МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Тернопільський національний економічний університет
Факультет комп’ютерних інформаційних технологій
Кафедра комп'ютерної інженерії

Троць Ігор Іванович

Засіб фільтрації трафіку на основі аналізу журналів доступу до сервісів системою Fail2Ban / Traffic filtering tool based on log files accessing services analysis using Fail2Ban

напрям підготовки: 6.050102 - Комп’ютерна інженерія
фахове спрямування - Комп’ютерні системи та мережі
Бакалаврська робота

Виконав студент групи КСМ-42/1
Троць І.І.

Науковий керівник: к.т.н.,
Возняк С.І.

Тернопіль - 2018

РЕЗЮМЕ

Дипломний проект містить 52 сторінок пояснюючої записки, 3 рисунки, 7 таблиць, 2 додатки. Обсяг графічного матеріалу 2 аркуші формату А3.

Метою даного дипломного проекту є розробка засобу фільтрації трафіку на основі аналізу журналів доступу до сервісів.

Оскільки програмні фаєрволи не дають можливості налаштувати усі параметри, та використовують значну частину ресурсів сервера, а апаратні рішення є дуже дорогими у використанні та є не рентабельними у моїй мережі.

Для управління усіма конфігураціями сервісу Fail2Ban був розроблений web-інтрефейс, який спростить внесення змін у параметри правил. Web-інтрефейс дозволив реалізувати такі функції:

- видалення та додавання нових збійних виправлень;
- додавати заборонені ip-адреси для блокування та розблокування ip-адреси;
- налаштувати часу пошуку, та перегляду списку файлів що заблоковані;
- сигналізацію для сповіщення, коли ip-адресу заблоковано та не заблоковано, за допомогою налаштувань інтервалу часу;
- тестування регулярних викликів для тестування, ігнорування та повторного виправлення помилок у поточних журналах, щоб швидко створювати та налаштовувати регулярні виправлення;
- звітностті коли IP-адреса заблокована, і показувати тенденції за допомогою візуалізації.

Ключові слова: ФАЄРВОЛ, ТРАФІК, ЗАХИСТ ІНФОРМАЦІЇ, FAIL2BAN.

RESUME

The diploma project contains 52 pages of explanatory note, 3 figures, 7 tables, 2 appendices. Volume of graphic material 2 sheets of A3 format.

The purpose of this diploma project is to develop a means of traffic filtering based on the analysis of logs of access to services.

Because software firewalls do not allow you to configure all the settings, and use a significant portion of server resources, and hardware solutions are very expensive to use and are not cost effective in my network.

To manage all configurations of the Fail2Ban service, a web interface has been developed that will simplify changes to rule parameters. The web-interface allowed to implement the following functions:

- removing and adding new bug fixes;

- add forbidden ip-addresses to block and unblock ip-addresses;

- set the search time and view the list of locked files;

- alarm for notification when the ip-address is blocked and not blocked, using the time interval settings;

- Test regular calls to test, ignore, and re-fix bugs in current logs to quickly create and configure regular fixes.

- Reporting when an IP address is blocked, and show trends through visualization.

Keywords: FIREWORK, TRAFFIC, INFORMATION PROTECTION, FAIL2BAN.

ЗМІСТ

ВСТУП

Інтенсивний розвиток глобальних комп'ютерних мереж, поява нових технологій пошуку інформації привертають все більше уваги до мережі Internet з боку приватних осіб і різних організацій. Багато організацій приймають рішення про інтеграцію своїх локальних і корпоративних мереж в глобальну мережу. Використання глобальних мереж у комерційних цілях, а також при передачі інформації, яка містить відомості конфіденційного характеру, тягне за собою необхідність побудови ефективної системи захисту інформації. В даний час в Україні глобальні мережі застосовуються для передачі комерційної інформації різного рівня конфіденційності, наприклад для зв'язку з віддаленими офісами з головної штаб квартири організації або створення Web-сторінки організації з розміщеною на ній рекламою і діловими пропозиціями.

Навряд чи потрібно перераховувати всі переваги, які отримує сучасне підприємство, маючи доступ до глобальної мережі Internet. Але, як і багато інших нові технології, використання Internet має і негативні наслідки. Розвиток глобальних мереж призвело до багаторазового збільшення кількості користувачів і збільшення кількості атак на комп'ютери, підключені до мережі Internet. Щорічні втрати, зумовлені недостатнім рівнем захищеності комп'ютерів, оцінюються десятками мільйонів доларів. При підключенні до Internet локальної або корпоративної мережі необхідно подбати про забезпечення інформаційної безпеки цієї мережі. Глобальна мережа Internet створювалася як відкрита система, призначена для вільного обміну інформацією. У силу відкритості своєї ідеології Internet надає для зловмисників значно більші можливості в порівнянні з традиційними інформаційними системами. Тому питання про проблему захисту мереж і її компонентів стали досить важливими та актуальними в цей час, час прогресу комп'ютерних технологій.

1 АНАЛІЗ ТА ПОСТАНОВКА ЗАВДАННЯ

1.1 Огляд існуючих фаєрволів

Для того щоб задовільнити вимоги широкого кола користувачів, існує три типи фаєрволів: мережного рівня, прикладного рівня та рівня з'єднання. Кожен з цих трьох типів використовує свій, відмінний від інших, підхід до захисту мережі. Фаєрвол мережного рівня представлений екрануючим маршрутизатором. Він контролює лише дані мережевого і транспортного рівнів службової інформації пакетів. Мінусом таких маршрутизаторів є те, що ще п'ять рівнів залишаються неконтрольованими.

Адміністратори, які працюють з екрануючими маршрутизаторами, повинні пам'ятати, що у більшості приладів, що здійснюють фільтрацію пакетів, відсутні механізми аудиту та подачі сигналу тривоги. Іншими словами, маршрутизатори можуть піддаватися атакам і відбивати велику їх кількість, а адміністратори навіть не будуть проінформовані [1].

Фаєрвол прикладного рівня також відомий як проксі-сервер. Фаєрволи прикладного рівня встановлюють певний фізичний поділ між локальною мережею і Internet, тому вони відповідають найвищим вимогам безпеки.

Проте, оскільки програма повинна аналізувати пакети і приймати рішення щодо контролю доступу до них, фаєрволи прикладного рівня неминуче зменшують продуктивність мережі, тому в якості сервера-посередника використовуються більш швидкі комп'ютери.

Фаєрвол рівня з'єднання схожий на фаєрвол прикладного рівня тим, що обидва вони є серверами-посередниками. Відмінність полягає в тому, що фаєрволи прикладного рівня вимагають спеціального програмного забезпечення для кожної мережевої служби на зразок FTP або HTTP. Натомість, фаєрволи рівня з'єднання обслуговують велику кількість протоколів [2].

Брандмауери з пакетними фільтрами приймають рішення про те, пропускати пакет або відкинути, переглядаючи IP-адреси, прапори або номер TCP портів в заголовку цього пакету.

IP-адреса та номер порту - це інформація мережевого і транспортного рівнів відповідно, але пакетні фільтри використовують і інформацію прикладного рівня, тому що всі стандартні сервіси в TCP/IP асоціюються з певним номером порту [3].

Для опису правил проходження пакетів складаються таблиці з певними правилами які наведені у таблиці 1.1.

Таблиця 1.1 – Опис правил проходження пакетів

Дія	1	2
тип пакету		
адреса джерела		
порт джерела		
прапори		
адреса призначення		

Поле "дія" може приймати значення пропустити або відкинути, тип пакета - TCP, UDP чи ICMP. Прапори - прапори із заголовка IP-пакета. Поля "порт джерела" і "порт призначення" мають сенс тільки для TCP і UDP пакетів.

Брандмауери з серверами прикладного рівня використовують сервера конкретних сервісів - TELNET, FTP, що запускаються на брандмауері і пропускають через себе весь трафік, що відноситься до даного сервісу. Таким чином, між клієнтом і сервером утворюються два з'єднання: від клієнта до брандмауера і від брандмауера до місця призначення.

Повний набір підтримуваних серверів розрізняється для кожного конкретного брандмауера, однак найчастіше зустрічаються сервера для наступних сервісів:

- термінали;
- передача файлів;
- електронна пошта;
- http;
- rsh;
- nntp.

Використання серверів прикладного рівня дозволяє вирішити важливе завдання - приховати від зовнішніх користувачів структуру локальної мережі, включаючи інформацію в заголовках поштових пакетів або служби доменних імен (DNS). Іншим позитивним моментом є можливість аутентифікації на рівні користувача (аутентифікація - процес підтвердження ідентичності; в даному випадку це процес підтвердження, чи справді користувач є тим, за кого він себе видає). Трохи докладніше про аутентифікацію буде сказано нижче [4].

При описі правил доступу використовуються такі параметри як: назва сервісу, ім'я користувача, допустимий часовий діапазон використання сервісу, комп'ютери, з яких можна користуватися сервісом, схеми аутентифікації. Сервера протоколів прикладного рівня дозволяють забезпечити найбільш високий рівень захисту - взаємодія із зовнішнім світом реалізується через невелику кількість прикладних програм, що повністю контролюють весь вхідний і вихідний трафік.

Сервер рівня з'єднання являє собою транслятор TCP з'єднання. Користувач утворює з'єднання з певним портом на брандмауері, після чого останній встановлює з'єднання з місцем призначення по інший бік від брандмауера. Під час сеансу цей транслятор копіює байти в обох напрямках, діючи як дріт. Як правило, пункт призначення задається заздалегідь, в той час як джерел може бути багато (з'єднання типу один - багато). Використовуючи різні порти, можна створювати різні конфігурації.

Такий тип сервера дозволяє створювати транслятор для будь-якого визначеного користувачем сервісу, що базується на TCP, здійснювати контроль доступу до цього сервісу, збір статистики щодо його використання [5].

Нижче наведені основні переваги і недоліки пакетних фільтрів і серверів прикладного рівня.

До позитивних якостей пакетних фільтрів слід віднести наступні:

- відносно невисока вартість;
- гнучкість у визначенні правил фільтрації;
- невелика затримка при проходженні пакетів.

Недоліки у даного типу брандмауерів наступні:

- локальну мережу видно (маршрут) з INTERNET;
- правила фільтрації пакетів важкі в описі, потрібні дуже хороші знання технологій TCP і UDP;
- при порушенні працездатності брандмауера всі комп'ютери стають повністю незахищеними або недоступними;
- аутентифікацію з використанням IP-адреси можна обдурити використанням IP-спуфінга (атакуюча система видає себе за іншу, використовуючи її IP-адресу);
- відсутність аутентифікації на рівні користувача.

До переваг серверів прикладного рівня слід віднести наступні:

- локальна мережа невидима з INTERNET;
- при порушенні працездатності брандмауера пакети перестають проходити через брандмауер, тим самим не виникає загрози для захищеності ним машин;
- захист на рівні додатків дозволяє здійснювати велику кількість додаткових перевірок, знижуючи тим самим можливість злому з використанням дірок у програмному забезпеченні;

- аутентифікація на призначеному для користувача рівні може бути реалізована системою негайного попередження про спробу злому.

