Лабораторная работа № 16

Базовая защита от атак типа «brute force

Шубина София Антоновна

Содержание

Список иллюстраций

1 Цель работы

Получить навыки работы с программным средством Fail2ban для обеспечения базовой защиты от атак типа «brute force».

2 Задание

1. Установите и настройте сервер Samba.
2. Настройте на клиенте доступ к разделяемым ресурсам.
3. Напишите скрипты для Vagrant, фиксирующие действия по установке и настройке сервера Samba для доступа к разделяемым ресурсам во внутреннем окружении виртуальных машин server и client. Соответствующим образом необходимо внести изменения в Vagrantfile.

3 Выполнение лабораторной работы

3.1 Защита с помощью Fail2ban

Загрузим нашу операционную систему и перейдем в рабочий каталог с проектом:

```
cd /work/sashubina/vagrant
```

Затем запустим виртуальную машину server:

```
vagrant up server
```

На сервере установим fail2ban:

```
dnf -y install fail2ban
```

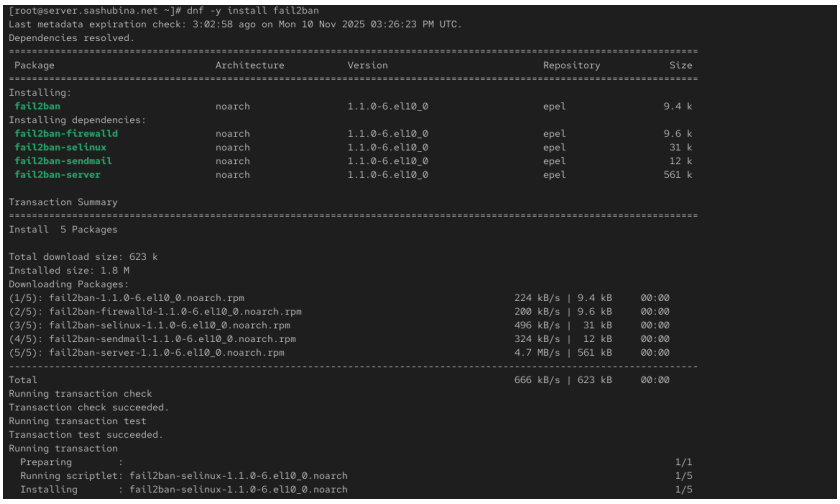

Рис. 3.1: загрузка

Запустим сервер fail2ban:

```
systemctl start fail2ban
systemctl enable fail2ban
```

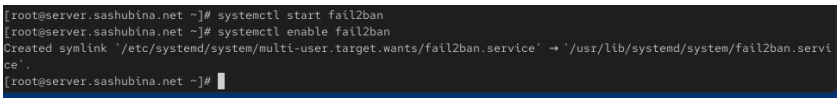

Рис. 3.2: запуск

В дополнительном терминале запустим просмотр журнала событий fail2ban:

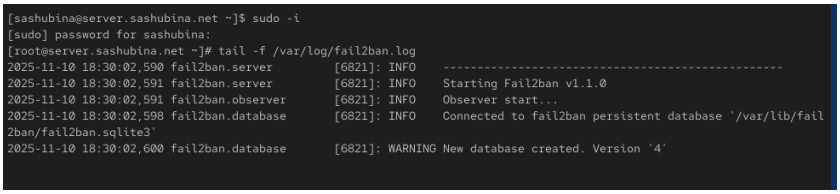

Рис. 3.3: Запуск просмотра журнала событий fail2ban

В логах fail2ban зафиксирован успешный запуск системы защиты: 10 ноября 2025 года в 18:30:02 была инициирована версия 1.1.0, запущен наблюдатель и установлено подключение к базе данных SQLite3. Примечательно, что система создала новую базу данных версии 4, что указывает на первичную настройку или полный сброс предыдущей конфигурации. Процесс старта завершился без ошибок, все компоненты инициализированы корректно

