POSITIVE TECHNOLOGIES

ЗАО «ПОЗИТИВ ТЕКНОЛОДЖИЗ»
107061, МОСКВА, ПРЕОБРАЖЕНСКАЯ ПЛ., Д. 8
ТЕЛ. +7 495 744-01-44, ФАКС +7 495 744-01-87, PT@PTSECURITY.COM
PTSECURITY.RU, MAXPATROL.RU, SECURITYLAB.RU

СИСТЕМА ЗАЩИТЫ ПРИЛОЖЕНИЙ ОТ НЕСАНКЦИОНИРОВАННОГО ДОСТУПА POSITIVE TECHNOLOGIES APPLICATION FIREWALL

РУКОВОДСТВО АДМИНИСТРАТОРА

Версия 3.3

ОГЛАВЛЕНИЕ

1. Особенности PT Application Firewall

Система защиты приложений от несанкционированного доступа Positive Technologies Application Firewall[1] - самообучающийся динамический межсетевой экран, способный снижать риски атак на приложение при их появлении. PT Application Firewall сочетает традиционные методы черного и белого списков с новейшим подходом самообучения. При помощи эвристических алгоритмов межсетевой экран изучает особенности трафика и деятельности пользователей, взаимодействующих с бизнес-приложениями. Данные, отражающие стандартные действия пользователей, используются для обнаружения потенциальных атак и отклонений от поведения типичного пользователя.

Механизм нормализации позволяет обрабатывать HTTP-запросы с учетом специфики защищаемого веб-сервера, что снижает риск применения таких атак, как HTTP Parameter Contamination (HPC) и HTTP Parameter Pollution (HPP). PT Application Firewall учитывает тип веб-сервера, с которым взаимодействует, и впоследствии симулирует его поведение, что дает более эффективную защиту от целенаправленных атак.

PT Application Firewall имеет продвинутые алгоритмы, направленные на защиту от фрода и роботов, использующие поведенческий анализ для обнаружения признаков атак подбора, нехарактерной активности и попыток полного копирования сайта.

Основные возможности:

- высокий уровень защиты при минимальном количестве ложных срабатываний;
- моментальная реакция на угрозы с помощью виртуальных обновлений до устранения уязвимости;

  Примечание: данная функция особенно эффективна при использовании совместно с PT Application Inspector, который имеет механизм автоматической генерации эксплойтов и полностью интегрируется с PT Application Firewall.
- блокирование современных атак, таких как HPP, HPC и XXE;
- противодействие брутфорс-атакам, защита от фрода и роботов;
- быстрое обучение на основе журналов событий;
- анализ журналов веб-сервера при расследовании инцидентов;
- защита от атак нулевого дня;
- обнаружение DDoS-атак на уровне приложения;
- интеграция с системами для защиты от DDoS, решениями DLP и антивирусами;
- гибкие модели развертывания: программно-аппаратный комплекс, виртуальный программно-аппаратный комплекс;
- поддержка SSL-ускорителя;
- соответствие требованиям PCI DSS;
- защита внутреннего и внешнего XML-трафика.

1. Далее по тексту: PT AF.

1.1. SOA Firewall

PT AF SOA Firewall позволяет защитить внутренний и внешний XML-трафик.

Протоколы на основе языка XML (SOAP, HTTP, REST) используются сегодня в самых различных отраслях: в банках, телекомах, промышленности, здравоохранении и на транспорте. Благодаря универсальности и гибкости сервис-ориентированные платформы (SOA) позволяют связывать самые разнородные бизнес-приложения в критически важных инфраструктурах, включая ERP, АСУ ТП, АБС, порталы госуслуг и СМЭВ.

```
1 GET /api/xml.php HTTP/1.1
2 Host: ptdemo.com
3 Content-Type: text/xml
4 Content-Length: 152
5
6 <?xml version="1.0" encoding="ISO-8859-1"?>
7  <!DOCTYPE foo [
8    <!ELEMENT foo ANY >
9    <!ENTITY xxe SYSTEM "file:///etc/passwd" >]>
```

Рис. 1 –

Распространенность XML делает особенно опасными уязвимости, связанные с возможностями этого языка. В 2014 году в 46% систем онлайн-банкинга обнаружена критическая уязвимость «Внедрение внешних сущностей XML», которая позволяет злоумышленнику получить содержимое файлов на атакуемом сервере. А техника атак XXE OOB позволила экспертам Positive Technologies обнаружить уязвимости в продуктах Microsoft, Oracle, ModSecurity и в компонентах SCADA-систем Siemens. При этом классические межсетевые экраны пакетного уровня не ориентированы на выявление таких угроз.