Недоліками цього типу є:

- вища, ніж для пакетних фільтрів вартість;
- неможливість використання протоколів RPC і UDP;
- продуктивність нижча, ніж для пакетних фільтрів.

Також потрібно розглянути апаратні фаєрволи від відомих та провідних компаній які являються лідерами у даній сфері. Одним з прикладів таких апаратних фаєрволів є рішення від компанії Cisco. Візьмемо для прикладу модель 55xx серії. На рисунку 1.1 представлено усю лінійку фаєрволів 55xx серії, починаючи від ASA 5506-X моделі, та закінчуючи ASA 5555-X моделлю [6].

Рисунок 1.1 – Апаратні фаєрволи Cisco ASA серії 55xx

Серія ASA базуються на процесорах x86. Починаючи з версії 7.0 ASA використовують однакові образи операційної системи, але функціональність залежить від того, на якому пристрої вона запущена. Управляти пристроєм можна через telnet, SSH, веб-інтерфейс або за допомогою програми ASDM. Функціональність залежить від типу ліцензії, яка визначається введеним серійним номером. ASA - це уніфікований пристрій керування загрозами, який об'єднує декілька функцій безпеки мережі в одній коробці [7].

Також одним з конкурентнозданих рішень є рішення від компанії Juniper, а саме моделі SRX серії. Juniper SRX - універсальна комплексна система безпеки

мережі, яка здатна замінити кілька пристроїв, що використовуються для підключень до мережі передачі даних. Серія пристроїв SRX має всі можливості маршрутизатора, брандмауера, пристрої UTM, IPS і концентратора VPN. Лінійка представлена великим вибором моделей, від компактних платформ з фіксованою конфігурацією (напр. SRX100 і ін.) Для невеликих мереж і до модульних платформ (напр. SRX5000 і ін.) С високою щільністю портів для великих підприємств. Працює вся лінійка міжмережевих екранів Juniper SRX під управлінням ОС Junos.

Фаєрвол SRX компанії Juniper Networks є відмінним рішенням для корпоративних мереж [8]. Як брандмауера для малого підприємства найкраще підійдуть моделі SRX100 і SRX240. На рисунку 1.2 представлена уся лінійка фаєрволів SRX серії.

Рисунок 1.2 – Апаратні фаєрволи Juniper серії SRX

Великі підприємства, провайдери з великою пропускною здатністю в якості системи безпеки можуть вибрати SRX650, SRX3000, SRX5000. Міжмережеві екрани Juniper SRX є більш якісними і економічними аналогами серії Cisco ASA.

Також потрібно розглянути апаратні рішення від компанії D-Link, а саме NetDefend серії представлених на рисунку 1.3.

Рисунок 1.3 – Апаратні фаєрволи D-Link серії NetDefend

Міжмережеві екрани NetDefend UTM оснащені системою виявлення і запобігання вторгнень, антивірусом, фільтрацією Web-вмісту і контролем додатків. Використовуваний в даних пристроях апаратний прискорювач збільшує продуктивність IPS і AV, керуючої бази контролю додатків і пошуку в Web, що містить мільйони URL для фільтрації Web-вмісту (WCF). Сервіси поновлення IPS, антивіруса і бази даних URL захищають офісну мережу від вторгнень, черв'яків, шкідливих кодів і задовольняють потребам бізнесу з управління доступом співробітників до Інтернет. Міжмережеві екрани NetDefend UTM використовують унікальну технологію IPS - компонентні сигнатури, які дозволяють розпізнавати і забезпечувати захист, як проти відомих, так і проти невідомих атак [9].

В результаті, дані пристрої допомагають при атаках значно знизити вплив на такі важливі аспекти, як корисне навантаження, закрита інформація, а також запобігти поширенню інфекцій та комп'ютерні вторгнення. База даних IPS включає інформацію про глобальні атаках і вторгнення, зібрану на публічних сайтах. Міжмережеві екрани NetDefend UTM забезпечують високу ефективність сигнатур IPS, постійно створюючи й оптимізуючи сигнатури NetDefend через D-Link Auto-Signature Sensor System.

Також, можна налаштувати динамічний рівномірний розподіл смуги пропускання між користувачами, що б хтось із них не захопив всю пропускну смугу каналу. Приклад на рисунку 1.4.

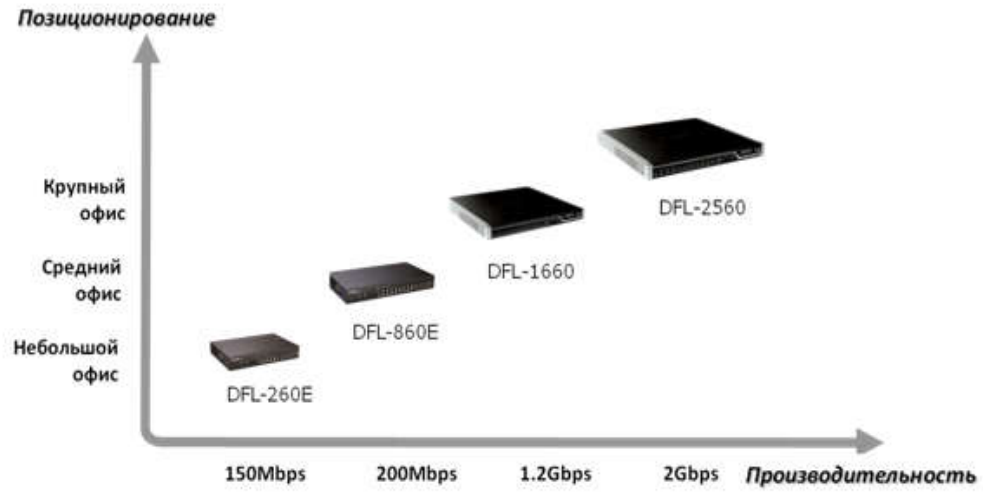


Рисунок 1.4 – Динамічний рівномірний розподіл смуги пропускання

1.2 Аналіз фільтрації трафіку

Фільтрація вхідних пакетів - у невеликих локальних мережах при настроюванні брандмауерів основна увага звичайна приділяється вхідному ланцюжку, зв'язаної з зовнішнім мережним інтерфейсом. Як було сказано вище, при фільтрації пакетів враховуються адреса джерела, адреса призначення, порт джерела, порт призначення і прапори, що визначають стан TCP-з'єднання. У наступних розділах докладно розглядаються дані, що можуть міститися в зазначених полях і рішення, прийняті брандмауером [10]. Графічне представлення алгоритму фільтрації трафіку зображено ДП.КСМ.07144/14.00.00.000 А2.

Фільтрація на основі адреси джерела - на рівні фільтрації пакетів єдиний спосіб ідентифікації відправника – перевірка IP-адреси джерела в заголовку пакета. Обмежені можливості брандмауера надають широкі можливості для фальсифікації пакетів, при якій відправник заміняє свою адресу в заголовку пакета іншим значенням. Для підміни може бути обрана неіснуюча чи реальна адреса, що належить іншому вузлу. Це дозволяє зловмиснику незаконно проникнути у вашу систему чи атакувати інші вузли, ховаючись під вашим ім'ям. При цьому спроби простежити джерело повідомлень направляються по помилковому сліді.

Фільтрація на основі адреси призначення - у більшості випадків фільтрація на основі адреси призначення виконується автоматично. Мережний інтерфейс просто ігнорує пакети, не адресовані безпосередньо йому. Виключенням є широкомовні пакети, адресовані усім вузлам мережі. Адреса 255.255.255.255 являє собою загальну широкомовну адресу. Ви можете визначити широкомовну адресу для конкретної мережі, додавши до номера мережі необхідну кількість десяткових чисел 255. Припустимо, наприклад, що номер мережі вашого провайдера 192.168.0.0, а ваша IP-адреса 192.168.10.30. У цьому випадку широкомовна адреса буде мати вид 192.168.255.255 або 255.255.255.255. Широкомовний пакет, спрямований за адресою 0.0.0.0, безсумнівно фальсифікований. Ціль передачі такого пакета — ідентифікувати версію UNIX. У відповідь на такий пакет система UNIX версії BSD передає ICMP-повідомлення про помилку з кодом 3.

Приведений приклад може служити додатковим аргументом при виборі між забороною (deny) і відмовленням у проходженні (reject) пакета. У даному випадку повідомлення про помилку і є та інформація про систему, що прагне одержати зломщик [11].

Фільтрація на основі порту джерела - номер порту джерела, що міститься в заголовку пакета, призначений для ідентифікації програми-відправника повідомлення, що виконується на вилученому вузлі. У запитах вилучених клієнтів вашому серверу містяться різні номери портів, а у відповідях сервера клієнтам — той самий порт. У складі запиту вилученого клієнта вказується непривілейований порт. Так, наприклад, номер порту в запиті до Web-сервера повинний лежати в діапазоні від 1024 до 65535. У відповіді вилученого сервера повинний бути зазначений порт, виділений для конкретної мережної служби. Якщо ви звернулися до вилученого Web-сервера, то в його відповіді буде міститися номер вихідного порту, рівний 80. Цей порт використовується HTTP-серверами [12].

Фільтрація на основі порту призначення - порт призначення визначає програму на вашому комп’ютері, який призначений пакет. У запитах вилучених клієнтів, переданих на сервер, міститься той самий порт призначення, а у відповідях сервера клієнтам – різні номери портів.

У пакеті, що містить звертання клієнта до сервера, знаходиться номер порту, виділений для забезпечення роботи конкретного типу мережної служби. Так, наприклад, пакет, спрямований Web-серверу, містить номер порту 80. У відповідях вилучених серверів на запити клієнта міститься непривілейований номер порту в діапазоні від 1024 до 65535.

Фільтрація на основі інформації про стан TCP-з'єднання - у деяких правилах обробки пакетів використовуються прапори, що визначають стан TCP-з'єднання. Будь-яке з'єднання проходить через визначені стани. Стани клієнта і сервера розрізняються між собою. У першому пакеті, відправленому вилученим клієнтом, установлений прапор SYN (прапор АСК скинутий). Передача такого пакета є першим кроком у встановленні TCP-з'єднання. В усіх наступних пакетах, переданих клієнтом, установлений прапор АСК, а прапор SYN скинутий [12].

Як правило, брандмауери дозволяють проходження пакетів, що містять звертання клієнтів, незалежно від стану прапорів SYN і АСК. Пакети, передані вилученими серверами, завжди є відповідями на попередні звертання клієнтів-програм. У кожнім пакеті,, що надійшов від вилученого сервера, повинний бути встановлений прапор АСК, оскільки TCP-з'єднання ніколи не встановлюється з ініціативи сервера.