Создадим файл с локальной конфигурацией fail2ban:

```
touch /etc/fail2ban/jail.d/customisation.local
```

Рис. 3.4: создание файла

И в этом файле etc/fail2ban/jail.d/customisation.local зададим время блокирования на 1 час (время задаётся в секундах) и включим защиту SSH:

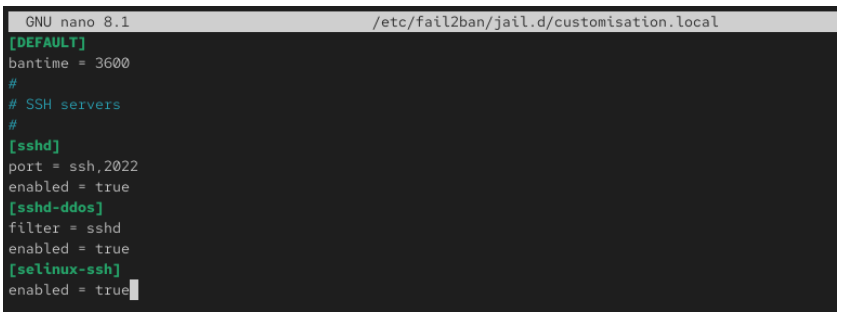

Рис. 3.5: Добавление времени блокировкии и включение защиты SSH customisation.local

Перезапустим сервер fail2ban:

```
systemctl restart fail2ban
```

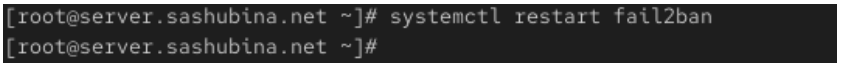

Рис. 3.6: запуск сервера

И посмотрим журнал событий:

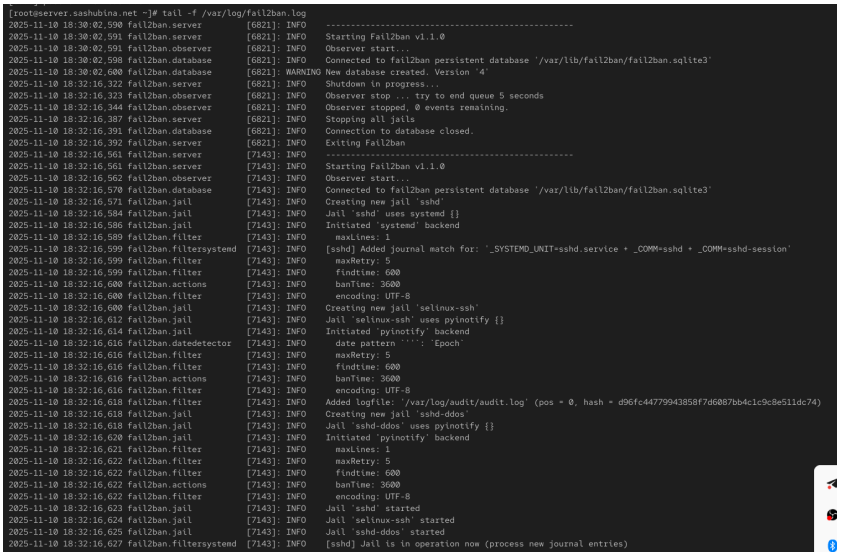

Рис. 3.7: Просмотр журнала событий fail2ban

В логах fail2ban зафиксирован полный цикл перезапуска системы защиты: в 18:30:02 произошла первичная инициализация с созданием новой базы

данных, затем в 18:32:16 выполнен плановый останов службы с корректным завершением всех процессов, после чего сразу осуществлен повторный запуск. В ходе перезапуска успешно активированы три ключевых jail-а: “sshd” для мониторинга системного журнала через systemd, “selinux-ssh” для анализа аудит-логов через inotify и “sshd-ddos” для защиты от DDoS-атак на SSH-сервис. Все компоненты инициализированы без ошибок, фильтры настроены с параметрами maxRetry=5, findtime=600 и bantime=3600, что свидетельствует о полнофункциональной работе системы безопасности.