Новый модуль XML-анализа (SOA Firewall) позволяет противодействовать атакам на распределенные веб-сервисы любого уровня сложности, выявляя и блокируя не только типовые веб-угрозы, но и попытки взлома с использованием специфических XML-уязвимостей.

1.1.1. Особенности PT AF SOA Firewall

К основным возможностям SOA Firewall относятся:

- Защита на уровне XML/SOAP. Фильтрация запросов путем проверки валидности XML-документа: проверка на соответствие RFC (Well-formed XML), анализ на соответствие XSD-схемам (Valid XML).
- Самообучение для блокирования неизвестных атак: обучение модуля HMM на XML-запросах позволяет составить позитивную модель приложения и блокировать аномальные запросы.
- Поведенческий анализ против роботов (сканнеры, подбор паролей, фрод, ботнеты) благодаря выявлению аномалий в действиях пользователей.
- Защита от DDoS и XML Dos (XDoS). Проверка внешних сущностей в тегах <!ENTITY> и проверка сущностей в тегах <!DOCTYPE>.
- Защита от типовых веб-атак: поиск в ключах и значениях XML-документа сигнатур таких атак, как SQLi, XSS, Path Traversal, Remote Code Execution, Local/Remote File Inclusion и др.

• Предотвращение утечек: анализ ответов сервера на наличие чувствительной информации.

• Защита от внутренних угроз: разграничение прав доступа в соответствии с политиками безопасности, блокировка нелегитимных запросов со стороны сотрудников, контроль доступа к конфиденциальным документам и верификация контента.

• Высокая производительность и масштабируемость:

1.1.2. Модели использования SOA Firewall

Возможные сферы применения XML-версии защитного экрана включают:

• Системы интеграции бизнес-приложений крупных компаний (ERP, SAP, CRM), обмен данными как внутри одной распределенной компании, так и между разными организациями;

• СМЭВ: взаимодействие между министерствами, ситуационными центрами и государственными сервисами, а также между подразделениями внутри ведомств;

• Банки: взаимодействие между автоматизированной банковской системой и веб-серверами;

• СМИ и широко используемые веб-приложения: взаимодействие с рекламными и обменными сетями.

• Страховые компании и электронные полисы: взаимодействие с базами данных.

• Ритейловые сети: обмен данными с поставщиками.