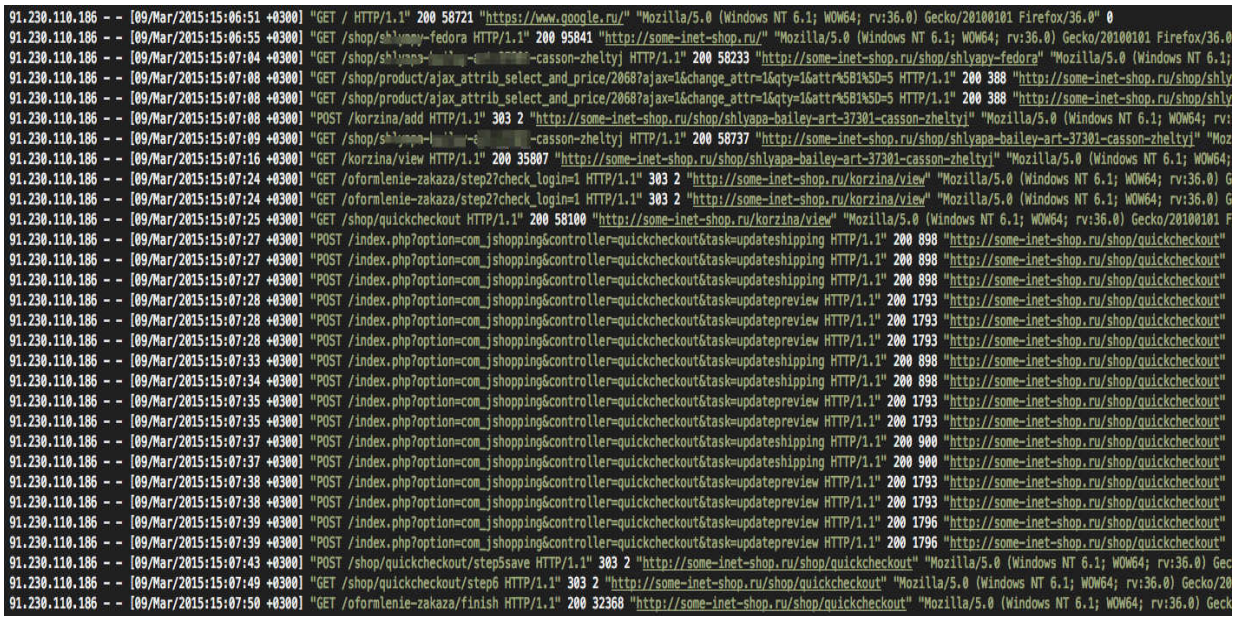
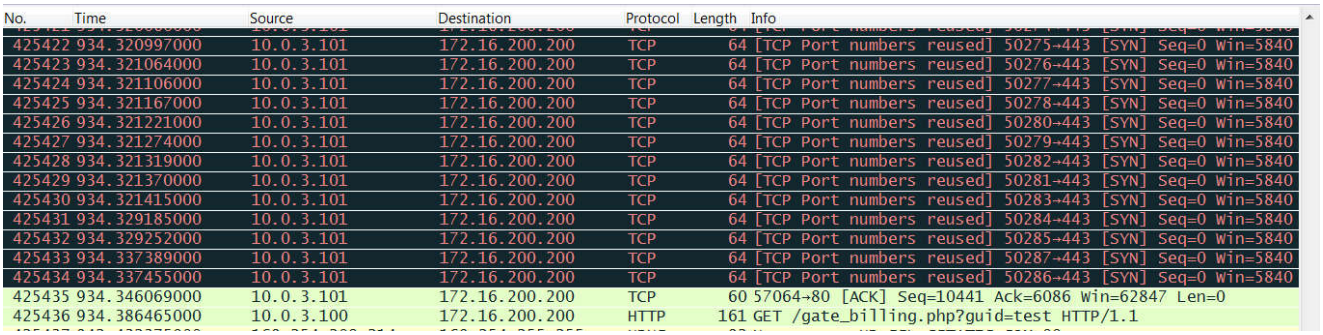


Рисунок 1.5 – Приклад dos атаки

Також можна проаналізувати трафік який надходить на сервер за допомогою файлів які бережуть в собі всю інформацію про вхідні та вихідні підключення.

Одним з типів атак є Dos атака, приклад такої атаки зображено на рисунку 1.5. Також існує атака під назвою Ddos, це атака типу Dos, але ресурси при такій атаці в декілька десятків разів більші ніж при звичайній Dos атаці, приклад такої атаки можна побачити на рисунку 1.6.



No.	Time	Source	Destination	Protocol	Length	Info
425422	934.320997000	10.0.3.101	172.16.200.200	TCP	64	[TCP Port numbers reused] 50275→443 [SYN] Seq=0 Win=5840
425423	934.321064000	10.0.3.101	172.16.200.200	TCP	64	[TCP Port numbers reused] 50276→443 [SYN] Seq=0 Win=5840
425424	934.321106000	10.0.3.101	172.16.200.200	TCP	64	[TCP Port numbers reused] 50277→443 [SYN] Seq=0 Win=5840
425425	934.321167000	10.0.3.101	172.16.200.200	TCP	64	[TCP Port numbers reused] 50278→443 [SYN] Seq=0 Win=5840
425426	934.321221000	10.0.3.101	172.16.200.200	TCP	64	[TCP Port numbers reused] 50280→443 [SYN] Seq=0 Win=5840
425427	934.321274000	10.0.3.101	172.16.200.200	TCP	64	[TCP Port numbers reused] 50279→443 [SYN] Seq=0 Win=5840
425428	934.321319000	10.0.3.101	172.16.200.200	TCP	64	[TCP Port numbers reused] 50282→443 [SYN] Seq=0 Win=5840
425429	934.321370000	10.0.3.101	172.16.200.200	TCP	64	[TCP Port numbers reused] 50281→443 [SYN] Seq=0 Win=5840
425430	934.321415000	10.0.3.101	172.16.200.200	TCP	64	[TCP Port numbers reused] 50283→443 [SYN] Seq=0 Win=5840
425431	934.329185000	10.0.3.101	172.16.200.200	TCP	64	[TCP Port numbers reused] 50284→443 [SYN] Seq=0 Win=5840
425432	934.329252000	10.0.3.101	172.16.200.200	TCP	64	[TCP Port numbers reused] 50285→443 [SYN] Seq=0 Win=5840
425433	934.337389000	10.0.3.101	172.16.200.200	TCP	64	[TCP Port numbers reused] 50287→443 [SYN] Seq=0 Win=5840
425434	934.337455000	10.0.3.101	172.16.200.200	TCP	64	[TCP Port numbers reused] 50286→443 [SYN] Seq=0 Win=5840
425435	934.346069000	10.0.3.101	172.16.200.200	TCP	60	57064→80 [ACK] Seq=10441 Ack=6086 Win=62847 Len=0
425436	934.386465000	10.0.3.100	172.16.200.200	HTTP	161	GET /gate_billing.php?guid=test HTTP/1.1

Рисунок 1.6 – Приклад ddos атаки

1.3 Постановка завдання

Проаналізувавши існуючі типи фаєрволів їх переваги та недоліки, структуру мережі та її потреби, було вирішено використовувати програмний фаєрвол, оскільки його вартість є низькою в порівнянні з апаратним та ефективність не сильно поступається апаратному фаєрволу.

Існує дуже багато програмних фаєрволів які мають свої сильні та слабкі сторони. Переглянувши їх порівняльну характеристику я зупинився на фаєрволі під назвою Fail2Ban, оскільки цей програмний фаєрвол є безкоштовним та має дуже гнучку систему налаштувань яка підійде під усі потреби мережі.

Завданням дипломної роботи є створення фаєрволу який задовольнить усі вище перераховані потреби та не буде поступатись аналогічним рішенням.

2 СТРУКТУРА ТА АЛГОРИТМИ FAIL2BAN

2.1 Структура Fail2Ban

З точки зору архітектури Fail2ban являє собою систему "клієнт-сервер". Серверна частина - це багатопоточна програма, яка прослуховує Unix-сокети, чекаючи надходження команд, і відправляє клієнту необхідну інформацію. Все це відбувається в режимі реального часу. Сам сервер не володіє жодною інформацією про поточний статус файлів конфігурації, тому при запуску знаходиться в стані "за замовчуванням", в якому не визначені жодні блокування та інші параметри.

Клієнтська частина - є зовнішньою, інтерфейсним компонентом всієї описуваної підсистеми. Клієнт встановлює з'єднання через сокет сервера і посилає через нього команди для конфігурації сервера і виконання необхідних операцій. Клієнт може зчитувати і передавати вміст конфігураційних файлів або просто відправити на сервер одну команду, використовуючи для цього командний рядок shell-оболонки або власний інтерактивний режим, який активізується за допомогою ключа "-i" [13]. Графічне представлення структури fail2ban зображено у ДП.КСМ.07144/14.00.00.000 С1.

Набір файлів конфігурації може мати вигляд, показаний на рисунку 2.1. Структура каталогів і конфігураційних файлів отримана за допомогою стандартної утиліти tree.

При внесенні змін в конфігурацію на рисунку 3.1. структурі зазвичай додаються файли з розширенням .local (fail2ban.local, jail.local і т.п.). Параметри настройки, що містяться в .local-файлах, мають перевагу над аналогічними параметрами, записаними в .conf-файлах. На практиці це означає, що спочатку зчитується вміст .conf-файлів, а потім вміст .local-файлів, тому значення раніше визначених параметрів можуть бути замінені. Таким чином, в .local-файлах можна зберігати лише ті значення параметрів, які потрібно скорегувати. Більш того, всі необхідні зміни конфігурації слід вносити до відповідних .local-файли, а не в файли з розширенням .conf.

Це допомагає підтримувати коректність загальної структури конфігурації і уникнути проблем при оновленні всієї підсистеми Fail2ban.

```
$ tree /etc/fail2ban/
/etc/fail2ban/
├── action.d
│   ├── complain.conf
│   ├── dshield.conf
│   ├── hostsdeny.conf
│   ├── ipfilter.conf
│   ├── ipfw.conf
│   ├── iptables-allports.conf
│   ├── iptables.conf
│   ├── iptables-multiport.conf
│   ├── iptables-multiport-log.conf
│   ├── iptables-new.conf
│   ├── mail-buffered.conf
│   ├── mail.conf
│   ├── mail-whois.conf
│   ├── mail-whois-lines.conf
│   ├── mynetwatchman.conf
│   ├── sendmail-buffered.conf
│   ├── sendmail.conf
│   ├── sendmail-whois.conf
│   ├── sendmail-whois-lines.conf
│   └── shorewall.conf
├── fail2ban.conf
├── filter.d
│   ├── apache-auth.conf
│   ├── apache-badbots.conf
│   ├── apache-nohome.conf
│   ├── apache-noscript.conf
│   ├── apache-overflows.conf
│   ├── common.conf
│   ├── courierlogin.conf
│   ├── couriersmtp.conf
│   ├── cyrus-imap.conf
│   ├── exim.conf
│   ├── gssftpd.conf
│   ├── lighttpd-fastcgi.conf
│   ├── named-refused.conf
│   ├── pam-generic.conf
│   ├── php-url-fopen.conf
│   ├── postfix.conf
│   ├── proftpd.conf
│   ├── pure-ftpd.conf
│   ├── qmail.conf
│   ├── sasl.conf
│   ├── sieve.conf
│   ├── sshd.conf
│   ├── sshd-ddos.conf
│   ├── vsftpd.conf
│   ├── webmin-auth.conf
│   ├── wuftpd.conf
│   └── xinetd-fail.conf
└── jail.conf
2 directories, 49 files
```

Рисунок 2.1 – Файли і каталоги fail2ban

Ведення журналів самої підсистеми Fail2ban зазвичай визначається двома параметрами:

logtarget - задає напрямок потоку для виведення інформації про роботу fail2ban. Цей параметр може мати одне з наступних значень: STDOUT, STDERR, SYSLOG або ім'я файлу. За замовчуванням (якщо цей параметр не визначений) присвоюється ім'я файлу / var / log / fail2banlog.

loglevel - визначає ступінь подробиці виведеної інформації. Можливі значення: ERROR (тільки інформація про помилки), WARN (інформація про помилки і попереджувальні повідомлення), INFO (повна інформація про роботу), DEBUG (висновок більш докладних описів всіх дій, станів, помилок, необхідних для налагодження підсистеми). За замовчуванням заданий рівень 3 - INFO.

