МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Харківський національний університет імені В.Н.Каразіна

Факультет математики і інформатики

Кафедра теоретичної та прикладної інформатики

Кваліфікаційна робота

бакалавр

на тему “Розробка фаерволу для фільтрації контенту в мережі інтернет”

Виконав: студент 4 курсу, групи МФ-42

спеціальність 122 «Комп’ютерні науки»

освітньо-професійна програма

«Інформатика»

Поляк Я. В.

Керівник Є.С. Меняйлов

Рецензент_____

(прізвище та ініціали)

Харків – 2023 року

1. ВСТУП

1.1 Формулювання мети роботи, задач та обґрунтування актуальності теми

Ефективність роботи в офісі має важливе значення оскільки робота в колективі та атмосфера робочого місця відіграє не останнє місце в ефективності та відповідно і в отриманих результатах. Беручи до уваги дані з ресурсу ZIPPIA (рис. 1.1) ми бачимо що 77% працівників користуються соціальними мережами на роботі. А середнє значення втраченого часу досягає 12%.

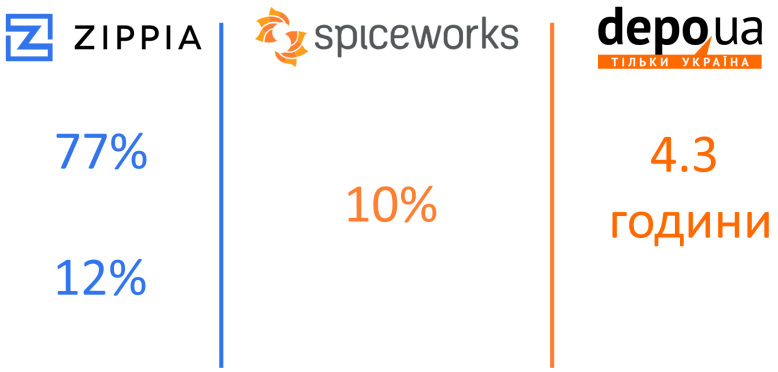


Рисунок 1.1 – Дані з ресурсів ZIPPIA, spiceworks та depoua.

Портал spiceworks повідомляє про те що в середньому кожен працівник використовує 10% робочого часу на непотрібні сайти та соціальні мережі. Схожі дані демонструє і стаття порталу depoua, конкретизуючи те що з 4 годин 30 хвилин загального витраченого часу більша частина а саме 2 години працівник витрачає на сайти новині та соціальні мережі. Виходячи з цього,

стає зрозуміло для того щоб працівник виконував всі покладені на нього задачі необхідно так чи інакше контролювати його роботу, враховуючи що більшість роботи проходить з використанням персональних комп’ютерів які підключенні до мережі потрібно контролювати або фільтрувати дані які надходять до цієї мережі.

Для цього існує така річ як фаєрвол, який допомагає як захистити мережу від зловмисників так, і фільтрувати контент який до мережі надходить. Термін фаєрвол походить від англійського слова «firewall», що означає протипожежну стіну, яка перешкоджає поширення вогню та пом'якшує шкідливий вплив на людину. В мережевій безпеці фаєрвол являє собою систему на основі програмного або апаратного забезпечення, яка є своєрідним посередником між безпечними та неперевіреними мережами (рис. 2.1) , а також їх частинами.

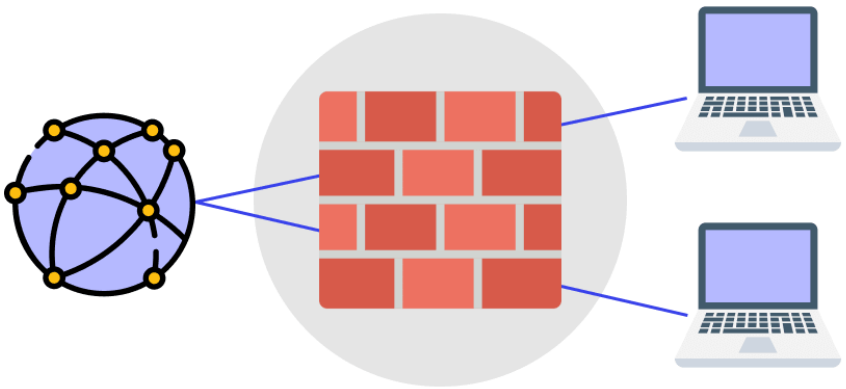


Рисунок 1.2 – Ілюстрація роботи фаєрволу

Головна функція його функція фільтрація шкідливого та потенційно небезпечного контенту та з’єднань.

Актуальність фаєрволу в сьогоднішній час полягає в тому, що він є необхідним інструментом для захисту комп'ютерної мережі від шкідливих атак та вірусів. З огляду на постійне збільшення кількості кіберзагроз та

розвиток нових технологій, фаєрвол стає ще більш актуальним і важливим інструментом для забезпечення безпеки в мережі. За допомогою фаєрволу можна обмежувати контент з конкретної IP-адреси, або навіть з діапазону адрес. Також можливо блокувати контент який находить з конкретного доменного імені, або контент який містить ключові слова. Основною метою роботи є розробка фаєрволу який дасть змогу обмежувати контент який поступає до офісної мережі. Це дозволить забезпечити безпеку даних мережі від шкідливих атак, вірусів та інших загроз. Також це буде регулюванням доступу до певних типів веб-сайтів та додатків, що підвищить ефективність роботи працівників та зменшить відволікання.

Розроблений фаєрвол повинен містити в собі наступні елементи:

- блокування контенту з сайтів за доменним іменем
- блокування за ключовими словами
- блокування за конкретною IP-адресою, діапазоном адрес
- блокування за портом
- зрозумілий код

Таким чином розроблений фаєрвол зможе виконувати функцію фільтра для офісної мережі, що піде тільки на користь як працівникам зменшивши їх відволікання від непотрібної інформації, так і компанії внаслідок підвищення ефективності роботи.

1.2 Стислий огляд відомих результатів в області дослідження

Фаєрволи є одним з найважливіших інструментів безпеки мережі, вони відіграють вирішальну роль у захисті офісних мереж від кіберзагроз і небажаного контенту. Хоча існує багато різних типів фаєрволів, найпоширенішими є програмні та апаратні.

Програмні встановлюються на окремих комп’ютерах або серверах і можуть блокувати вхідний і вихідний мережевий трафік на основі заздалегідь

визначених правил. Зазвичай вони дешевші, ніж апаратні брандмауери, але можуть бути менш ефективними для фільтрації мережевого трафіку через те що вони фільтрують контент тільки на серверах та комп'ютерах де вони встановленні на відміну від апаратних. Одними з найпопулярніших програмних брандмауерів вважають:

–Windows Firewall: це вбудований брандмауер, який постачається з операційними системами Microsoft Windows. Він простий у використанні та налаштуванні та забезпечує базовий захист від вхідного мережевого трафіку. Однак він може бути не таким ефективним, як деякі брандмауери сторонніх виробників, щодо блокування вихідного трафіку або надання додаткових функцій;

–Norton Personal Firewall: це популярне програмне забезпечення від компанії Norton. Він надає такі розширені функції, як запобігання вторгненням і захист конфіденційності, але може бути доволі дорогим для простого користувача, також може впливати на роботу комп'ютера;

–McAfee Personal Firewall: це ще одне популярне програмне забезпечення брандмауера, яке надає такі функції, як виявлення вторгнень у мережу, контроль програм і блокування IP-адрес. Однак ціна та потреба в значних системних ресурсах є вагомим мінусом.

Апаратні фаєрволи являють собою фізичні пристрої, які встановлюються між внутрішньою мережею компанії та інтернетом. Фільтруючи мережевий трафік на рівні мережі, вони дозволяють адміністраторам контролювати, які типи трафіку дозволені через фаєрвол, а які слід заблокувати. Хоча апаратні брандмауери дорожчі за програмні брандмауери, вони забезпечують вищий рівень безпеки та їх складніше обійти.