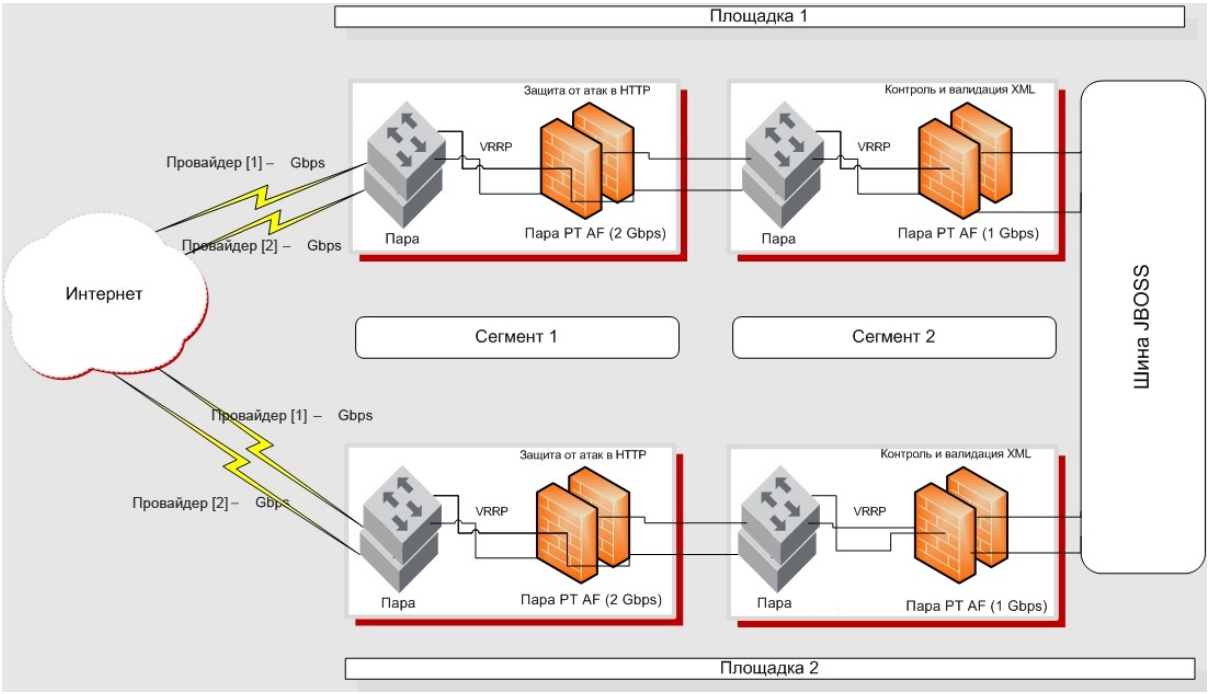


Рис. 2 – Модель применения защитного экрана

1.1.3. Модули защиты

Система автоматически разбирает и обрабатывает XML-документ, выполняя проверку каждого элемента всеми модулями защиты.

Модуль *Защита XML* осуществляет многочисленные проверки XML-атак всех типов. Цель проверок – предотвращение использования злоумышленником небезопасно сформированный XML-запрос в приложении или веб-службе. До начала проверки следует удостовериться, что заголовок Content-Type с типом text/xml содержится в документе.

Данный модуль проверяет, удовлетворяет ли документ требованиям XML-формата, так как неправильно сформированный XML-документ может привести к нарушению безопасности системы. В XML-документе необходимо осуществить следующие проверки для обеспечения корректной обработки входящих XML-запросов:

- отсутствие спецсимолов <,> и &, не использующихся для XML-разметки;
- в документ включены только Unicode символы;
- все теги правильно вложены, в начале и конце тега соблюдается регистр символов;
- корневой элемент содержит прочие элементы документа.

Также модуль позволяет предотвратить атаку на XML-вложение, которая может привести к нарушению безопасности на сервере.

Дополнительно в модуле *Защита XML* проверяются XSD-схемы, загружаемые пользователем (см. главу «XML-схемы» и «Защита XML»). Цель подобных проверок - предотвращение нарушения безопасности приложения злоумышленником, использующим специально сконструированные недопустимые XML-сообщения. То есть модуль осуществляет валидацию запросов к XML-API по загруженным схемам.