Ще один параметр, що визначає функціональність Fail2ban, - це socket, який задає ім'я файлу, використовуваного для обміну інформацією між клієнтом і сервером. За замовчуванням цим параметром присвоюється ім'я.

2.2 Алгоритм роботи Fail2Ban

Будь-який сервіс, доступний з Інтернету, піддається атакам зловмисників. Якщо сервіс вимагає аутентифікації, несанкціоновані користувачі і боти будуть намагатися проникнути в систему шляхом перебору облікових даних. Наприклад, сервіс SSH стане об'єктом атак ботів, які спробують пройти аутентифікацію за допомогою стандартних облікових даних. На щастя, сервіси типу fail2ban допомагають пом'якшити такі атаки. Сервіс fail2ban динамічно коригує правила брандмауера, щоб заблокувати адреси, які безуспішно намагаються увійти в систему певну кількість разів [14].

Основна ідея fail2ban полягає у відстеженні логів загальних сервісів для виявлення помилок аутентифікації. Коли fail2ban моніторить логи сервісу, він дивиться на налаштований для цього сервісу фільтр. Фільтр призначений для визначення збоїв аутентифікації цього конкретного сервісу на основі складних регулярних виразів. Шаблони регулярних виразів визначаються в змінної failregex.

Fail2ban пропонує готові файли фільтрів для популярних сервісів. Коли рядок в файлі балки сервісу збігається з параметром failregex у відповідному фільтрі, fail2ban виконує заданий фільтром дію. Дія визначається в змінної action в залежності від уподобань адміністратора.

Дія за замовчуванням - блокування потенційно шкідливого хоста / IP-адреси шляхом зміни правил брандмауера iptables.

Можна розширити цю дію - налаштувати електронні повідомлення адміністратора з даними про зловмисника або рядками балки, які викликали вказане дію.

Також можна змінювати дії і вказати щось інше замість стандартного iptables. Цей параметр може бути настільки складним або простим, наскільки цього вимагає ваша установка; в fail2ban є безліч різних конфігураційних файлів і опцій брандмауера [15].

За замовчуванням fail2ban блокує адресу на 10 хвилин у разі виявлення трьох невдалих спроб авторизації протягом 10 хвилин. Кількість невдалих спроб аутентифікації, необхідне для блокування адреси, переопределяется в розділі SSH конфігураційного файлу за замовчуванням - він дозволяє до 6 спроб. Адміністратор може налаштувати цей параметр самостійно.

На рисунку 2.2 графічно представлена робота системи фільтрації трафіку за домогою Fail2Ban.

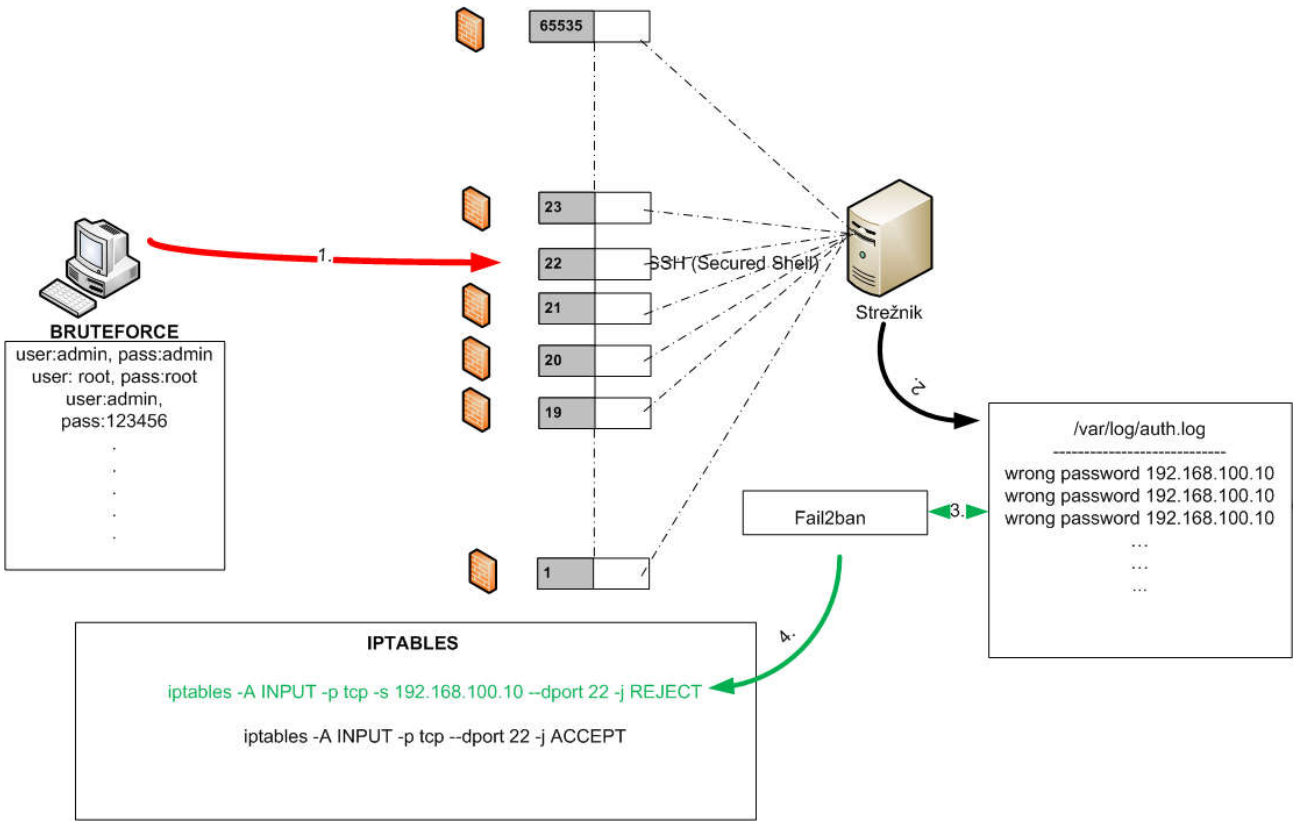


Рисунок 2.2 – Алгоритм робота fail2ban

При використанні мети за замовчуванням (iptables) для відстеження SSH-трафіку при запуску сервісу fail2ban створює новий ланцюжок.

Він додає нове правило в ланцюжок INPUT, яка направляє весь TCP-трафік, спрямований на порт 22, в створення нового обговорення. У новому ланцюжку fail2ban добавляє одне правило, яке повертається в ланцюжок INPUT [16].

Це змушує трафік стрибати в новий ланцюжок, а потім повертатися назад. Спочатку це не впливає на трафік. Однак коли IP-адреса перевищує кількість спроб аутентифікації, в початок нового ланцюжка додається правило, яке буде скидати трафік від цієї IP-адреси. Коли термін блокування закінчиться, правило в iptables буде видалено. Ланцюжок і пов'язані з ним правила видаляються при виході з fail2ban.

2.3 Реалізація захисту мережі

Навіть поверхневий огляд структури конфігураційних файлів, наведену на рисунку 2.3, дозволяє зрозуміти, що основним файлом є fail2ban.conf. І дійсно, в цьому файлі містяться загальні параметри конфігурації для демона (сервісу) fail2ban-server, такі як рівень моніторингу журналів і цільові об'єкти.

Крім того, тут можна визначити шлях (path) до сокету, використовуваному для обміну даними між клієнтом і сервером.

Незважаючи на те, що файл fail2ban.conf визначає основну конфігурацію всієї підсистеми, все-таки найбільш важливим для функціонування, мабуть, слід вважати файл jail.conf, який містить опису так званих "ізоляторів" (jails) [17].

Поточну конфігурацію кожного ізолятора описує набір наступних параметрів:

- ignoreip - список IP-адрес, які не повинні бути заблоковані, причому можна задавати діапазон адрес за допомогою маски підмережі або список IP-адрес, відокремлених один від одного пробілом;
- bantime - тривалість інтервалу блокування (в секундах) для IP-адреси;
- findtime - тривалість інтервалу виявлення "підозрілих збігів" (в журналах).

- Якщо за цей інтервал часу нічого підозрілого не виявлено, то лічильнику (counter) присвоюється значення 0;
- maxretry - кількість "підозрілих збігів" (тобто, значення лічильника counter), при якому спрацьовує певна операція по відношенню до відстежується IP-адресою;
- action - операція, яка повинна бути виконана, якщо значення лічильника стало рівним значенню параметра maxretry. За замовчуванням сервіс (порт) блокується для відслідковується IP-адреси;
- port - найменування сервісу або номер відповідного цьому сервісу порту, за яким ведеться спостереження;
- filter - ім'я фільтра, який повинен використовувати даний ізолятор, щоб виявляти "підозрілі збіги" в журналах (фільтри зберігаються в підкаталозі /etc/fail2ban/filter.d);
- logpath - шлях до файлу журналу, який повинен оброблятися за допомогою заданого фільтра.

Вміст файлу jail.conf розділяється на секції, які відповідають різним сервісам. Найперша секція називається [DEFAULT] і призначається для визначення загальних параметрів, значення яких мають силу в усіх наступних секціях даного файлу. Один з варіантів вмісту секції [DEFAULT] показаний на рисунку 2.3.

```
1 ignoreip = 127.0.0.1
2 bantime = 600
3 findtime = 600
4 maxretry = 3
```

Рисунок 2.3 – Параметри, які визначаються в секції default

Набір параметрів, наведених на рисунку 2.2, означає, що після трьох невдалих спроб (значення параметра maxretry) отримати доступ до системи з одного і того ж IP-адреси протягом 10 хвилин (600 секунд, значення параметра findtime) цю адресу буде автоматично заблокований на 10 хвилин (600 секунд, параметр bantime).