Найпопулярніші апаратні фаєрволи:

–Cisco ASA: це популярний брандмауер, який пропонує компанія Cisco. Він надає такі розширені функції, як підтримка VPN, фільтрація вмісту та

запобігання несанкціонованому вторгненню, але є доволі дорогим і складним у налаштуванні.

− SonicWall: надає такі функції, як SSL VPN, глибока перевірка пакетів і фільтрація вмісту. Однак він потребує додаткових ліцензій для виконання певних функцій.

−Fortinet: надає такі функції, як веб-фільтрація, контроль програм і запобігання вторгненню. Однак Fortinet складний для налаштування та може потребувати багато системних ресурсів.

Всупереч своїй користі, фаєрволи мають деякі недоліки. Одним із головних недоліків є те, що вони можуть блокувати корисний контент, який не є загрозою для мережі. Наприклад, може бути заблокованим доступ до веб-сайту, який потрібен працівнику для його роботи, або перешкоджати завантаженню необхідного оновлення програмного забезпечення. Це може призвести до зниження продуктивності в офісі. Ще одним вагомим недоліком фаєрволів є те, що їх можна обійти. Деякі співробітники можуть використовувати віртуальні приватні мережі (VPN), щоб обійти брандмауер і отримати доступ до заблокованого контенту. Інші можуть використовувати проксі-сервери або інші методи для обходу. Це може призвести до значного ризику для безпеки, оскільки може дозволити зловмисникам отримати доступ до внутрішньої мережі компанії. Також значним недоліком можна вважати ціну за фаєрволи, яка в деяких випадках може бути рівномірною заробітній платі працівника компанії.

Щоб зменшити загрозу проникнення, важливо вибрати фаєрвол, який відповідає потребам і характеристикам конкретної офісної мережі. Налаштування повинно дозволяти доступ до необхідного вмісту, одночасно блокуючи несанкціонований трафік. Також важливо підтримувати стан оновленнями програмного забезпечення з останніми виправленнями безпеки, щоб забезпечити максимальний захист. Вибравши правильний фаєрвол і додаткові параметри його налаштування, адміністратори можуть забезпечити

високий рівень захисту своєї офісної мережі, зберігаючи продуктивність і ефективність співробітників.

Мета дослідження — полягає в розробці надійного та ефективного фаєрволу для фільтрації контенту в офісних мережах, який працює на операційній системі Linux. Створений фаєрвол повинен регулювати мережевий трафік на основі набору правил і параметрів які введе користувач, що дасть змогу забезпечити максимальну безпеку мережі, дозволяючи доступ до необхідних ресурсів.

Враховуючи поставлену мету, в роботі будуть поставлені наступні задачі:

1. Проаналізувати наявні аналогові системи, їх переваги та недоліки;
2. Аналіз інформації яку необхідно блокувати\дозволяти та її основні типи;
3. Вибір та аргументація методів дослідження;
4. Розробка архітектури фаєрволу;
5. Розробка ключових функцій та опис принципу їх роботи;
6. Тестування фаєрволу;
7. Аналіз результатів розробки та формулювання висновків.

1.3 Відомості про одержані результати та їх новизна

Результатом використання розробленого фаєрволу є: підвищення безпеки та захисту офісних мереж від шкідливого контенту. Важливим етапом у формуванні заборон та дозволів відповідного контенту до фаєрволу є присутність балансу між безпекою та продуктивністю співробітників. Хоча фаєрвол є важливими для захисту мереж, він може бути надто обмежувальним, блокуючи доступ до корисного вмісту, який не є загрозою. Щоб підтримувати продуктивність і ефективність співробітників,

адміністратори повинні налаштувати фаєрволи продумано, щоб дозволити доступ до необхідного вмісту, зберігаючи при цьому високий рівень безпеки. З точки зору розробки фаєрволу такий функціонал надано, адміністраторам необхідно ввести дані в відповідні поля та натиснути кнопку для блокування\розблокування певного контенту Потрібно розуміти, що типи сайтів, які повинні блокуватися брандмауером для офісних працівників, більшою мірою залежать від двох ключових елементів політики компанії та характеру її діяльності.

Проте, існують загальні рекомендації щодо шкідливих та небажаних для відвідування сайтів, загалом рекомендується блокувати сайти, які можуть становити загрозу безпеці або негативно впливати на продуктивність співробітників.

Деякі з типів сайтів, які слід заблокувати при використанні розробленого фаєрволу:

- Неофіційні або шкідливі: сайти що містять зловмисне програмне забезпечення, фішинг-шахрайство чи інший тип шкідливого вмісту, слід блокувати, щоб запобігти випадковому завантаженню або доступу до шкідливого вмісту співробітниками;
- Вміст для дорослих: доступ до вмісту для дорослих слід заблокувати, щоб запобігти доступу до неприйнятного чи образливого матеріалу на пристроях або в мережах компанії.
- Соціальні медіа та розважальні сайти: хоча блокувати всі соціальні медіа та розважальні сайти може бути непрактично, рекомендується обмежити доступ до цих сайтів у робочий час, щоб запобігти відволіканню співробітників або марнуванню часу.
- Торент-сайти та сайти обміну файлами: ці сайти можуть становити значний ризик для безпеки, оскільки вони часто розміщують незаконний або піратський вміст, який може містити зловмисне програмне забезпечення або інші типи зловмисного програмного забезпечення.

З іншого боку, є також сайти, які повинні бути дозволені брандмауером, щоб забезпечити працівникам доступ до інформації, необхідної їм для ефективного виконання роботи.

Деякі приклади сайтів, які слід дозволити, включають:

- Сайти, пов’язані з роботою. Мають бути дозволені сайти, необхідні працівникам для виконання своїх службових обов’язків, як-от сайти новин, пов’язані з бізнесом, галузеві сайти та інструменти для співпраці, як-от Microsoft Teams.
- Освітні та дослідницькі сайти: сайти, необхідні працівникам для проведення досліджень або збору інформації, пов’язаної з їхньою роботою, мають бути дозволені, якщо вони не класифікуються як загрози безпеці.
- Надійні джерела новин. Слід надати доступ до надійних джерел новин, щоб співробітники могли бути в курсі поточних подій, які можуть вплинути на їх роботу.

Дотримуючись цих вказівок, адміністратори можуть забезпечити високий рівень захисту своєї офісної мережі при використанні розробленого фаєрволу та зберігаючи при цьому продуктивність та ефективність працівників. Враховуючи те що більшість з теперішніх фаєрволів на linux мають значну складність конфігурування та неінтуїтивний інтерфейс запропонований в роботі фаєрвол буде містити в собі ряд певних переваг та своєрідну новизну захисту.

Найважливішим у фаєрволі є функціонал тому створений в роботі фаєрвол буде містити в собі функції які легко будуть налаштовуватись адміністратором мережі, такі як введення ключових слів для блокування контенту, IP-адрес, портів та доменних імен що дасть змогу адміністратору в повною мірою керувати контентом яких надходить до мережі. Стосовно інтерфейсу, то він буде максимально простий, щоб не навантажувати користувача непотрібними діями та інформацією. Проте, дизайн та палітра кольорів повинні бути зрозумілими та приємними в використанні.

Підсумовуючи вище написане, фаєрвол є важливим інструментом для захисту офісних мереж від кіберзагроз і небажаного контенту. В розробці фаєрволу вдалось знайти баланс між блокуванням сайтів, які становлять загрозу безпеці або негативно впливають на продуктивність співробітників і водночас ресурсів, необхідних для ефективного виконання своїх робочих функцій. Адміністратори мають всі необхідні функції для збалансування безпеки та продуктивності співробітників.