В файле /etc/fail2ban/jail.d/customisation.local включим защиту HTTP:

Рис. 3.8: Включение защиты HTTP в файле customisation.local

Перезапустим сервер fail2ban:

```
systemctl restart fail2ban
```

Рис. 3.9: запуск сервера

И посмотрим журнал событий:

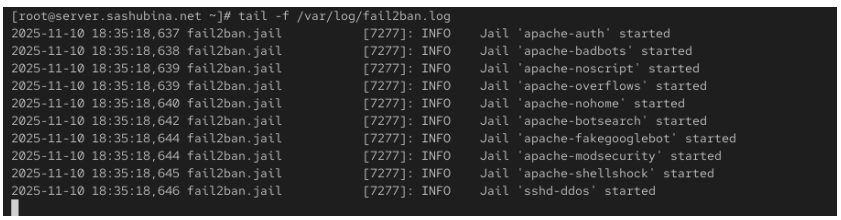

Рис. 3.10: Просмотр журнала событий fail2ban

В логах fail2ban зафиксирован успешный запуск дополнительных защитных модулей для веб-сервера Apache в 18:35:18. Были активированы специализированные jail-ы для мониторинга различных типов угроз: аутентификация (apache-auth), вредоносные боты (apache-badbots), попытки выполнения скриптов (apache-noscript), переполнения буфера (apache-overflows), доступ к несуществующим домашним директориям (apache-nohome), поисковые боты (apache-botsearch), фейковые Google-боты (apache-fakegooglebot), срабатывания ModSecurity (apache-modsecurity) и атаки типа Shellshock (apache-shellshock). Все модули инициализированы без ошибок, что свидетельствует о полноценной защите веб-сервера от широкого спектра сетевых угроз.

В файле /etc/fail2ban/jail.d/customisation.local включим защиту почты:

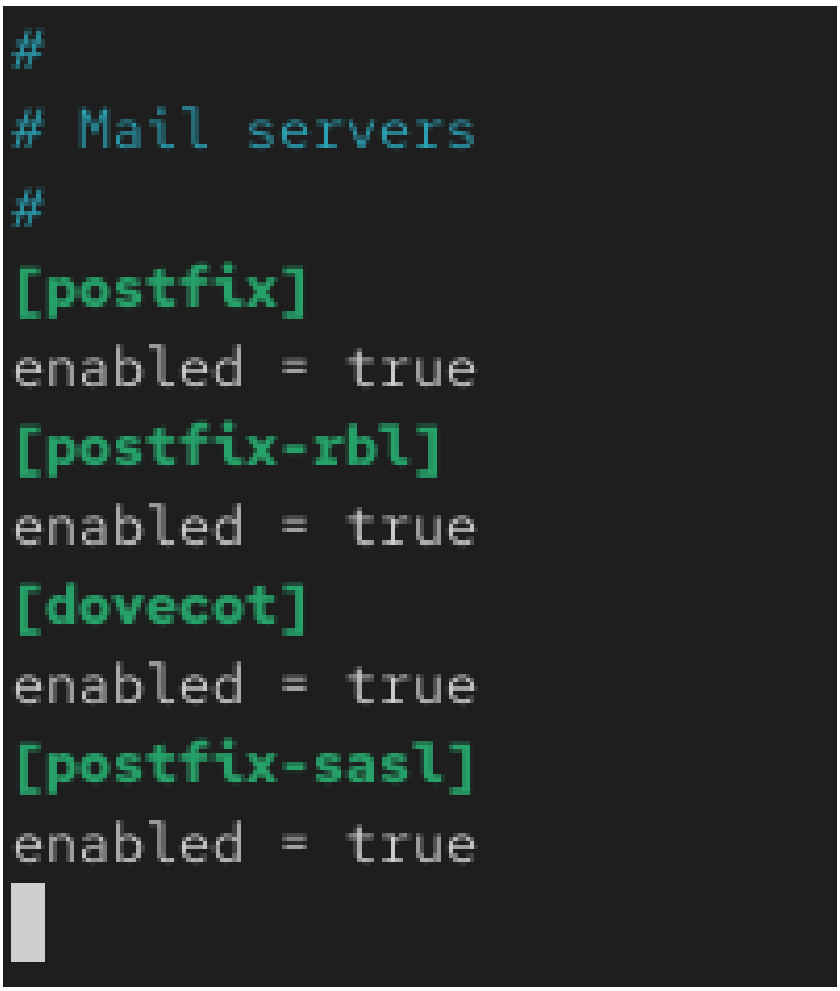

Рис. 3.11: Включение защиты почты в файле customisation.local

Перезапустим сервер fail2ban:

```
systemctl restart fail2ban
```

И посмотрим журнал событий:

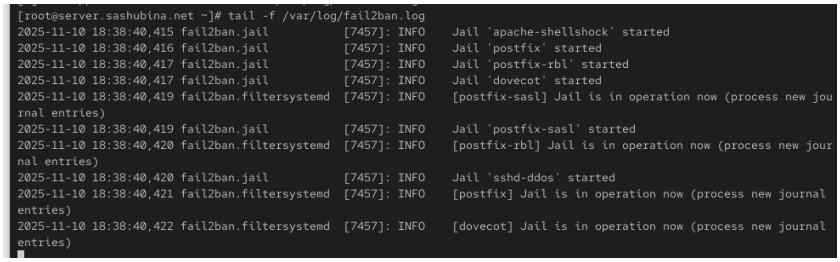

Рис. 3.12: Просмотр журнала событий fail2ban

В логах fail2ban зафиксирован успешный запуск дополнительных защитных модулей для почтовых сервисов и веб-сервера. Были активированы jail-ы для мониторинга почтовой системы Postfix (основной сервис, RBL-фильтрация и SASL-аутентификация), а также для Dovecot (сервис доступа к почтовым ящикам). Дополнительно подтверждена работа ранее настроенных модулей защиты веб-сервера Apache от атак типа Shellshock и SSH от DDoS-атак. Все jail-ы перешли в рабочее состояние и начали мониторинг соответствующих системных журналов, что свидетельствует о полноценной защите критических сетевых сервисов сервера.

3.2 Проверка работы Fail2ban

На сервере посмотрим статус fail2ban и статус защиты SSH в fail2ban

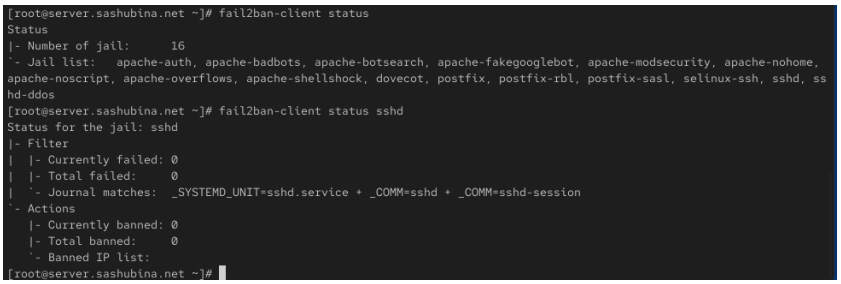

Рис. 3.13: Просмотр статуса fail2ban

Команда fail2ban-client status показывает, что система защиты успешно запущена и активно мониторит 16 различных сервисов, включая веб-сервер Apache через множество специализированных jail-ов (auth, badbots, botsearch и другие), почтовые сервисы Postfix и Dovecot, а также SSH-соединения через три различных фильтра (sshd, selinux-ssh и sshd-ddos). При этом статус конкретного jail-а для SSH показывает, что в данный момент нет заблокированных IP-адресов и не зафиксировано неудачных попыток подключения, что свидетельствует об отсутствии активных атак на сервер в момент проверки. Все компоненты системы функционируют корректно, обеспечивая комплексную защиту сетевых

сервисов.