Для параметра ignoreip заданий тільки один адреса 127.0.0.1 (localhost), оскільки блокувати "самого себе" немає ніякого сенсу. Приклад простої конфігурації ізолятора для сервісу ssh приведений рисунку 2.4.

```
1 [ssh]
2
3 enabled = true
4 port = ssh
5 filter = sshd
6 logpath = /var/log/secure
7 maxretry = 5
```

Рисунок 2.4 – Конфігурація ізолятора для сервісу ssh

Тут визначена типова конфігурація для спостереження за сервісом ssh, яка наказує сканування журналу / var / log / secure з використанням фільтра sshd (файл sshd.conf). Якщо в журналі буде виявлено 5 спроб атаки на порт сервісу ssh, то IP-адреса, з якого проводилися ці спроби, буде заблокований (дія за замовчуванням). При необхідності до описаної конфігурації на рисунку 3.5 можна додати визначення параметра action на рисунку 2.5.

```
1 action = iptables
```

Рисунок 2.5 – Конфігурація ізолятора для сервісу ssh

Цей параметр визначає, які операції (дії) необхідно виконати в тому випадку, коли виявлені "підозрілі збіги" за допомогою заданого фільтра. Значним є ім'я файлу, розташованого в підкаталозі /etc/fail2ban/action.d/, але без зазначення його розширення .conf [18].

Таким чином, в даному випадку параметр action посилається на файл /etc/fail2ban/action.d/iptables.conf, і блокування "агресивної" IP-адреси буде виконано за допомогою зміни правил мережевого екрану iptables.

Для Web-сервера Apache ізолятор конфигурируется майже аналогічним чином з невеликими відмінностями, як показано на рисунку 2.6.

```
[apache-multiport]

enabled = true
port = http,https
filter = apache-auth
logpath = /var/log/httpd/*error_log
bantime = 1800
maxretry = 3
```

Рисунок 2.6 – Конфігурація ізолятора для web-сервера apache

Відмінності полягають в наступному: блокуватися будуть відразу два порти (http і https), сканування буде виконуватися в декількох файлах журналів, а умови блокування задані більш жорсткі - вже після трьох невдалих спроб аутентифікації IP-адреса буде заблокована на півгодини.

3 СТВОРЕННЯ ФАЄРВОЛУ ЗА ДОПОМОГОЮ FAIL2BAN

3.1 Удосконалення конфігураційних файлів

Насамперед потрібно налаштувати захист по протоколу SSH за допомогою програми Fail2ban. Для цього потрібно зайти в файл jail.local та знайти секцію ssh.

Вона повинна бути активна за замовчуванням. Проте, про всяк випадок, потрібно переконатись, що в значенні параметра enabled встановлено true, а не false.

Після цього потрібно вказати значення параметрів, на підставі яких Fail2ban повинен виконувати відстеження активності:

- filter - фільтр, який буде використовуватися. За замовчуванням це /etc/fail2ban/filter.d/sshd.conf;
- action - дії, які буде виконувати Fail2ban при виявленні атакуючого IP-адреси, всі правила реагування на дії зловмисника описані у файлі /etc/fail2ban/action.d. Відповідно, в якості значення параметра action не може бути вказана інформація, якої немає в файлі /etc/fail2ban/action.d;
- logpath - повний шлях до файлу, в який буде записуватися інформація про спроби отримання доступу до VPS.
- findtime - час в секундах, протягом якого спостерігається сайтів із підозрілою активністю;
- maxretry - дозволену кількість повторних спроб підключення до сервера;
- bantime - проміжок часу, протягом якого потрапив в чорний список IP залишатиметься заблокованим.

Також варто звернути увагу на той факт, що зовсім необов'язково прописувати значення вищевказаних параметрів в кожній секції. Якщо їх не згадувати, в дію вступлять настройки, зазначені в головному розділі «DEFAULT». Головне, щоб для змінної «enabled» було зазначено значення «true» [19].

Розглянемо застосування параметрів реагування більш детально. Приклад конфігурації Fail2ban на порту SSH:

```
[ssh]
enabled = true
port    = ssh
filter = sshd
action = iptables[name=sshd, port=ssh, protocol=tcp]
         sendmail-whois[name=ssh, dest=****@yandex.ru, sender=fail2ban@***.ru]
logpath = /var/log/auth.log
maxretry = 3
bantime = 600
```

Запис вище означає, що, якщо виконано більше 3 невдалих спроб підключення до основних портів SSH, то IP-адреса, з якої виконувалася авторизація, потрапить в бан на 10 хвилин. Правило заборони буде додано в iptables. У той же час власник сервера отримає повідомлення на e-mail, вказаний в значенні змінної dest, про те, що вказаний IP був заблокований за спробу отримання несанкціонованого доступу по протоколу SSH. Також в повідомленні буде вказана WHOIS інформація про заблокованому IP. Приклад такого повідомлення від Fail2ban наведено в додатку А.

Для того щоб повідомлення про блокування зловмисника відправлялося на e-mail, необхідна наявність поштового клієнта на сервері.

Додатково для захисту SSH активовуємо наступну секцію:

```
[ssh-ddos]
enabled = true
port    = ssh
filter = sshd-ddos
logpath = /var/log/auth.log
```

Для зберігання великих списків заблокованих IP-адрес можна використовувати комбінацію Netfilter / Iptables і IPsets. Щоб налаштувати роботу Fai2ban таким чином, активовуємо розділ [ssh-iptables-ipset4]:

```
[ssh-iptables-ipset4]
enabled = true
port    = ssh
filter = sshd
action = iptables[name=SSH, port=ssh, protocol=tcp]
logpath = /var/log/auth.log
findtime = 300
maxretry = 3
bantime = 600
```

Після редагування конфігураційного файлу потрібно зберегти внесені зміни. Аналогічним чином потрібно захистити і інші сервіси які будуть використовуватись у мережі.

Захист поштового сервера відбувається налаштуванням параметрів які відповідають за усі дії з електронною поштоую. Приклади налаштувань конфігураційного файлу для захисту поштового сервера postfix який я використовую у своїй мережі [20].

```
[postfix]
enabled = true
port    = smtp,ssmtp,submission
action = iptables[name=Postfix-smtp, port=smtp, protocol=tcp]
filter = postfix
logpath = /var/log/mail.log
bantime = 86400
maxretry = 3
```

findtime = 3600

ignoreip = 127.0.0.1

Важливим за чим потрбіно слідкувати, щоб шлях до файлу зберігання логів (logpath) було вказано коректно. В іншому випадку перезапуск Fail2ban завершиться повідомленням про помилку. Якщо якийсь із лог-файлів відсутній на сервері, його потрібно створити самостійно командою touch, наприклад, touch /var/log/mail.log. Обов'язково потрібно призначити йому необхідні права доступу командою chmod 755 /var/log/mail.log [21].

[sasl]

enabled = true

port = smtp,ssmtp,submission,imap2,imap3,imaps,pop3,pop3s

filter = postfix-sasl

action = iptables[name=Postfix-smtp, port=smtp, protocol=tcp]

logpath = /var/log/mail.log

bantime = 86400

maxretry = 3

findtime = 3600

Захист поштового сервера dovecot можна активувати наступним чином:

[dovecot]

enabled = true

port = smtp,ssmtp,submission,imap2,imap3,imaps,pop3,pop3s

filter = dovecot

action = iptables-multiport [name = dovecot-pop3imap, port="pop3, pop3s, imap, imaps", protocol=tcp]

logpath = /var/log/mail.log

maxretry = 3

findtime = 3600

bantime = 86400

Для захисту веб-сервера Apache я використовую такі параметри Fail2ban:

[apache]

enabled = true

port = http,https

filter = apache-auth

logpath = /var/log/apache2/error.log

maxretry = 3

Захист сервісу multiport можна активувати наступним чином:

[apache-multiport]

enabled = true

port = http,https

filter = apache-auth

logpath = /var/log/apache2/error.log

maxretry = 3

Захист сервісу overflows можна активувати наступним чином:

[apache-overflows]

enabled = true

port = http,https

filter = apache-overflows

logpath = /var/log/apache2/error.log

maxretry = 2

Як можна було помітити, в використовуваних вище секціях файлу jail.local відсутні значення параметра action. В цьому випадку при виявленні атаки на сервіс apache програма Fail2ban буде виконувати дію, визначену в секції [DEFAULT], а саме action = iptables-multiport. Це означає, що атакуючий IP-адреса буде заблокований в iptables за допомогою так званого модуля multiports [22].

Модуль multiports дозволяє налаштувати правило відразу для діапазонів портів. Для захисту FTP-сервера vsftpd за допомогою Fail2ban я використовую такі параметри:

```
[vsftpd]
enabled = true
port    = ftp,ftp-data,ftps,ftps-data
filter = vsftpd
logpath = /var/log/vsftpd.log
action = iptables[name=VSFTPD, port=21, protocol=tcp]
bantime = 600
maxretry = 3
findtime = 1800
```

Не потрібно забувати про необхідність перезапуску Fail2ban після кожного редагування конфігураційного файлу. На цьому налаштування Fail2ban можна вважати завершеним.

3.2 Інтерфейс керування системою Fail2Ban

Для зручності в управлінні усіма конфігураціями сервісу Fail2Ban був розроблений web-інтрефейс, який спростить внесення змін у параметри правил.

Web-інтрефейс - це файл fail2ban, який зв'язується з файлом fail2con через файл fail2rest [23].

Web-інтрефейс дозволяє керувати наступним:

- failregex для видалення та додавання нових збійних виправлень;
- заборонені ip-адреси для блокування та розблокування ip-адреси;
- per jail config для налаштування часу пошуку, та перегляду списку файлів що заблоковані;
- сигналізація для сповіщення, коли ip-адресу заблоковано та не заблоковано, за допомогою налаштувань інтервалу часу;
- тестування регулярних викликів для тестування, ігнорування та повторного виправлення помилок у поточних журналах, щоб швидко створювати та налаштовувати регулярні виправлення;
- звітність використовує час, коли IP-адреса заблокована, і показує тенденції за допомогою візуалізації.

Графічний вид web-інтерфесу представлений на рисунку 3.1-3.4.

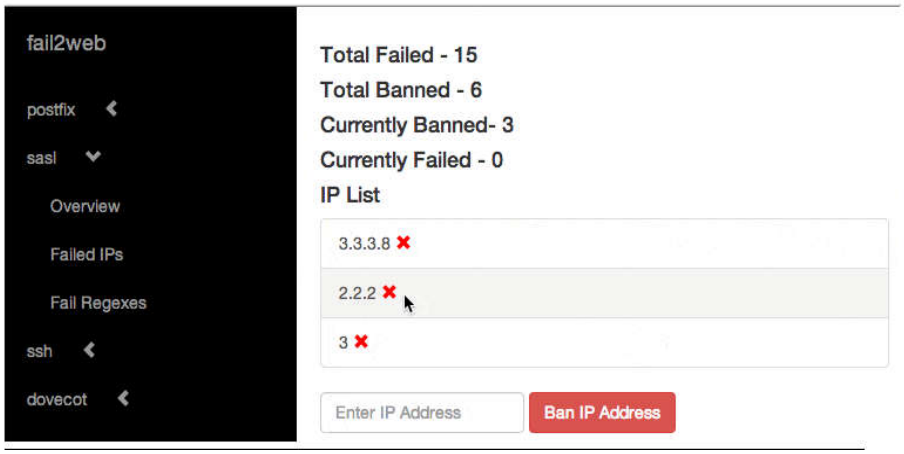


Рисунок 3.1 – Графічне представлення web-інтерфейсу

Інтерфейс представлений у виді меню яке ділиться на підпункти. У кожної вкладки меню є свій функціонал та на кожній з них виводиться різна інформація. Дане меню є дуже зручним та інтуїтивним у використанні.