2. ОСНОВНА ЧАСТИНА

2.1 Постановка задачі

Однією з ключових задач у забезпеченні безпеки операційної системи Linux є належна настройка фаєрволу. Фаєрвол являє собою програмне забезпечення, яке контролює доступ до мережевих ресурсів і регулює трафік на основі визначених правил. Створення фаєрволу це процес налаштування таких правил та параметрів, щоб забезпечити максимальний рівень безпеки операційної системи. Для створення фаєрволу в операційній системі Linux можна скористатися декількома різними інструментами. Одним з найпоширеніших є iptables програма, яка дозволяє керувати мережевим трафіком на рівні ядра операційної системи.

Основні задачі роботи можна окреслити в декілька етапів, а саме:

- Створення файлу правил конфігурацій фаєрволу;
- Створення та налаштування правил;
- Визначення параметрів та правил доступу до мережевих ресурсів, які будуть дозволені для вхідного трафіку;
- Визначення параметрів та правил доступу до мережевих ресурсів, які будуть заблоковані для трафіку;

При створенні фаєрволу в операційній системі Linux важливо враховувати багато факторів, таких як тип мережі, яка використовується, конфігурація мережевих пристроїв, тип мережевих сервісів, які використовуються, рівень безпеки, який необхідно забезпечити, та багато інших. Тому важливо ретельно продумати кожну з цих складових перед початком створення фаєрволу. Створення файлу правил конфігурації фаєрволу є першим кроком в процесі налаштування фаєрволу. Файл правил конфігурації містить визначені правила та параметри, які визначають, який

трафік дозволений або заблокований на мережевому рівні. Для створення файлу правил конфігурації фаєрволу в операційній системі Linux можна використовувати такі інструменти, як iptables-save або iptables-restore.

Другим етапом у створенні фаєрволу є створення та налаштування правил. Правила визначають, який трафік дозволяється або заблоковується на мережевому рівні. Правила можуть бути створені для конкретних протоколів, портів, IP-адрес або підсітей. Крім того, правила можуть бути встановлені з різними параметрами, наприклад, забороняти або дозволяти конкретні види трафіку.

Третім етапом у створенні фаєрволу є визначення параметрів та правил доступу до мережевих ресурсів, які будуть дозволені для вхідного трафіку.

2.2 Розвинутий огляд сучасного стану справ в області дослідження

Linux операційна система з відкритим вихідним кодом, яка зазвичай використовується для роботи з серверами. Вона надає багато інструментів для захисту мережі, включаючи фаєрволи. Фаєрволи дозволяють користувачам контролювати доступ до мережі, блокуючи небажані порти та IP-адреси, а також налаштовувати правила доступу на основі MAC-адрес. За допомогою цих функцій, користувачі можуть захистити свою мережу від атак з мережі, шкідливого програмного забезпечення або просто небажаного контенту. Одним з найбільш популярних інструментів для фаєрволів на Linux є iptables, ілюстрація роботи Iptables зображена на рисунку 2.1.

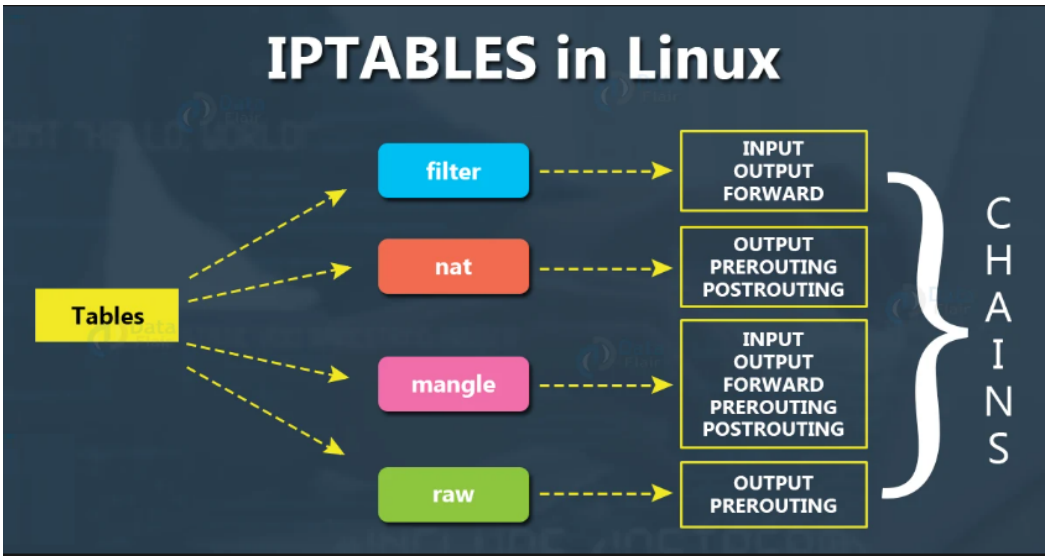


Рисунок 2.1 – Принцип роботи Iptables

Iptables - це традиційний фаєрвол для Linux, в якому користувач може використовувати правила для контролю доступу до мережі. За допомогою iptables, користувач може налаштувати правила для кожного порту та IP-адреси, що дозволяє точно настроїти захист мережі.

Однак, iptables має складний синтаксис, тому використання цього інструмента може бути важким для новачків.

Новіші інструменти для фаєрволів на Linux, такі як firewalld, надають простіший інтерфейс для користувачів, щоб налаштувати правила фаєрвола. Firewalld має графічний інтерфейс та підтримку наборів правил, що дозволяє користувачам налаштовувати правила для різних сценаріїв. Також, firewalld підтримує зони, що дозволяє налаштовувати правила доступу для різних мережевих зон, таких як домашня мережа або мережа в офісі.

Одним з найбільш важливих аспектів фаєрволів на Linux є їх спроможність виявляти та блокувати атаки. Для цього, більшість фаєрволів мають функції, такі як виявлення вторгнень та блокування небезпечних дій. Наприклад, fail2ban - це програма, яка перевіряє журнали системних подій та блокує IP-адреси, з яких здійснювалися небезпечні дії. Ще одним аспектом

фаєрволів на Linux є їх здатність до інтеграції з іншими системами захисту мережі, такими як системи виявлення вторгнень (IDS). Інтеграція фаєрволів з системами IDS дозволяє користувачам належним чином аналізувати мережевий трафік та виявляти небезпечні дії.

Щодо роботи фаєрволу в офісі перш за все, слід зазначити, що використання фаєрволу для блокування контенту офісних працівників може мати як позитивні, так і негативні наслідки. З одного боку, такий захід може допомогти зберегти час та ресурси, що можуть бути витрачені на перегляд небезпечного або непродуктивного контенту. З іншого боку, це може порушити приватність працівників та викликати незадоволення серед колективу. За даними досліджень, у більшості випадків компанії, що використовують фаєрволи для блокування контенту, роблять це з метою забезпечити безпеку мережі та попередити витрати часу працівників на роботу з непродуктивним контентом. Таким чином, фаєрволи можуть бути корисним інструментом для захисту мережі від шкідливого контенту та вірусів. Нижче наведено актівіті-діаграму фаєрволу, яка візуально демонструє процес фільтрації трафіку в офісній мережі:

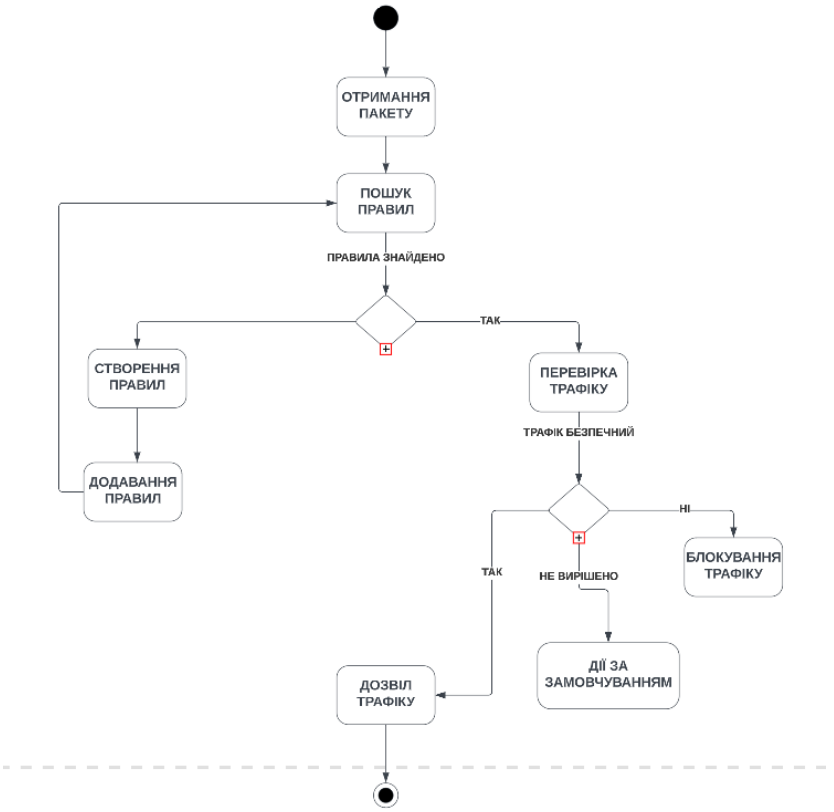


Рисунок 2.2 – Activity Diagram фаєрволу

Одним з головних викликів, який виникає при роботі з фаєрволами на Linux, є забезпечення безпеки при налаштуванні правил. Некоректне налаштування правил може призвести до несправностей в мережі та вразливостей у системі захисту. Тому, перед початком роботи з фаєрволами, користувачам необхідно ознайомитися з основними принципами фаєрволів та налаштування правил. Також, слід зазначити, що користувачі можуть легко обійти блокування, використовуючи різні методи, такі як використання VPN або проксі-серверів. Тому, використання фаєрволу для блокування контенту повинно бути доповнене іншими заходами безпеки та надійності.

2.3 Методи дослідження

Методи дослідження є фундаментальними інструментами в дослідженнях різних наукових дисциплін, включаючи інформатику та комп'ютерні науки. В контексті створення фаєрволу для фільтрації контенту в офісі на операційну систему Linux, методи дослідження можуть використовуватися для того, щоб виявити потенційні загрози та проблеми безпеки мережі, а також для знаходження оптимальних рішень для налаштування фаєрволу.

Один з методів дослідження, який можна використовувати в контексті створення фаєрволу для фільтрації контенту в офісі на операційну систему Linux, аналіз мережевого трафіку. Аналіз мережевого трафіку дозволяє виявити потенційно небезпечні з'єднання, які можуть бути використані для атак на мережу, а також для виявлення зайвого трафіку, який може бути блокований для забезпечення оптимальної продуктивності мережі. Для проведення аналізу мережевого трафіку можна використовувати спеціальні програми, такі як Wireshark, tcpdump, tshark, ngrep та інші. Ці програми дозволяють перехоплювати трафік на різних рівнях мережевої моделі OSI (Open Systems Interconnection) і аналізувати його.

Наприклад, Wireshark є однією з найпопулярніших програм для аналізу мережевого трафіку. Вона підтримує різні протоколи мережі, такі як TCP, UDP, HTTP, DNS та інші. Щоб використовувати Wireshark для аналізу мережевого трафіку, спочатку необхідно встановити програму на операційну систему Linux. Після встановлення програми можна запустити її та вибрати мережевий інтерфейс, на якому буде проводитись аналіз трафіку. Після вибору мережевого інтерфейсу Wireshark почне перехоплювати трафік, який протікає через цей інтерфейс.

Вікно програми складається з трьох основних частин(рис.2.3): панелі зі списком пакетів, панелі з деталями пакету та панелі зі статистикою.

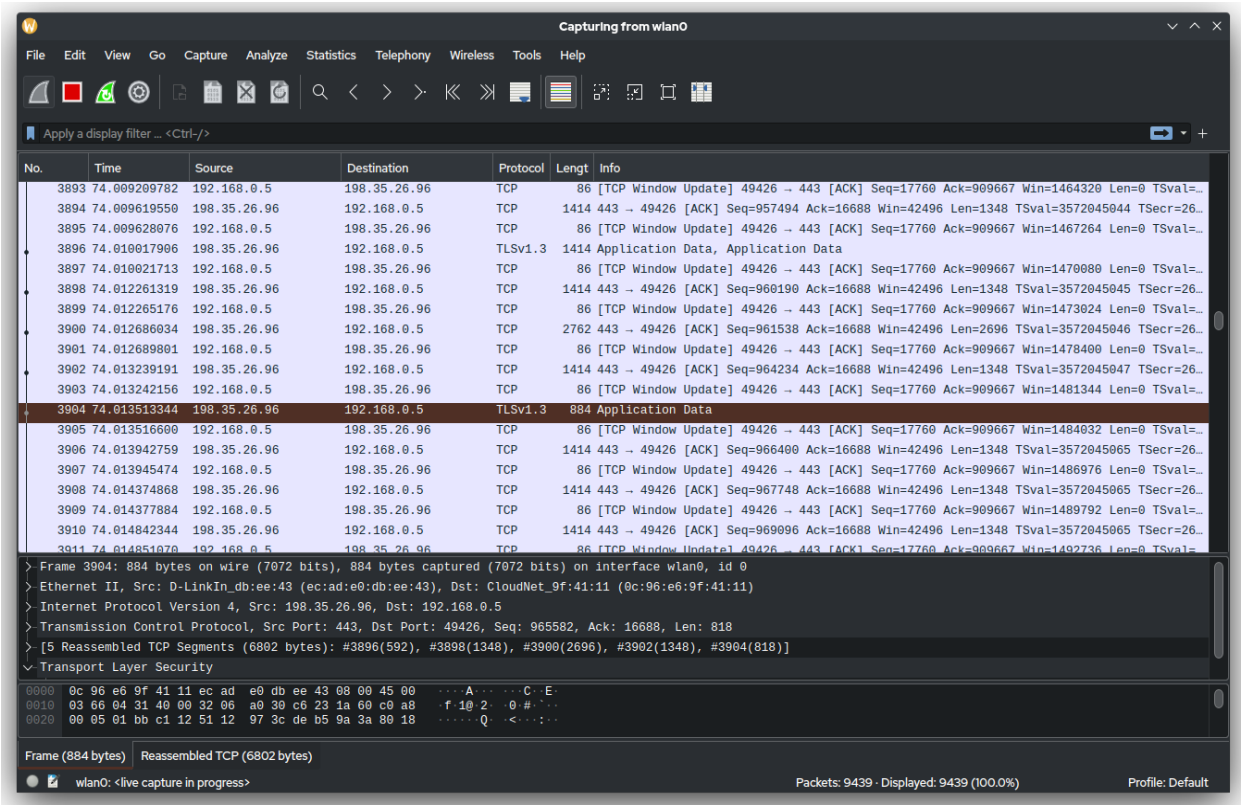


Рисунок 2.3 – Головне вікно програми Wireshark

У панелі зі списком пакетів можна переглядати заголовки пакетів та їх часову мітку. У панелі з деталями пакету можна детально розібрати кожен пакет, що протікає через мережевий інтерфейс. У панелі зі статистикою можна переглядати статистику роботи Wireshark.