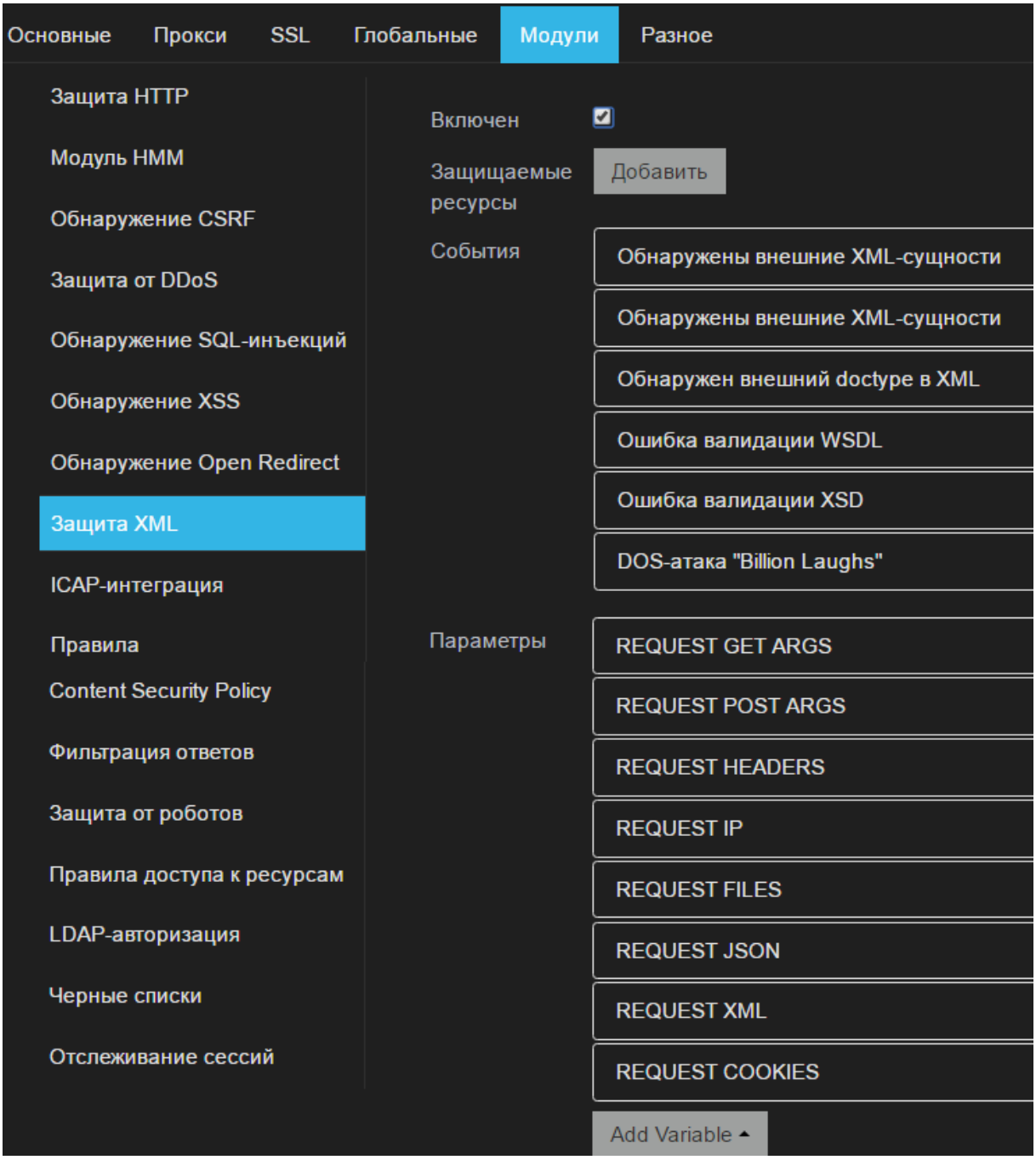


Рис. 3 – Защита XML

Модуль *Обнаружение XSS* осуществляет проверку XSS (Cross-Site Scripting), то есть рассматривает как заголовки, так и тело запросов пользователей на возможные атаки межсайтового скриптинга. Если возникает подозрение в атаке Cross-Site Scripting, то модуль блокирует запрос.

Модуль позволяет предотвратить атаку на веб-системы, заключающуюся во внедрении в выдаваемую веб-системой страницу вредоносного кода и взаимодействии внедренного кода с веб-сервером злоумышленника.