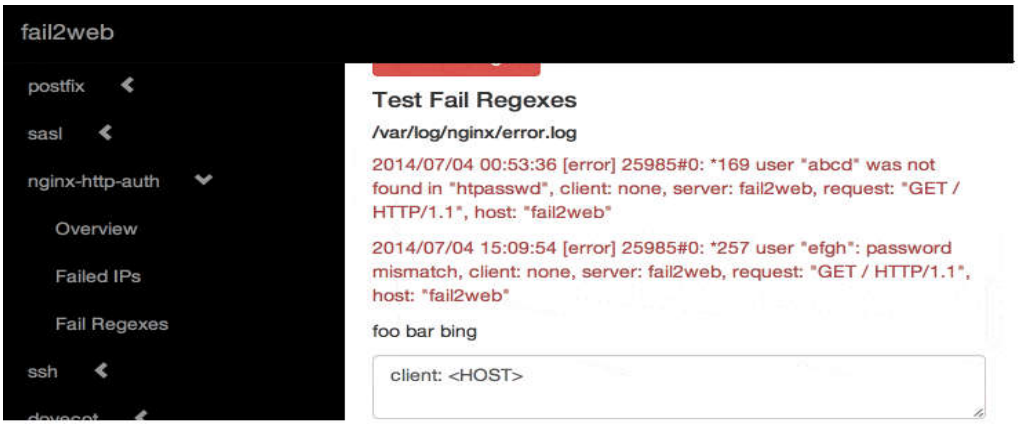


Рисунок 3.2 – Графічне представлення web-інтерфейсу, вкладка ssh

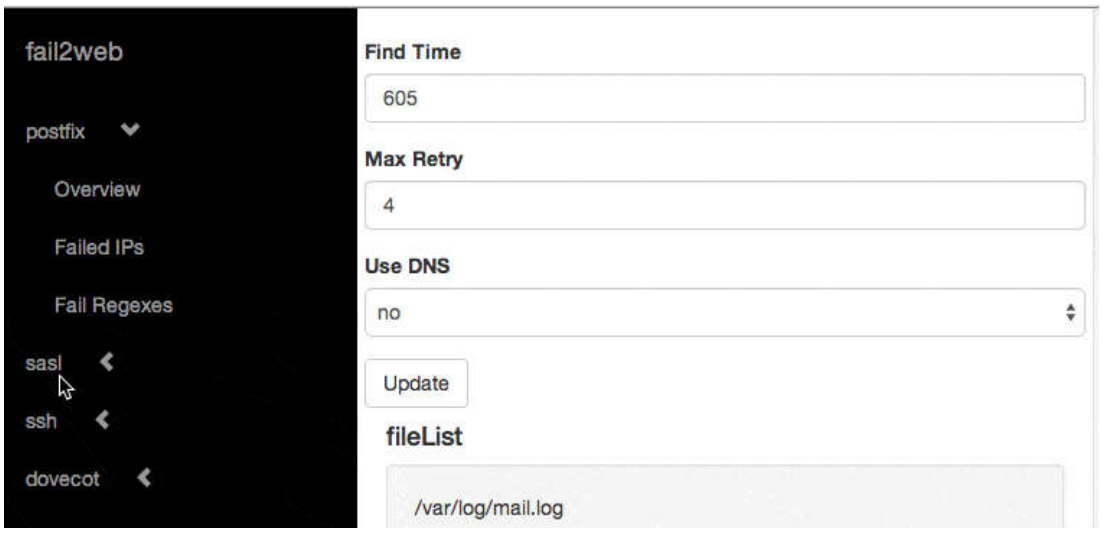


Рисунок 3.3 – Графічне представлення web-інтерфейсу, вкладка sasl

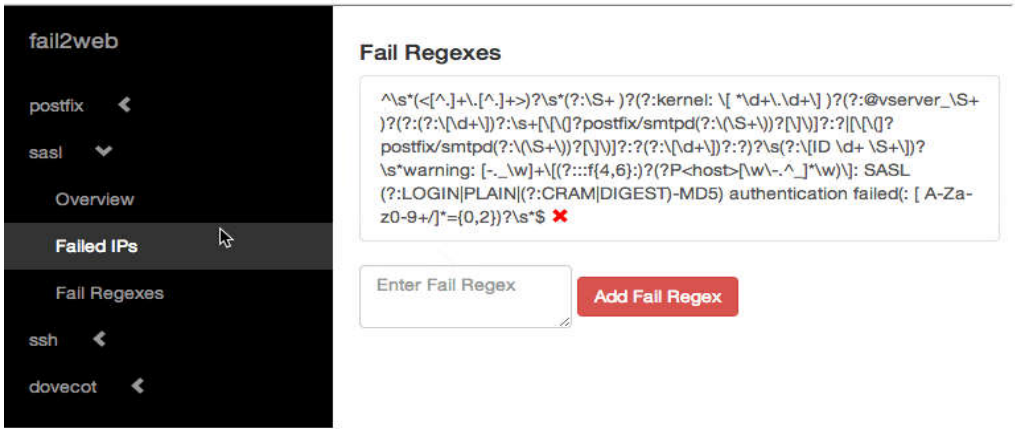


Рисунок 3.4 – Графічне представлення web-інтерфейсу, вкладка falled ips

За допомогою такого не складного інтерфейсу управління усіма ресурсами фаєрволу стає набагато простішим та зрозумілішим.

3.3 Моніторинг роботи системи

Моніторинг журналу fail2ban - те, що потрібно дослідити саме на цьому етапі. Чи можна використовувати fail2ban для постійного блокування адрес, коли їх заблокували кілька разів за допомогою звичайного фільтра fail2ban.

Здається, це можливо, хоча може знадобитися встановити різні блокування для різних портів. Наприклад, для повторних правопорушників відповідно до фільтра sendmail потрібно додати наступне /etc/fail2ban/jail.local:

```
[fail2ban-smtp]
enabled = true
порт = smtp
фільтр = fail2ban-smtp
logpath = /var/log/fail2ban.log
maxretry = 3
findtime = 21600
бантим = 86400
```

І тоді потрібно створити файл /etc/fail2ban/filter.d/fail2ban-smtp.conf з наступними налаштуваннями:

```
failregex = \ [sendmail] Ban <HOST>
ignoreregex =
```

За допомогою цих параметрів fail2ban відстежуватиме власний лог-файл, і якщо HOST буде заборонено три рази (maxretry) через шість годин (findtime), вони отримають нову заборону, яка триватиме 24 годину. Якщо встановити значення bantime як негативне, то HOST, про який йде мова, ніколи не буде заблоковано.

Подібні правила можна налаштувати для інших існуючих заборон, і їх можна поєднати, якщо вони мають один і той же порт.

Після того як фаєрвол сам відстежує власний лог-файл потрібно всі дані перенести у графік за допомогою додаткового програмного забезпечення. Після встановлення якого весь трафік можна проаналізувати за допомогою графіку зображеного на рисунку 3.5.

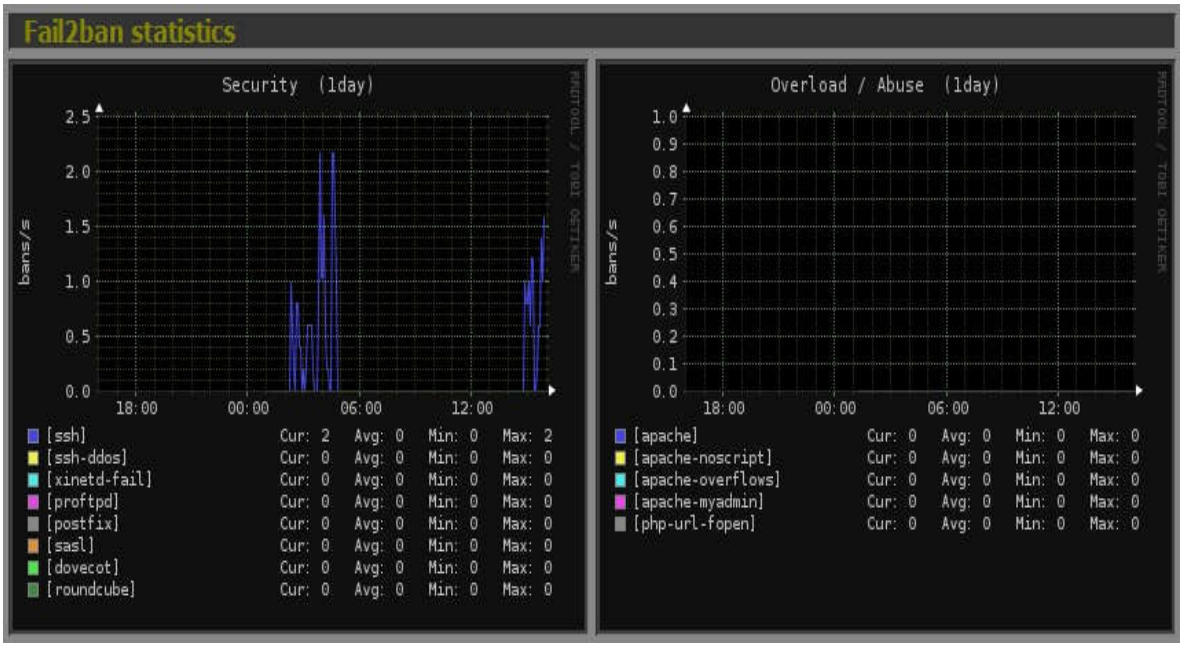


Рисунок 3.5 – Моніторинг трафіку у вигляді графіку

Після виконання усієї вище перерахованої роботи слідкувати та аналізувати роботу фаєрволу буде в декілька разів простіше та зрозуміліше. Це полегшить роботу над внесенням змін у роботу системи та знаходженням помилок у роботі системи.

4 ТЕХНІКО-ЕКОНОМІЧНЕ ОБГРУНТУВАННЯ

4.1 Стадії технологічного процесу

У розробці проектного рішення задіяні наступні спеціалісти - розробники, а саме: керівник проекту (К); студент-дипломник (С); консультант техніко-економічного розділу (КТЕО) [24].

Форму поділу робіт по всіх основних етапах і видах робіт, які повинні бути виконані показано в таблиці 4.1.

Таблиця 4.1 - Середній час виконання проекту та стадії технологічного процесу

/п	Назва операції (стадії)	Виконавець, посада	Середній час виконання операції, год.
	Підготовка	Студент	7
	Розробка проекту системи	Керівник ДП	16
		Консультант ТЕО, доцент	2
		Студент	207
	Проектування технічної частини системи	Студент	16
	Розробка програмного продукту системи	Студент	6
	Встановлення та налаштування прогр. забезпечення	Студент	8
	Тестування системи	Студент	3
Разом			279

4.1.2 Визначення витрат на оплату праці та відрахувань на соціальні заходи.