Інший метод дослідження, використання тестових сценаріїв. Тестові сценарії дозволяють перевірити різні конфігурації фаєрволу та перевірити, як вони впливають на продуктивність та ефективність мережі. Тестові сценарії також можуть допомогти виявити проблеми з безпекою, які можуть з'являтися при налаштуванні фаєрволу.

Ще одним методом є спостереження. Спостереження являє собою процес систематичного збирання даних про певне явище з метою їх аналізу

та вивчення для подальшого вдосконалення предмету спостерігання. У нашому випадку, можна спостерігати за тим, які сайти відвідують співробітники офісу та який контент переглядають. Це дозволить зрозуміти, який контент можна блокувати з метою підвищення продуктивності роботи співробітників та захисту від потенційно небезпечного або зайвого під час роботи контенту.

Спостереження є важливим методом дослідження в контексті створення фаєрволу для фільтрації контенту в офісі на операційну систему Linux. Для того, щоб збирати дані про відвідування сайтів та перегляд контенту, можна використовувати спеціальні програми-шпигуни, які будуть фіксувати дії користувачів. Завдяки цьому можна дізнатися, які сайти відвідують співробітники, який контент переглядають, який час вони проводять на цих сайтах та як часто вони до них звертаються.

Однак, необхідно пам'ятати про те, що цей метод дослідження може порушувати приватність користувачів та викликати недовіру серед колективу. Тому, перед використанням цього методу, необхідно попередньо погодити його з керівництвом компанії та забезпечити конфіденційність зібраних даних. Для того, щоб зрозуміти, який контент можна блокувати з метою підвищення продуктивності роботи співробітників та захисту від потенційно небезпечного контенту, необхідно аналізувати зібрані дані та зробити висновки щодо їх відповідності політиці безпеки компанії.

Наприклад, якщо виявлено, що співробітники витрачають надто багато часу на соціальні мережі або перегляд відео, то можна заблокувати доступ до цих сайтів з метою підвищення продуктивності роботи. Також можна блокувати сайти, які містять небезпечний контент, наприклад, сайти з порнографією або сайти з вірусними файлами.

2.4 Описання та обґрунтування алгоритмів та результатів дослідження

Створення фаєрволу для операційної системи Linux можна виконати двома шляхами або вписавши покрокові дії в командний рядок операційної системи або програмним кодом через Python.

Створення фаєрволу за допомогою командного рядка Linux - це досить простий інструмент, що не вимагає додаткових витрат на програмне забезпечення. Однак, цей підхід може бути складним для людей, які не мають досвіду в роботі з командним рядком. Такий підхід до створення фаєрволу дозволяє користувачеві налаштувати його за своїм бажанням та необхідними потребами.

Щоб створити фаєрвол в командному рядку Linux, необхідно мати деякі знання про команди терміналу та вміти працювати з текстовими файлами. Першим кроком для створення фаєрволу є перевірка, чи встановлений відповідний пакет iptables, який є найпоширенішим засобом налаштування фаєрволу в Linux. Для цього в терміналі потрібно ввести команду:

```
sudo apt-get install iptables
```

Після успішної установки можна перейти до створення правил для фаєрволу. Для того, щоб створити правило, необхідно знати, який тип пакетів необхідно блокувати. В кожному правилі потрібно вказати певні параметри, наприклад, IP-адресу або порт, який потрібно блокувати. Далі, користувач повинен використовувати команди iptables, які дозволяють додавати, видаляти та переглядати правила.

Команда для додавання правила має наступний синтаксис:

```
iptables -A <chain> <options>
```

Значення chain вказує на тип трафіку, до якого застосовується правило. Наприклад, ланцюги INPUT, OUTPUT та FORWARD відповідають за прийом, відправку та пересилання пакетів відповідно. Опції включають параметри, що визначають тип пакетів, які необхідно блокувати, наприклад, -p для визначення типу протоколу та -s або -d для вказівки IP-адрес.

Наприклад, для блокування доступу до веб-сайту можна використати наступну команду:

```
iptables -A INPUT -p tcp --destination-port 80 -j DROP
```

Ця команда додає правило для блокування доступу до веб-сайту через порт 80.

Після того, як користувач додає правила до фаєрволу, він може перевірити їх, використовуючи команду:

```
iptables -L
```

Ця команда виводить список всіх правил фаєрволу, які зараз застосовуються.

Щоб зберегти правила фаєрволу, потрібно виконати команду:

```
iptables-save
```

Ця команда зберігає всі правила в файл, який зазвичай знаходиться у каталозі /etc/sysconfig/iptables. Якщо виникне потреба відновити фаєрвол пізніше, ці правила можна завантажити з цього файлу, використовуючи команду:

```
iptables-restore
```

Однак, при створенні фаєрволу через командний рядок, може виникнути проблема складності управління правилами. Коли правил стає багато, важко відслідкувати, які саме правила застосовуються. Це може

призвести до помилок в конфігурації фаєрволу, що зробить систему менш захищеною. Також важко автоматизувати процес створення та управління фаєрволом через командний рядок.

Тому, для більш ефективного створення та управління фаєрволом, можна використовувати програмування на мові Python. У програмуванні на Python можна використовувати спеціальну бібліотеку iptables для роботи з фаєрволом. Ця бібліотека дозволяє створювати правила фаєрволу та керувати ними з програми, що написана на Python. Також для відображення інтерфейсу будемо використовувати tkinter є стандартною бібліотекою, що надає інтерфейс для створення графічних користувацьких інтерфейсів (GUI) у Python. Зокрема, tkinter дозволяє створювати вікна, елементи керування, кнопки, поля вводу, текстові мітки та інші елементи GUI для взаємодії користувача з програмою.

Створимо функцію яка буде блокувати контент з заданої користувачем IP-адреси та можливістю додатково ввести порт для блокування. Загальний код цієї функції виглядає наступним чином:

```
import tkinter as tk
import iptc
def block_website():
    ip_address = ip_address_entry.get()
    port = port_entry.get()
    rule = iptc.Rule()
    rule.protocol = "tcp"
    rule.src = ip_address
    if port:
        match = rule.create_match("tcp")
        match.sport = port
        rule.add_match(match)
    rule.target = iptc.Target(rule, "DROP")
```

```
    chain = iptc.Chain(iptc.Table(iptc.Table.FILTER),
"INPUT")
    chain.insert_rule(rule)
root = tk.Tk()
root.title("Block Website")
root.geometry("300x150")
ip_address_label = tk.Label(root, text="Enter IP
Address:")
ip_address_label.pack(pady=5)
ip_address_entry = tk.Entry(root)
ip_address_entry.pack(pady=5)
port_label = tk.Label(root, text="Enter Port
(optional):")
port_label.pack(pady=5)
port_entry = tk.Entry(root)
port_entry.pack(pady=5)
block_button = tk.Button(root, text="Block Website",
command=block_website)
block_button.pack(pady=5)
root.mainloop()
```

Цей код створює простий GUI (графічний інтерфейс користувача) за допомогою бібліотеки tkinter. На GUI є мітка з текстом "Enter IP Address:", поле введення для введення IP-адреси та кнопка з текстом "Block Website". Коли користувач вводить IP-адресу та натискає кнопку, викликається функція block_website(), яка створює нове правило за допомогою бібліотеки iptc. Це правило встановлюється в таблицю iptables, що дозволяє блокувати зазначений IP-адрес. Якщо порт введений, то створюється об'єкт матчу tcp, який встановлює відповідний порт для правила, і виконується блокування для вказаного порту та IP адреси. Якщо порт не введено, то блокується тільки IP адреса.

Ілюстрація цієї функції зображена на рисунку 2.4.

Рисунок 2.4 – Вікно створеного фаєрволу

Щоб дозволити трафік з певної IP-адреси, достатньо змінити один рядок в коді попередньої функції:

```
rule.target = iptc.Target(rule, "ACCEPT")
```

Таким чином замість відкидання інформації, ми дозволяємо її прохід.

