip = parts[4].split(':')[0]  # Extract the IP
address from the line
                udpflood_ips.append(src_ip)

    count = len(udpflood_ips)

    if count == 0:
        print("No UDP DNS flood attacks detected.")
    else:
        if count > 100:
```

```
            unique_ips = list(set(udpflood_ips))      # Remove
duplicates
            unique_ips_str = "\n".join(unique_ips)   # Convert the
list to a string with newline characters

            # Write the unique IP addresses to a file
            with open('/root/udpflooddns/udpflooddnsip', 'w') as
file:
                file.write(unique_ips_str+"\n")

            # Logic for handling attacks with more than 100 IP
addresses
            # Add IP addresses to the pfctl table

            with open('/root/udpflooddns/ipblockudpflood.txt', 'a')
as ipblockudpflood:
                subprocess.call(f"pfctl -t udpflooddns -T add -f
/root/udpflooddns/udpflooddnsip",                          shell=True,
stdout=ipblockudpflood, stderr=subprocess.STDOUT)

            email_message = f"UDP FLOOD DNS Attack IP count - {count}
IP unique - {count_unique}"
            msg = MIMEText(email_message)
            msg["Subject"] = tema
            msg["From"] = "alert@server.XXXXX.te.ua"
            msg["To"] = "alert.XXXXX.YYYYY@gmail.com"

            smtp_server = "mail.cstntu.lan"
            smtp_port = 25

            server = smtplib.SMTP(smtp_server, smtp_port)
            server.sendmail("alert@server.XXXXX.te.ua",
["alert.XXXXX.YYYYY@gmail.com"], msg.as_string())
            server.quit()
            print(email_message)

except subprocess.CalledProcessError as e:
```

```
if e.returncode == 1:
    print("No UDP DNS flood attacks detected (except).")
else:
    print(f"Error: {e}")
```