а затем установим максимальное количество ошибок для SSH, равное 2:

Рис. 3.14: установка количества ошибок

С клиента попытайтесь зайти по SSH на сервер с неправильным паролем:

Рис. 3.15: Попытки соединения по SSH с сервером с неправильным паролем

На сервере посмотрите статус защиты SSH, убедившись, что произошла блокировка адреса клиента:

Рис. 3.16: Проверка блокировки клиента на сервере

В логах fail2ban зафиксирована активная работа системы защиты: в текущий момент jail sshd обнаружил 1 неудачную попытку подключения, а всего за время

работы зафиксировано 3 неудачные аутентификации. В результате срабатывания защитного механизма был заблокирован IP-адрес 192.168.1.30, который в данный момент находится в списке заблокированных. Это демонстрирует корректную работу fail2ban - система успешно обнаруживает подозрительную активность и автоматически применяет блокировку нарушителей согласно заданным правилам безопасности.

Разблокируем IP-адрес клиента и вновь посмотрим статус защиты SSH, убедившись, что блокировка с клиента снята:

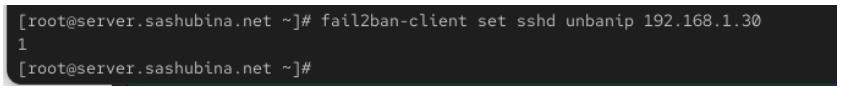

Рис. 3.17: Снятие блокировки с клиента

Вновь посмотрим статус защиты SSH:

```
fail2ban-client status sshd
```

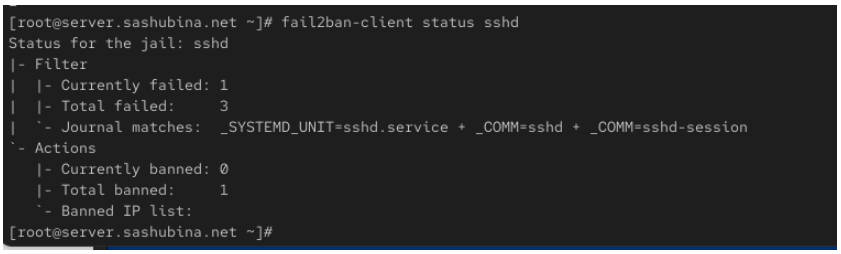

Рис. 3.18: просмотр статуса защиты

В текущем статусе jail sshd видно, что система защиты продолжает фиксировать попытки неудачных подключений (1 активная неудачная попытка и 3 всего), однако IP-адрес 192.168.1.30 больше не находится в списке заблокированных. Это указывает на то, что срок блокировки истек - fail2ban автоматически разблокировал адрес после заданного периода времени (вероятно, через 1 час согласно настройкам bantime = 3600), что демонстрирует корректную работу временных ограничений в системе защиты.

На сервере внесем изменение в конфигурационный файл /etc/fail2ban/jail.d/customisation.lo
добавив в раздел по умолчанию игнорирование адреса клиента:

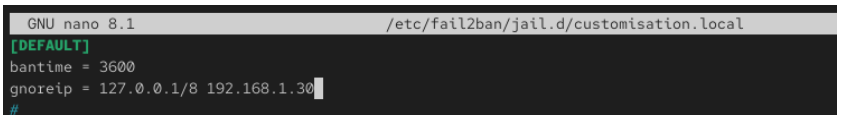

Рис. 3.19: Добавление в конфигурационный файл игнорирования адреса клиента

Перезапустим fail2ban

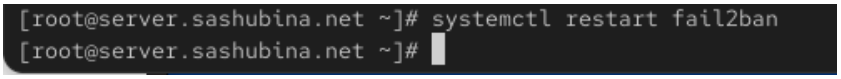

Рис. 3.20: перезапуск fail2ban

посмотрим журнал событий

Рис. 3.21: Просмотр журнала событий fail2ban

Логи fail2ban показывают успешный запуск системы защиты сервера. В 18:57:03 были активированы специализированные модули для мониторинга критических сервисов: apache-shellshock для защиты веб-сервера от уязвимостей Shellshock, postfix, postfix-rbl и postfix-sasl для защиты почтовой системы от атак и спама, dovecot для безопасности почтовых протоколов и sshd-ddos для противодействия DDoS-атакам на SSH-соединения. Все модули перешли в рабочее состояние и начали отслеживать соответствующие системные журналы, что свидетельствует о корректной настройке комплексной защиты серверных служб от основных типов сетевых угроз.

Вновь попытаемся войти с клиента на сервер с неправильным паролем и посмотрим статус защиты SSH:

```
[sashubina@client.sashubina.net ~]$ ssh sashubina@192.168.1.1
sashubina@192.168.1.1's password:
Permission denied, please try again.
sashubina@192.168.1.1's password:
Permission denied, please try again.
sashubina@192.168.1.1's password:
sashubina@192.168.1.1: Permission denied (publickey,gssapi-keyex,gssapi-
with-mic,password).
[sashubina@client.sashubina.net ~]$
```

Рис. 3.22: попытка входа под неправильным паролем

```
[root@server.sashubina.net ~]# fail2ban-client status sshd
Status for the jail: sshd
|- Filter
|  |- Currently failed: 1
|  |- Total failed:     3
|  `- Journal matches:  _SYSTEMD_UNIT=sshd.service + _COMM=sshd + _COMM=sshd-session
`- Actions
   |- Currently banned: 0
   |- Total banned:     0
   `- Banned IP list:
[root@server.sashubina.net ~]#
```

Рис. 3.23: Просмотр статусазащиты SSH после подключение к серверу с клиента по SSH с неправильным паролем

Теперь клиент не блокируется. В текущем статусе jail sshd видно, что система защиты зафиксировала 1 активную неудачную попытку подключения и 3 неудачные попытки всего, однако в данный момент нет заблокированных IP-адресов. Тк мы прописали, что ip клиента не будет блокироваться. IP-адрес клиента теперь в белом списке - fail2ban никогда не заблокирует его. Остальные IP-адреса при подозрительной активности будут блокироваться на 1 час

3.3 Внесение изменений в настройки внутреннего окружения виртуальных машины

На виртуальной машине server перейдем в каталог для внесения изменений в настройки внутреннего окружения /vagrant/provision/server/, создадим в нём

каталог protect, в который поместим в соответствующие подкаталоги конфигурационные файлы, а также создадим исполняемый файл protect.sh:

```
cd /vagrant/provision/server
mkdir -p /vagrant/provision/server/protect/etc/fail2ban/jail.d
cp -R /etc/fail2ban/jail.d/customisation.local /vagrant/provision/server/protect/e

touch protect.sh
chmod +x protect.sh
```

Рис. 3.24: создание файла

В каталоге /vagrant/provision/server создадим исполняемый файл smb.sh и внесем скрипт:

Рис. 3.25: Скрипта файла /vagrant/provision/server/protect.sh

Затем для отработки созданных скриптов в конфигурационном файле Vagrantfile необходимо добавить в соответствующих разделах конфигураций для сервера:

```
server.vm.provision "server protect",
  type: "shell",
```

```
preserve_order: true,
path: "provision/server/protect.sh"
```

```
    server.vm.provision "server protect",
                        type: "shell",
                        preserve_order: true,
                        path: "provision/server/protect.sh"

end
```