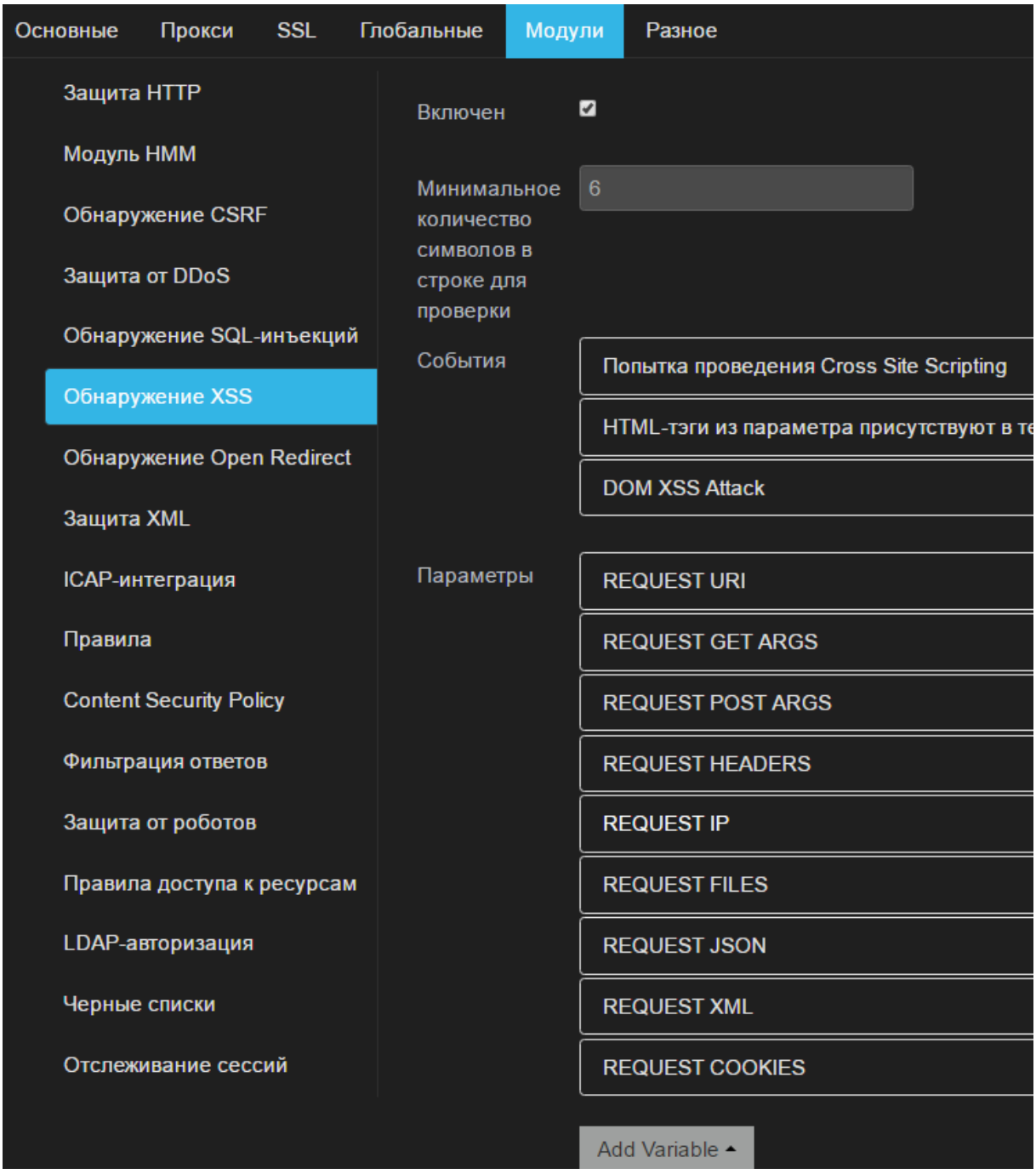


Рис. 4 – Модуль обнаружения XSS

Модуль *Обнаружение SQL-инъекции* рассматривает как заголовки, так и тело запросов пользователей на возможные атаки типа SQL-инъекций. Если возникает подозрение в SQL-инъекции, то модуль блокирует запрос.

Модуль позволяет предотвратить атаку на веб-системы, заключающуюся во взломе сайта или программы, работающей с базами данных, то есть блокирует внедрение в запрос произвольного SQL-кода, который, например, позволил бы злоумышленнику прочитать

содержимое любых таблиц, удалить, изменить или добавить данные, выполнить произвольные команды на атакуемом сервере.