Витрати на оплату праці включають заробітну плату (ЗП) всіх категорій працівників, безпосередньо зайнятих на всіх етапах проектування.

Витрати на оплату праці розробників проекту визначаються за формулою:

$$B_{ОП} = \sum_{i=1}^{N} \sum_{j=1}^{M} n_{ij} \cdot t_{ij} \cdot C_{ij} \ , \qquad (4.1)$$

де n_{ij} – чисельність розробників i-ої спеціальності j-го тарифного розряду осіб;

t_{ij} – затрачений час на розробку проекту співробітником i-ої спеціальності j-го тарифного розряду, год;

C_{ij} – годинна ставка працівника i-ої спеціальності j-го тарифного розряду, грн.,

Середньо годинна ставка працівника може бути розрахована за формулою:

$$C_{ij} = \frac{C_{ij}^{0}(1+h)}{РЧ_{i}}, \qquad (4.2)$$

де C_{ij} – основна місячна заробітна плата розробника i-ої спеціальності j-го тарифного розряду, грн.;

h – коефіцієнт, що визначає розмір додаткової заробітної плати (при умові наявності доплат);

$РЧ_i$ - місячний фонд робочого часу працівника i-ої спеціальності j-го тарифного розряду, год. (приймаємо 168 год.).

Таблиця 4.2 - Вихідні дані для розрахунку витрат на оплату праці

п/п	Посада виконавців	Місячний оклад (стипендія),грн.	Коефіцієнт Додаткової з/п	Підсумок
	Керівник ДП, викладач	2458	0,4	3440
	Консультант техніко-економічного розділу,доцент	6026	1,39	14 402
	Студент	1290	0	1290

Звідси, загальні витрати на оплату праці ($B_{ОП}$) дорівнюють:

$$B_{ОП} = 16 * \frac{3440}{168} + 2 * \frac{14402}{168} + 240 * \frac{1290}{168} = 2347{,}7 \text{ грн.}$$

Крім того, слід визначити відрахування на соціальні заходи. Величну відрахувань у спеціальні державні фонди визначають у відсотковому співвідношенні від суми основної та додаткової заробітних плат. Згідно діючого нормативного законодавства сума відрахувань у спеціальні державні фонди складає 20,5 % від суми заробітної плати:

$$B_{ф} = \frac{20{,}5}{100} * 2347{,}7 = 481{,}3 \text{ грн.}$$

4.1.3 Розрахунок матеріальних витрат

Матеріальні витрати — це вартість витрачених матеріалів, малоцінних та швидкозношуваних предметів на виробництво продукції, робіт або послуг, а також матеріалів і МШП, витрачених на адміністративні, збутові та інші потреби підприємства.

Загальна сума витрат на матеріальні ресурси (B_M) визначається за формулою:

$$B_M = \sum_{i=1}^{n} K_i \cdot Ц_i, \qquad (4.3)$$

де K_i - витрата i-го типу матеріалу, натуральні одиниці вимірювання;

$Ц_i$ – ціна за одиницю i-го типу матеріалу, грн.;

i- тип матеріального ресурсу;

n - кількість типів матеріальних ресурсів.

Звідси, витрати на матеріальні ресурси дорівнюватимуть:

B_M =344,30 грн.

Проведені розрахунки занесемо у таблицю 4.3

Таблиця 4.3 - Зведені розрахунки покупних виробів

№ п/п	Найменування купованих виробів	Одиниця виміру	Ціна, грн	Кількість купованих виробів	Сума, грн	Транспортні витрати (10% від суми)	Загальна сума, грн
1	Папір (формат А4)	уп	85,0	2	170,00	17,0	187,0
2	Ручка кулькова	шт	10,0	2	20,00	2,0	22,0
3	Диски CD-R	шт	9,0	1	9,00	0,90	9,90
4	Зошит, 96 арк	шт	24	1	24	2,4	26,40
5	Тонер для принтера	уп	90	1	90	9,0	99,0
Разом							344,30

4.1.4 Розрахунок витрат на електроенергію

Для розробки КС використовується електрообладнання, тому необхідно розрахувати витрати на електроенергію.

Загальна сума витрат на електроенергію розраховується за формулою:

$$B_E = \sum_{i=1}^{n} P_i \cdot k_i \cdot T_i \cdot Ц \ , \qquad (4.4)$$

де P_i - паспортна потужність *i*-го електрообладнання, кВт;

k_i - коефіцієнт використання потужності *i*-го електрообладнання;

T_i - час роботи і-го устаткування за весь період розробки, год;

Ц - ціна електроенергії, грн / кВт· год;

i - тип електрообладнання;

n - кількість електрообладнання.

Для розробки проекту даної системи використовується один ноутбук потужністю P = 0,2 кВт, який за весь період розробки працює 200 годин та друкуючий пристрій потужністю P = 0,37 кВт, який працює 2 години.

Проміжні розрахунки на витрату електроенергії подані в таблиці 4.4

Таблиця 4.4 - Витрати на електроенергію

Наймену-вання устатку-вання	Паспортна потужність, кВт	Коефіцієнт використ-тання потужності	Час роботи обладнання для розробки, год	Ціна електро-енергії	Сума, грн.
Ноутбук	0,2	0,98	200	0,90	35,28
Принтер	0,37	0,98	2	0,90	0,65
Разом					35,93

4.1.5 Розрахунок суми амортизаційних відрахувань

Амортизація – це процес перенесення вартості основних фондів на вартість новоствореної продукції з метою їх повного відновлення.

Для визначення амортизаційних відрахувань застосуємо метод прямолінійного списання. Загальна сума амортизаційних відрахувань (B_{AM}) визначається за формулою:

$$B_{AM} = \sum_{i=1}^{n} \frac{B_i \cdot H_i}{100}, \quad (4.5)$$

де B_i - вартість i-го устаткування на початок звітного періоду, грн.;

H_i - річна норма амортизації i-го устаткування, %;

i - тип обладнання;

n - кількість устаткування.

Для проектування даної системи використовувався один ноутбук 6300грн., та принтер вартістю 1150грн, та подані в таблиці 4.5, обрахування витрат.

Тоді:

$$B_{AM} = \frac{6300 * 10}{100} + \frac{1150 * 20}{100} = 860 \text{ грн.}$$

Таблиця 4.5 - Амортизація основних фондів

Найменування устаткування	Вартість устаткування, грн.	Річна норма амортизації, %	Сума, грн.
Ноутбук	6300,00	10	630,00
Принтер	1150,00	20	230,00
Разом			860,00

4.1.6 Визначення транспортних витрат

Транспортні витрати слід прогнозувати у розмірі 8–12 % від загальної суми матеріальних витрат.

$$B_T = 0.12 * B_M, \quad (4.6)$$

де B_T – транспортні витрати.

$$B_T = 0{,}12 \cdot 344{,}30 = 41{,}31 \text{ грн.}$$

4.1.7 Обчислення накладних витрат

Накладні витрати пов’язані з обслуговуванням виробництва, утриманням апарату управління підприємства (фірми) та створення необхідних умов праці можуть становити до 150 % від суми основної та додаткової заробітної плати працівників. Накладні витрати для даного проекту подані далі.

$$H_B = 1{,}5 * B_{ОП}, \quad (4.7)$$

де H_B – накладні витрати.

$$H_B=1,5\cdot 2\,354,25=3\,531,37 \text{ грн.}$$

4.1.8 Обчислення інших витрат

Інші витрати є витратами, які не враховані в попередніх статтях. Вони становлять 10% від заробітної плати:

$$I = 2354,25 * 0,1 = 235,425 \text{ грн}$$

4.1.9 Складання кошторису витрат та визначення собівартості

Загальні витрати ($B_{КС}$) розрахуємо за формулою:

$$B_{КС}= B_{ОП} +B_{Ф}+B_{М}+B_{Е}+ B_{АМ}+ B_{Т}+H_B \quad (4.8)$$

Тобто:

$$B_{КС} = 7\,702,7 \text{ грн.}$$

Результати проведених розрахунків зведемо у таблицю 4.6.

Таблиця 4.6 - Кошторис витрат

Зміст витрат	Сума, грн.
1	2
Витрати на оплату праці (осн. і дод. ЗП)	2 354,25
Відрахування на соціальні заходи	482,62
Матеріальні витрати	344,30
Витрати на електроенергію	35,93

Продовження таблиці 4.6

1	2
Амортизаційні відрахування	860,00
Транспортні витрати	41,31
Накладні витрати	3 531,37
Інші витрати	235,425
Разом	7 885,205

4.2 Визначення експлуатаційних витрат

Для оцінки економічної ефективності розроблюваного програмного продукту слід порівняти його з аналогом, тобто існуючим програмним забезпеченням ідентичного функціонального призначення.

Експлуатаційні одноразові витрати по програмному забезпеченню і аналогу включають вартість підготовки даних і вартість роботи комп'ютера (за час дії програми):

$$E_n = E_{1n} + E_{2n}, \quad (4.9)$$

де E_n - одноразові експлуатаційні витрати на ПЗ (аналог), грн.;

E_{1n} - вартість підготовки даних для експлуатації ПЗ (аналогу), грн.;

E_{2n} - вартість роботи комп'ютера для виконання проектного рішення (аналогу), грн.

Річні експлуатаційні витрати B_{en} визначаються за формулою:

$$B_{en} = E_n * N_n, \quad (4.10)$$

де N_n - періодичність експлуатації ПЗ (аналогу), раз/рік.

Вартість підготовки даних для роботи на комп'ютері визначається за формулою:

$$E_{1n} = \sum_{l=1}^{n} n_i t_i c_i \,, \qquad (4.11)$$

де i - категорії працівників, які приймають участь у підготовці даних (i=1,2,...n);

n_i - кількість працівників i-ої категорії, осіб.;

t_i - трудомісткість роботи співробітників i-ої категорії по підготовці даних, год.;

c_i - середньогодинна ставка працівника i-ої категорії з врахуванням додаткової заробітної плати, що знаходиться із співвідношення:

$$c_i = \frac{c_i^0 (1+b)}{m} \,, \qquad (4.12)$$

де c_i^0 - основна місячна заробітна плата працівника i-ої категорії, грн.;

b - коефіцієнт, який враховує додаткову заробітну плату (приймемо 0,57);

m - кількість робочих годин у місяці, год.

Для роботи з даними як для проектного рішення так і аналогу потрібен один працівник, основна місячна заробітна плата якого складає: c^0 = 3200 грн. Тоді:

$$c_i = \frac{3200(1+0{,}57)}{22*8} = 28{,}54 \text{ грн.}$$

Трудомісткість підготовки даних для проектного рішення складає 1 год., для аналога 1,5 год. Розрахунок витрат на підготовку даних та реалізацію проектного рішення на комп'ютері представлено в таблиці 4.1

Таблиця 4.1 - Розрахунок витрат на підготовку даних та реалізацію проектного рішення на комп'ютері

Час роботи співробітників, год.	Середньогодинна заробітна плата, грн./год.	Витрати, грн.
Проектне рішення		
1	28,54	28,54
Аналог		
1,5	28,54	42,81

Витрати на експлуатацію комп'ютера визначається за формулою:

$$E_{2n} = t * S_{МГ}, \quad (4.13)$$

де t - витрати машинного часу для реалізації проектного рішення (аналогу), год.;

$S_{МГ}$ - вартість однієї години роботи комп'ютера, грн./год.