Наступним створимо функцію яка буде блокувати контент за ключовими словами, код виглядає наступним чином:

```
def block_keywords(packet):
    if packet.haslayer(DNSQR):
        data = packet.getlayer(DNS).payload.payload.payload
        for keyword in blocked_keywords:
            if keyword.encode('utf-8') in data:
                print(f'Blocked keyword: {keyword}')
                packet.show()
                return
    packet.accept()
```

```
Ця функція призначена для блокування пакетів, які містять ключові слова зі списку blocked_keywords. Функція приймає пакет як параметр packet.
```

Спочатку функція перевіряє, чи містить пакет запит DNS (DNSQR) шар, щоб визначити, чи він містить доменне ім'я, яке потрібно

перевірити.Далі функція отримує дані запиту DNS, які містяться у полі payload внутрішнього шару DNS (DNSRR). Вона перевіряє, чи містить будь-яке з ключових слів зі списку blocked_keywordsca в цих даних. Якщо так, функція виводить повідомлення про блокування пакету з цим ключовим словом та відображає його з допомогою методу packet.show(). Функція повертає управління, не обробивши пакет.

Якщо запит DNS не містить ключових слів, функція повертає управління пакету з допомогою методу packet.accept(). Тобто, пакет пропускається, не блокується і далі може бути оброблений системою.

Код для функції додавання ключових слів:

```
def add_keyword():
    keyword = keyword_entry.get()
    blocked_keywords.append(keyword)
    keyword_list.insert(END, keyword)
```

Ця функція викликається при натисканні на кнопку "Додати ключове слово" і містить наступні дії. Отримання значення з поля введення keyword_entry. Це поле є інтерфейсом, де користувач може ввести ключове слово для блокування. Додавання отриманого значення в список блокованих ключових слів blocked_keywords. Цей список містить ключові слова, які використовуються для порівняння з даними, отриманими з DNS запиту. Відображення доданого ключового слова в інтерфейсі, вікні зі списком ключових слів keyword_list, яке дозволяє користувачу переглянути всі додані ключові слова.

Ця функція розширює можливості інтерфейсу, дозволяючи користувачеві додавати власні ключові слова для блокування. Видалення ключового слова з списку реалізовано наступним чином:

```
def remove_keyword():
    selection = keyword_list.curselection()
    if selection:
```

```
keyword = keyword_list.get(selection[0])
blocked_keywords.remove(keyword)
keyword_list.delete(selection[0])
```

Ця функція призначена для видалення ключового слова зі списку blocked_keywords і зі списку відображення keyword_list, якщо користувач вибрав яке-небудь ключове слово в keyword_list та натиснув кнопку видалення. selection = keyword_list.curselection(): Отримання вибраного елементу у списку keyword_list. Якщо нічого не вибрано, то selection буде None.

if selection:: перевірка наявності вибраного елементу у списку.

keyword = keyword_list.get(selection[0]): отримання значення вибраного елемента зі списку keyword_list.

blocked_keywords.remove(keyword): видалення keyword зі списку blocked_keywords.

keyword_list.delete(selection[0]): видалення вибраного елемента з keyword_list.

Ця частина фаєрволу зображена на рисунку 2.5

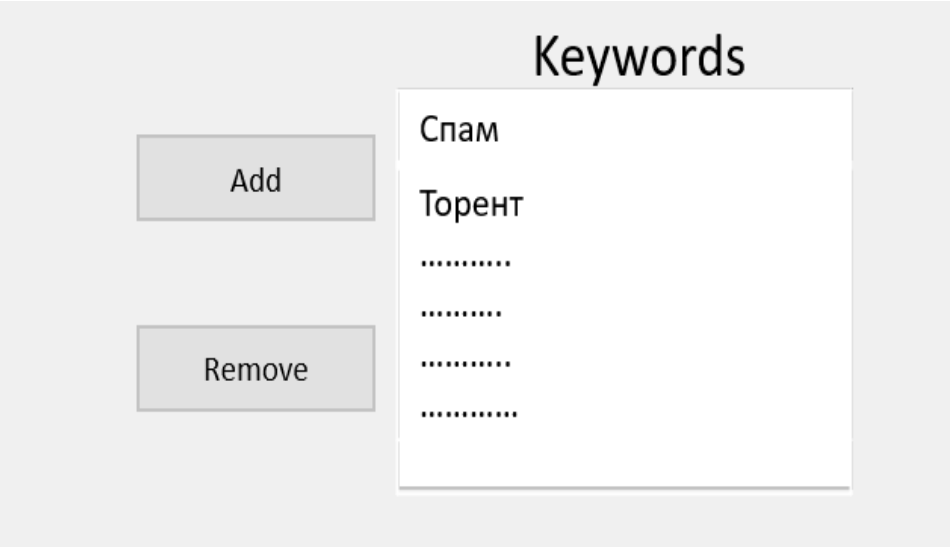


Рисунок 2.5 – Вікно ключових слів

Функціонал блокування за доменним ім’ям схожий по своїй структурі та механізму виконання до блокування за ключовими словами. Тому не має сенсу описувати вже описаний код.

Після завершення кодування та налаштування функціоналу блокування за доменним ім'ям, можна розглянути діаграму розгортання (рис. 2.6), яка зображає архітектуру та взаємозв'язки компонентів фаєрволу. Ця діаграма надає загальний огляд структури системи та взаємодії між її компонентами.

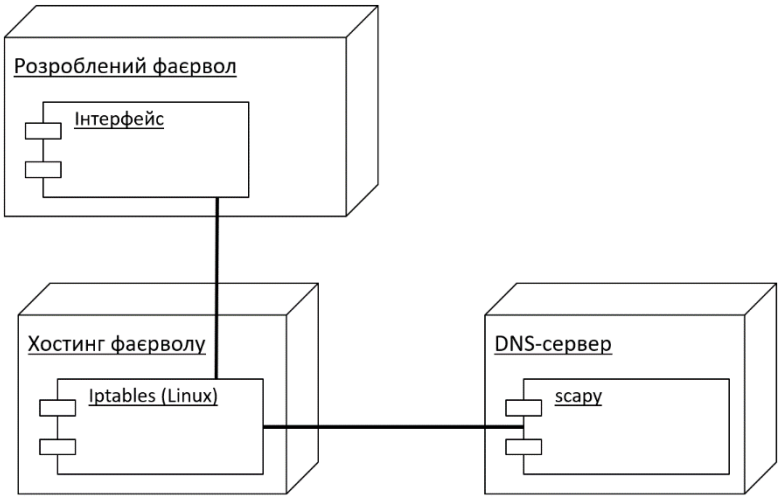


Рисунок 2.6 – Діаграма розгортання фаєрволу

В діаграмі розгортання показано розгортання фаєрволу на сервері для блокування трафіку на основі різних параметрів, таких як IP-адреси, порти, доменні імена та ключові слова. На діаграмі присутні наступні компоненти:

- Розроблений фаєрвол: Цей компонент є системою користувача, на якій запускається клієнтський додаток або інтерфейс для управління фаєрволом. Він надає можливість налаштування правил блокування трафіку та моніторингу активності;
- Хостинг фаєрволу: Цей компонент є сервером, на якому розташована програма фаєрволу. Використовується модуль iptables для налаштування правил фаєрволу та блокування трафіку на основі заданих

параметрів. Хостинг фаєрволу може бути фізичним сервером або віртуальною машиною;

- DNS-сервер: Цей компонент є сервером DNS, який використовує модуль scapy для аналізу мережевого трафіку і перехоплення DNS-запитів. Фаєрвол використовує цей DNS-сервер для блокування доменних імен або ключових слів, які вказуються в програмі фаєрволу;

Ці компоненти співпрацюють між собою, щоб забезпечити функціонал блокування трафіку на основі заданих параметрів. Клієнтський додаток та інтерфейс на системі користувача дозволяють налаштовувати правила фаєрволу, які потім передаються на сервер фаєрволу для застосування. Фаєрвол, в свою чергу, використовує модуль iptables для налаштування правил блокування трафіку, а також спирається на DNS-сервер для перехоплення та блокування доменних імен, IP-адрес, ключових слів, портів, які заборонені або потребують блокування та приймає рішення щодо перекриття доступу до таких ресурсів.