Рис. 3.26: Vagrantfile

4 Контрольные вопросы

1. Поясните принцип работы Fail2ban.
2. Настройки какого файла более приоритетны: jail.conf или jail.local?
3. Как настроить оповещение администратора при срабатывании Fail2ban?
4. Поясните построчно настройки по умолчанию в конфигурационном файле /etc/fail2ban/jail.conf, относящиеся к веб-службе.
5. Поясните построчно настройки по умолчанию в конфигурационном файле /etc/fail2ban/jail.conf, относящиеся к почтовой службе.
6. Какие действия может выполнять Fail2ban при обнаружении атакующего IP-адреса? Где можно посмотреть описание действий для последующего использования в на- стройках Fail2ban?
7. Как получить список действующих правил Fail2ban?
8. Как получить статистику заблокированных Fail2ban адресов?
9. Как разблокировать IP-адрес?
10. Fail2ban - это программное обеспечение, которое предотвращает атаки на сервер, анализируя лог-файлы и блокируя IP-адреса, с которых идут подозрительные или злонамеренные действия. Он работает следующим образом:

- Мониторит указанные лог-файлы на наличие заданных событий (например, неудачных попыток входа).
- Когда число попыток превышает определенный порог, Fail2ban временно блокирует IP-адрес, добавляя правила в файрвол.
- Заблокированный IP-адрес может быть разблокирован автоматически после определенного периода времени.

11. Настройки файла jail.local более приоритетны, чем настройки файла jail.conf. Если в файле jail.local определены одни и те же параметры, они будут использованы вместо параметров из jail.conf.

12. Чтобы настроить оповещение администратора при срабатывании Fail2ban, необходимо настроить отправку уведомлений по электронной почте или другим способом. Это можно сделать, изменяя настройки в файле jail.local, добавляя адрес электронной почты администратора и настройки SMTP-сервера.

13. Примеры настроек по умолчанию в конфигурационном файле /etc/fail2ban/jail.conf, относящиеся к веб-службе:

    - `[apache]` - секция, относящаяся к веб-серверу Apache.
    - `enabled = true` - включение проверки лог-файлов Apache.
    - `port = http,https` - указание портов для мониторинга.
    - `filter = apache-auth` - указание фильтра для обработки лог-файлов.
    - `logpath = /var/log/apache*/*error.log` - путь к лог-файлам Apache.
    - `maxretry = 5` - максимальное количество попыток до блокировки адреса.
    - `bantime = 600` - продолжительность блокировки в секундах.

14. Примеры настроек по умолчанию в конфигурационном файле /etc/fail2ban/jail.conf, относящиеся к почтовой службе:

- `[postfix]` - секция, относящаяся к почтовому серверу Postfix.
- `enabled = true` - включение проверки лог-файлов Postfix.
- `port = smtp,ssmtp` - указание портов для мониторинга.
- `filter = postfix` - указание фильтра для обработки лог-файлов.
- `logpath = /var/log/mail.log` - путь к лог-файлам Postfix.
- `maxretry = 3` - максимальное количество попыток до блокировки адреса.
- `bantime = 3600` - продолжительность блокировки в секундах.

15. Fail2ban может выполнять различные действия при обнаружении атакующего IP-адреса, такие как блокировка адреса через файрвол, добавление правил в IP-таблицы, отправка уведомлений администратору и другие. Описание доступных действий можно найти в документации или руководстве Fail2ban.

16. Для получения списка действующих правил Fail2ban можно использовать команду: `fail2ban-client status`.

17. Для получения статистики заблокированных адресов Fail2ban можно использовать команду: `fail2ban-client status <jail-name>`, где `<jail-name>` - имя конкретного jail, например, “ssh” или “apache”.

18. Разблокировать адрес можно с помощью следующей команды

```
fail2ban-client set sshd unbanip <ip-адрес клиента>
```

5 Выводы

В результате выполнения данной работы были приобретены практические навыки работы с программным средством Fail2ban для обеспечения базовой защиты от атак типа «brute force».