$E_{2n} = 1*0,9 = 10$ грн.; $E_{2a} = 1,5 \cdot 0,9 = 15$ грн.

$E_n = 28,54+0,9 = 38,54$ грн.; $E_a = 42,81+0,9 = 52,81$ грн.

$B_{en} = 13,2*252 = 3\,807,72$ грн.; $B_{ea} = 19,75*252 = 5\,700,24$ грн.

4.3 Розрахунок ціни споживання проектного рішення

Ціна споживання - це витрати на придбання і експлуатацію проектного рішення за весь строк його служби:

$$Ц_{С(П)} = Ц_П + B_{(е)прv}, \quad (4.14)$$

де $Ц_п$ - ціна придбання проектного рішення, грн.:

$$Ц_п = К(1 + \frac{П_Р}{100}) + К_0 + К_к \quad (4.15)$$

де $К$ - кошторисна вартість;

$П_р$ - рентабельність;

$К_о$ - витрати на прив'язку та освоєння проектного рішення на конкретному об'єкті, грн.;

$К_к$ - витрати на доукомплектування технічних засобів на об'єкті, грн.;

Договірна ціна ($Ц_Д$) для проектних рішень розраховується за формулою:

$$Ц_Д = В_{КС} \cdot \left(1 + \frac{р}{100}\right), \quad (4.16)$$

де $В_{КС}$ – кошторисна вартість, грн.;

$р$ - середній рівень рентабельності, % (приймаємо 30% за погодженням з керівником).

$$Ц_Д = 7938,125 \cdot (1+0,3) = 10\ 250,76 \text{ грн.}$$

Вартість витрат на експлуатацію проектного рішення (за весь час його експлуатації), грн.:

$$В_{епрv} = \sum_{t=0}^{T} \frac{В_{еп}}{(1+R)^t}, \quad (4.17)$$

де $В_{еп}$ - річні експлуатаційні витрати, грн.;

$Т$ - строк служби проектного рішення, років;

R - річна ставка проценту банку.

$$B_{елpv} = \sum_{t=1}^{5} \frac{3\,807{,}72}{(1+0{,}08)^t} = 17628{,}3 \text{ грн.}$$

$$B_{елpv} = \sum_{t=1}^{5} \frac{5\,700{,}24}{(1+0{,}08)^t} = 28275{,}2 \text{ грн.}$$

Тоді ціна споживання проектного рішення дорівнюватиме:

$$Ц_{сп} = 10\,250{,}76+17628{,}3=27\,879{,}06 \text{ грн.}$$

Аналогічно визначається ціна споживання для аналогу:

$$Ц_{са} = 9\,500{,}0+28\,275{,}2=37\,775{,}20 \text{ грн.}$$

4.4 Визначення економічної ефективності

Економічна ефективність — досягнення найбільших результатів за найменших затрат живої та уречевленої праці. Економічна ефективність є конкретною формою дії закону економії часу. За капіталістичного способу виробництва узагальнюючий показник економічної ефективності — норма прибутку.

Економічний ефект в сфері проектування рішення:

$$E_{ПР} = Ц_{П} - Ц_{А}, \qquad (4.18)$$

$$E_{ПР} = 10\,319{,}56 - 9\,500{,}0 = 819{,}56 \text{ грн.}$$

Річний економічний ефект в сфері експлуатації:

$$E_{кс} = B_{еа} - B_{еп,} \quad (4.19)$$

Тоді:

$$E_{кс} = 5\ 700{,}24 - 3\ 807{,}72 = 1\ 892{,}52 \text{ грн.}$$

Додатковий економічний ефект у сфері експлуатації:

$$\Delta E_{екс} = \sum_{t=1}^{T} E_{екс} (1+R)^{T-t} \quad (4.20)$$

Тоді:

$$\Delta E_{екс} = \sum_{t=1}^{5} 1892{,}52(1+0{,}08)^{5-t} = 10219{,}61 \text{ грн.}$$

Сумарний ефект складає:

$$E = E_{пр} + \Delta E_{екс,} \quad (4.21)$$

Тоді:

$$E = 819{,}56 + 10\ 219{,}61 = 11\ 039{,}17 \text{ грн.}$$

Таблиця 4.1 - Показники економічної ефективності проектного рішення

№	Найменування	Одиниці вимірювання	Значення показників	
			Базовий варіант	Новий варіант
	1	2	3	4
1	Ціна придбання	грн.	9500,0	10 319,56
2	Річні експлуатаційні витрати	грн.	28 275,2	17 628,3

	1	2	3	4
3	Ціна споживання	грн.	37 775,2	28 275,2
4	Економічний ефект в сфері проектування	грн.	-	819,56
5	Економічний ефект в сфері експлуатації	грн.	-	1 892,52
6	Додатковий ефект в сфері експлуатації	грн.	-	10 219,61
7	Сумарний ефект	грн.	11039,17	

Отже, в даному розділі проведено розрахунок витрат на розробку проектного рішення. Здійснено порівняння з існуючим аналогом, і цим показано, що дане проектне показує економну доцільність «Фільтрації трафіку Fail2Ban на основі аналізу файлів журналів доступів до сервісу». Згідно проведеного економічного обґрунтування дане проектне рішення є конкурентноздатним.

ВИСНОВКИ

Проаналізувавши усі готові фаєрволи, а саме програмні та апаратні фаєрволи від багатьох розробників було прийнято рішення, що написання власного фаєрволу буде найкращим рішенням. Оскільки програмні фаєрволи не дають можливості налаштувати усі параметри, та використовують значну частину ресурсів сервера, а апаратні рішення є дуже дорогими у використанні та є не рентабельними у моїй мережі.

При написанні усіх правил у фаєрволі було враховано усі протоколи які будуть використовуватись у мережі, щоб оптимізувати роботу фаєрволу та зменшити витрати ресурсів на роботу фаєрволу.

Для зручності у використанні був розроблений інтерфейс який полегшує роботу з фаєрволом, та економить час на внесені змін та моніторингу роботи усіх сервісів.

Дане рішення на мою думку є найкращим яке можна було реалу зівати у цій мережі, оскільки після завершення усіх робіт над фаєрволом він задовольняє усі потреби мережі в ефективності, зручності та надійності.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Семенов Ю. А. Телекоммуникационные технологии. Интернет-университет информационных технологий [Электронный ресурс]. – Режим доступа: http://book.itep.ru

2. Олифер В. Г. Компьютерные сети. Принципы, технологии, протоколы: учебник для вузов: 3-е изд. / В. Г Олифер., Н. А. Олифер // – СПб.: Питер, 2006.

3. Зайченко Ю. П. Анализ и оптимизация характеристик сетей MPLS по заданным показателям качества / Ю. П. Зайченко, Ахмед А. М. Шарадка // Вісник національного технічного університету України КПІ сер. Інформатика управління та обчислювальна техніка. Вип. 43. – 113–123 с.

4. Будылдина Н. В. Разработка программного обеспечения для оптимизации мультисервисных сетей / Н. В. Будылдина., П. А. Коновалов // Открытое образование, июнь 2006. – 58 с.

5. Зайцев Д. А. Моделирование телекоммуникационных сетей в системе NS. / Д. А. Зайцев, Т. Н. Шинкарчук // Наукові праці ОНАЗ ім. О. С. Попова. – 2006.– № 2.

6. Кучерявый Е.А. Управление трафиком и качество обслуживания в сети Интернет / Е.А. Кучерявый // – М.: Наука и Техника. – 2007. – 336 с.

7. Panwar Li. Y. S. On the Performance of MPLS TE Queues for QoS Routing // Panwar Li. Y. Liu C.J. Simulation series. – 2004. – Vol. 36; part 3. – P. 170–174.

8. Аткинсон Л. Mysql. Библиотека профессионала – М Энергоатомиздат, 2002 – 496 с.

9. Дейт К. Руководство по реляционной СУБД DB2 – М.: Финансы и статистика, 1988 – 320 с.

10. Мейер М.Теория реляционных баз данных – М.: Мир, 1987 – 608 с.

11. Семантична модель: База даних [Електронний ресурс] Режим доступу: http://citforum.ru/database/advanced_intro/27.shtml.

12. Риккарди Г. Системы баз данных. Теория и практика использования в Internet – М.:Вильямс, 2001 – 240с.

13. ДейтК. Дж. Введение в системы баз даннях – СПб: Изд-во "Пітер", 2005 – 1315с.

14. Роберт І.В. Сучасні інформаційні технології в освіті: дидактичні проблеми, перспективи використання ― М.: Школа-Пресс, 1994― 205с.

15. Розділ 2.Основи UML – діаграм: [Електронний ресурс] Режим доступу:https://docs.kde.org/trunk4/uk/kdesdk/umbrello/uml-basics.html

16. Мюллер Р.Дж. Базы данных и UML. Проектирование / Перевод. с англ. Е. Молодцова. – М.: Издательство “Лори”,2002 – 432с.

17. Острей О.Р. Діаграми класів UML як засіб моделювання інформаційної системи моніторингу освіти / М.: - 2008. № 2. – С.85-89.

18. Електронна енциклопедія Вікіпедія: JSON[Електронний ресурс] Режим доступу: https://uk.wikipedia.org/wiki/JSON

19. Компонентне або модульне тестування: [Електронний ресурс] Режим доступу: http://www.protesting.ru/testing/levels/component.html

20. Електронна енциклопедія Вікіпедія: Модульне тестування [Електронний ресурс] Режим доступу: https://uk.wikipedia.org/wiki/Модульне

21. Електронна енциклопедія Вікіпедія: Список кодів стану HTTP [Електронний ресурс] Режим доступу:https://uk.wikipedia.org/wiki/список

22. Шапошников І. Web-сайт своїми руками./ І. Шапошников – СПб: Изд-во "Пітер", 2002 – 390с.

23. Гаевский А. Ю. Самоучитель по созданию Web-страниц: HTML, JavaScript, Dynamic HTML / А. Ю. Гаевский, В. А.Романовский. — К.: А.С.К., 2002 — 472с.

24. Методичні вказівки до написання техніко – економічного розділу для дипломних проектів на здобуття освітньо – кваліфікаційного рівня «Бакалавр» напряму підготовки 6.050102 «Комп’ютерна інженерія» / І.Р. Паздрій. – Тернопіль: ТНЕУ, 2015. – 36с.

25. Методичні рекомендації до виконання дипломного проекту з освітньо – кваліфікаційного рівня «Бакалавр» напряму підготовки 6.050102 «Комп'ютерна інженерія» фахового спрямування «Комп'ютерні системи та мережі» / Л.О. Дубчак О.М. Березький, Р.Б Трембач, Г.М. Мельник, Ю.М. Батько, С.В. Івасьєв / Під ред. О.М. Березького. – Тернопіль: ТНЕУ, 2016.- 60с.