2.5 Аналіз результатів

Захист комп'ютерних систем від несанкціонованого доступу та збереження конфіденційної інформації є важливою задачею кожної організації, яка працює з електронними даними. Після тестування фаєрволу для обмеження контенту в офісних мережах було проведено аналіз результатів роботи. У результаті спостережень було виявлено декілька позитивних та негативних аспектів роботи фаєрволу.

Один з позитивних аспектів полягає в тому, що фаєрвол успішно блокує доступ до веб-сайтів, які містять ключові слова, зазначені у списку обмежень. Блокування контенту відбувається без затримок, а результати роботи фаєрволу в цьому плані є надійними та ефективними. Перевагою фаєрволу можна вважати ще й те, що він забезпечує збереження конфіденційної

інформації, оскільки блокує доступ до веб-сайтів, які містять потенційно небезпечну або небажану інформацію. Наприклад, фаєрвол може блокувати доступ до веб-сайтів, які містять шкідливий код або містять віруси. Так розроблений фаєрвол додає можливість налаштування на рівні користувача. Це дозволяє заблокувати доступ до конкретних веб-сайтів або додатків лише для певних співробітників, що може бути корисно при вирішенні питань безпеки та зменшення ризиків.

Негативним аспектом роботи фаєрволу стало те, що при великій кількості обмежень на веб-контент, фаєрвол може затримувати роботу мережі. Наприклад, при блокуванні доступу до веб-сайтів з великою кількістю фото- та відео-матеріалів, може виникати затримка завантаження сторінок, що може вплинути на робочий процес та знизити продуктивність працівників. Також є певна особливість, яка полягає у тому, що фаєрвол може блокувати доступ до веб-сайтів, які не містять заборонених ключових слів. Це стається у випадках, коли вміст веб-сторінки заборонений за іншими критеріями. Тому необхідно налаштовувати фаєрвол з розумінням цієї особливості та враховувати її при використанні.

3.ВИСНОВКИ

В результаті виконання роботи було проаналізовано поняття фаєрволу архітектуру та принципи роботи. Розглянуто аналогічні системи виявленої їх переваги та недоліки. Що допомогло в формуванні деяких вимог щодо розробленого фаєрволу для фільтрації контенту в офісних мережах для операційної системи Linux. Після даного етапу було сформовано мету та поставлено необхідні задачі для розробки фаєрволу. Обрано методи дослідження для розробки, проаналізовано вже наявні результати в області дослідження. Після чого було описано обрані алгоритми вирішення недоліків існуючих систем та описання ключового функціоналу Під час тестування було перевірено функціональність фаєрволу та його здатність ефективно блокувати небезпечний контент. фаєрволу. Результати розробки показали, що фаєрвол успішно виконує свої функції з обмеження контенту в офісних мережах. Він ефективно блокує доступ до веб-сайтів, що містять небезпечний контент, такий як віруси та шкідливі програми. Крім того, фаєрвол є досить простим у використанні та налаштуванні, що зменшує ймовірність помилок під час експлуатації. Проаналізовано результати розробки та наведено висновки що до них.

4. СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Міжмережевий екран [Електронний ресурс] – Режим доступу до ресурсу: https://sites.google.com/site/zahistlokalnoiemerezi/zahist/mizmerezevij-ekran

2. 30 SURPRISING SOCIAL MEDIA AT WORK STATISTICS [2023]: WHAT EVERY MANAGER SHOULD KNOW [Електронний ресурс] – Режим доступу до ресурсу: https://www.zippia.com/advice/social-media-at-work-statistics/#:~:text=The%20average%20employee%20spends%2012,per%20day%20on%20social%20media

3. Data snapshot: How web filtering affects workplace security and productivity [Електронний ресурс] – Режим доступу до ресурсу: https://community.spiceworks.com/blog/3073-data-snapshot-how-web-filtering-affects-workplace-security-and-productivity

4. Що таке Firewall? [Електронний ресурс] – Режим доступу до ресурсу: https://2ip.ua/ua/blog/firewall

5. Security - Firewall [Електронний ресурс] – Режим доступу до ресурсу: https://ubuntu.com/server/docs/security-firewall

6. Best firewall for Linux [Електронний ресурс] – Режим доступу до ресурсу: https://linuxconfig.org/best-firewall-for-linux

7. Iptables Tutorial: Ultimate Guide to Linux Firewall [Електронний ресурс] – Режим доступу до ресурсу: https://phoenixnap.com/kb/iptables-tutorial-linux-firewall

8. Iptables Tutorial – Securing Ubuntu VPS with Linux Firewall [Електронний ресурс] – Режим доступу до ресурсу: https://www.hostinger.com/tutorials/iptables-tutorial

9. Iptables [Електронний ресурс] – Режим доступу до ресурсу: https://uk.wikipedia.org/wiki/Iptables

10. Посібник для початківців для iptables, брандмауер Linux [Електронний ресурс] – Режим доступу до ресурсу: https://ua.phhsnews.com/articles/howto/the-beginners-guide-to-iptables-the-linux-firewall.html

11. Top 10 Firewall Hardware Devices in 2022 [Електронний ресурс] – Режим доступу до ресурсу: https://www.spiceworks.com/it-security/network-security/articles/top-10-firewall-hardware-devices/

12. What are Network Firewalls? [Електронний ресурс] – Режим доступу до ресурсу: https://www.gartner.com/reviews/market/network-firewalls

13. БРАНДМАУЕР ЯК ОСНОВНИЙ ЗАХИСТ СИСТЕМ ІНФОРМАЦІЙНОГО І КІБЕРНЕТИЧНОГО НАПРЯМУ. ПЕРЕВАГИ ТА НЕДОЛІКИ [Електронний ресурс] – Режим доступу до ресурсу: https://dut.edu.ua/ua/news-1-569-8737-brandmauer-yak-osnovniy-zahist-sistem-informaciynogo-i-kibernetichnogo-napryamu-perevagi-ta-nedoliki_kafedra-cistem-tehnichnogo-zahistu-informacii

14. Технології комп’ютерної безпеки [Електронний ресурс] – Режим доступу до ресурсу: https://core.ac.uk/download/pdf/14052345.pdf

15. About python-iptables [Електронний ресурс] – Режим доступу до ресурсу: https://python-iptables.readthedocs.io/en/latest/intro.html

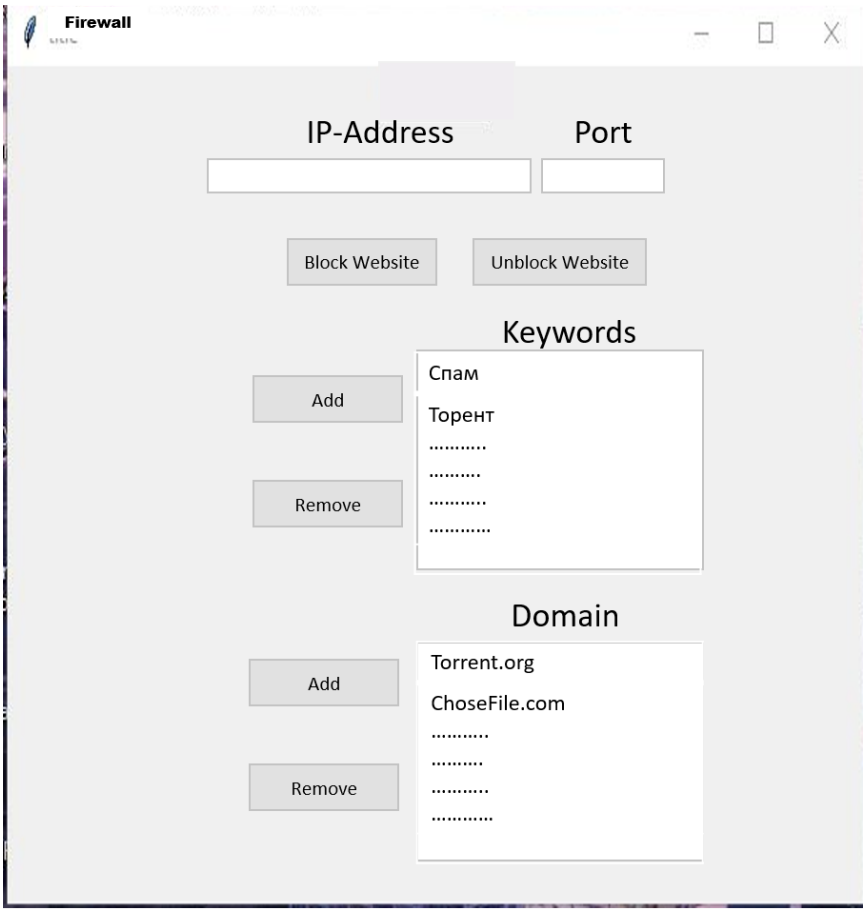
16. How to write specific iptables rules using python-iptables [Електронний ресурс] – Режим доступу до ресурсу: https://stackoverflow.com/questions/20734319/how-to-write-specific-iptables-rules-using-python-iptables

17. python-iptables 1.0.1 [Електронний ресурс] – Режим доступу до ресурсу: https://pypi.org/project/python-iptables/

5. ДОДАТКИ

Додаток А

Згальний вигляд розробленого фаєрволу



Додаток Б

Код функцій блокування за IP-адресою та портом:

```
import tkinter as tk

import iptc

def block_website():

    ip_address = ip_address_entry.get()
```

```
    port = port_entry.get()
    rule = iptc.Rule()
    rule.protocol = "tcp"
    rule.src = ip_address
    if port:
        match = rule.create_match("tcp")
        match.sport = port
        rule.add_match(match)
    rule.target = iptc.Target(rule, "DROP")
    chain = iptc.Chain(iptc.Table(iptc.Table.FILTER), "INPUT")
    chain.insert_rule(rule)
root = tk.Tk()
root.title("Block Website")
root.geometry("300x150")
ip_address_label = tk.Label(root, text="Enter IP Address:")
ip_address_label.pack(pady=5)
ip_address_entry = tk.Entry(root)
ip_address_entry.pack(pady=5)
port_label = tk.Label(root, text="Enter Port (optional):")
port_label.pack(pady=5)
port_entry = tk.Entry(root)
port_entry.pack(pady=5)
```

```
block_button = tk.Button(root, text="Block Website", command=block_website)
block_button.pack(pady=5)
root.mainloop()
```

Додаток В

Код для функцій блокування за ключовими словами та доменним іменем:

```
import os
from scapy.all import *
import tkinter as tk
blocked_domains = []
blocked_keywords = []
def packet_callback(packet):
    if packet.haslayer(DNSQR):
        domain = packet[DNSQR].qname.decode('utf-8')
        if domain in blocked_domains:
            print(f'Blocked domain: {domain}')
            packet.show()
            return
        data = packet.getlayer(DNS).payload.payload.payload
        for keyword in blocked_keywords:
            if keyword.encode('utf-8') in data:
                print(f'Blocked keyword: {keyword}')
```

```
            packet.show()
            return
    def add_domain():
        domain = domain_entry.get()
        if domain not in blocked_domains:
            blocked_domains.append(domain)
            domain_list.insert(tk.END, domain)
    def remove_domain():
        selection = domain_list.curselection()
        if len(selection) == 1:
            domain = domain_list.get(selection[0])
            blocked_domains.remove(domain)
            domain_list.delete(selection)
    def add_keyword():
        keyword = keyword_entry.get()
        if keyword not in blocked_keywords:
            blocked_keywords.append(keyword)
            keyword_list.insert(tk.END, keyword)
    def remove_keyword():
        selection = keyword_list.curselection()
        if len(selection) == 1:
            keyword = keyword_list.get(selection[0])
            blocked_keywords.remove(keyword)
```

```
        keyword_list.delete(selection)

    # Create GUI
    root = tk.Tk()
    root.title("DNS Blocker")

    # Domain section
    domain_frame = tk.Frame(root)
    domain_frame.pack(side=tk.LEFT, padx=10, pady=10)

    domain_label = tk.Label(domain_frame, text="Blocked Domains:")
    domain_label.pack()

    domain_list = tk.Listbox(domain_frame, height=10, width=30)
    domain_list.pack()

    domain_entry = tk.Entry(domain_frame, width=30)
    domain_entry.pack(pady=5)

    add_domain_button = tk.Button(domain_frame, text="Add Domain", command=add_domain)
    add_domain_button.pack(side=tk.LEFT, padx=5)

    remove_domain_button = tk.Button(domain_frame, text="Remove Domain", command=remove_domain)
    remove_domain_button.pack(side=tk.LEFT, padx=5)

    # Keyword section
    keyword_frame = tk.Frame(root)
    keyword_frame.pack(side=tk.LEFT, padx=10, pady=10)

    keyword_label = tk.Label(keyword_frame, text="Blocked Keywords:")
```

```
keyword_label.pack()

keyword_list = tk.Listbox(keyword_frame, height=10, width=30)

keyword_list.pack()

keyword_entry = tk.Entry(keyword_frame, width=30)

keyword_entry.pack(pady=5)

add_keyword_button = tk.Button(keyword_frame, text="Add Keyword", command=add_keyword)

add_keyword_button.pack(side=tk.LEFT, padx=5)

remove_keyword_button = tk.Button(keyword_frame, text="Remove Keyword", command=remove_keyword)

remove_keyword_button.pack(side=tk.LEFT, padx=5)

# Start packet capture

os.system('iptables -A INPUT -p udp --dport 53 -j NFQUEUE')

os.system('iptables -A OUTPUT -p udp --sport 53 -j NFQUEUE')

nfqueue = NetfilterQueue()

nfqueue.bind(0, packet_callback)

try:

    nfqueue.run()

except KeyboardInterrupt:

    os.system('iptables -F')

    os.system('iptables -X')

# Start GUI event loop

root.mainloop()
```

6. АНОТАЦІЯ

Кваліфікаційна робота за темою «розробка файерволу для фільтрації контенту в мережі інтернет» змістовно складається з двох розділів.

В першому формується мета та основні задачі дослідження. Аналізуються основні відомості щодо предметної області.

В другому розділі розробляються конкретні задачі які необхідно вирішити в ході виконання роботи. Розглядаються аналоги фаєрволів, описуються їх ключові переваги та недоліки. Встановлюються методи досліджень та алгоритми за якими буде проводитись дослідження та безпосередня розробка. Аналізується результат дослідження та проводяться висновки щодо до предмету розробки.

6.1 ABSTRACT

The qualification work on the topic "development of a firewall for content filtering on the Internet" meaningfully consists of two sections.

In the first, the purpose and main tasks of the research are formed. Basic information about the subject area is analyzed.

In the second section, specific tasks are developed that must be solved in the course of the work. Analogues of firewalls are considered, their key advantages and disadvantages are described. Research methods and algorithms are established, according to which research and direct development will be carried out. The result of the research is analyzed and conclusions are made regarding the subject of development.