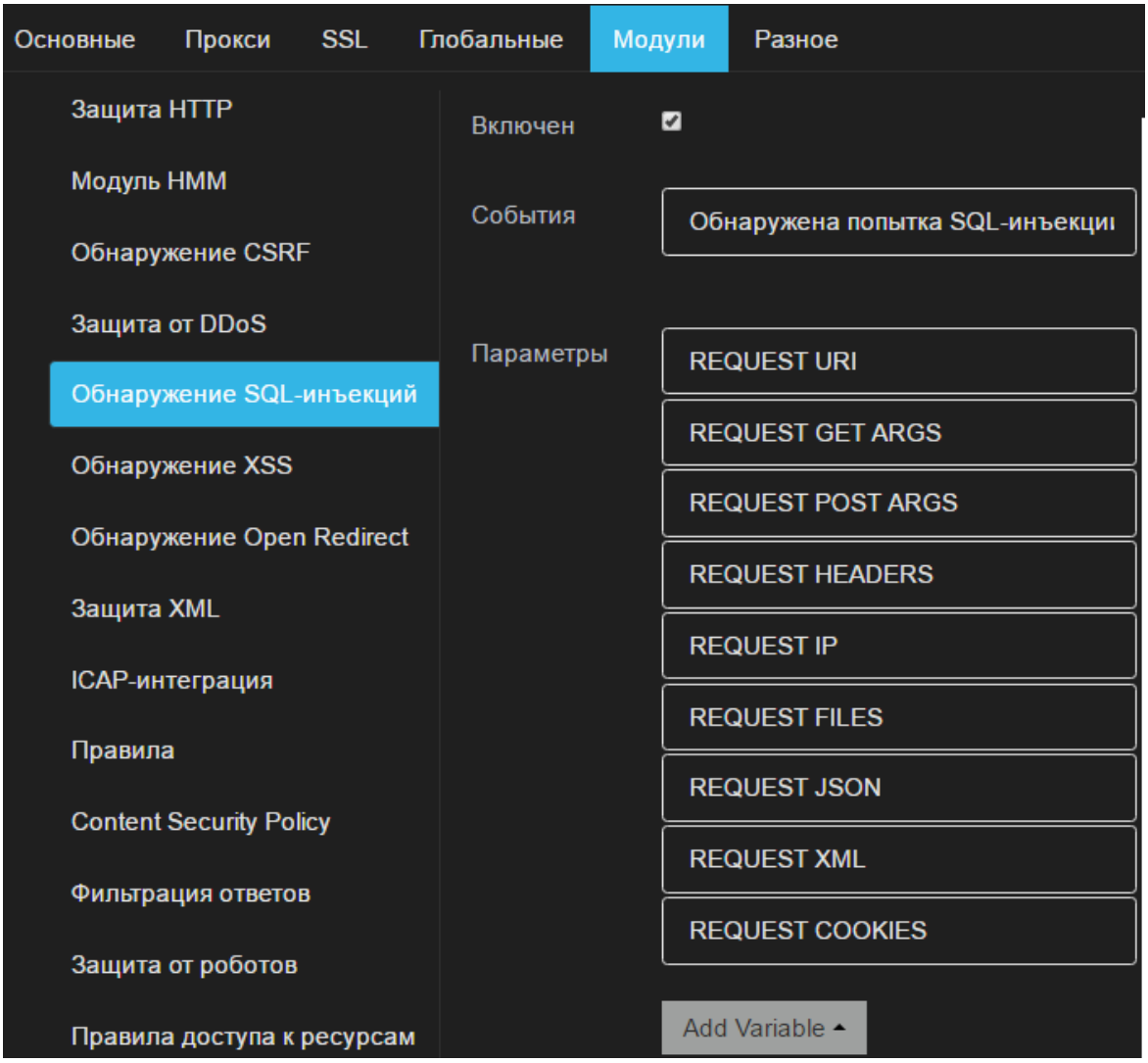


Рис. 5 – Модуль обнаружения SQL-инъекции

Если злоумышленник пытается загрузить свои файлы на сервер, то срабатывает модуль *ICAP-интеграция*. Подробную информацию о модуле можно найти в главе «ICAP-интеграция».

Модуль *Фильтрация ответов* проверяет взаимодействие веб-служб. Злоумышленник может использовать противоречия в совместимости, чтобы начать атаку на приложение XML. Запросы, которые не отвечают требованиям стандарта взаимодействия, блокируются. Дополнительно модуль позволяет предотвратить утечку конфиденциальной информации злоумышленнику, отфильтровывая XML SOAP-ошибки.

2. Технические характеристики моделей

В Табл. 1 указаны следующие характеристики программно-аппаратных комплексов PT AF:

- RPS (Requests Per Second) – количество HTTP-запросов на PTAF в секунду. На вкладке *Мониторинг* размещен график *Proxy Requests Per Second*, который предназначен для контроля производительности системы;
- CC (Concurrent Connection) – количество открытых одновременных (параллельных) пользовательских соединений. Под соединением понимается любое соединение с завершенным процессом трехэтапного согласования (3WHS) и хотя бы с одним обработанным HTTP-запросом и ответом;
- CPS (Connection Per Second) – число соединений в секунду. В данном случае под соединением подразумевается процесс трехэтапного согласования TCP (SYN,SYN-ACK и ACK), сопровождаемого запросом HTTP GET и HTTP Response с последующим закрытием TCP-соединения (методом RESET или FIN). Данный процесс повторяется для каждой HTTP-транзакции. Из чего следует, что установка и завершение TCP-соединения происходят в каждой HTTP-транзакции;
- TPS (Transaction Per Second) – число транзакций в секунду. Здесь под транзакцией подразумевается процесс трехэтапного согласования TCP (SYN,SYN-ACK и ACK), сопровождаемого многочисленными HTTP-транзакциями (GET & RESPONSE, GET & RESPONSE и т.д.) с последующим закрытием TCP-соединения. Для данного параметра установка и завершение TCP-соединения происходят не в каждой HTTP-транзакции.

Примечание: количество транзакций в секунду обычно больше чем количество соединений в секунду, и параметры TPS и CPS не сопоставимы.

Таблица 1. Показатели производительности продуктов PT AF

	PTAF-CH50	PTAF-CH100	PTAF-CH200-N1	PTAF-CH200-N2	PTAF-CH300
Максимальные значения параметров производительности для HTTP-трафика					
Количество HTTP-запросов/транзакций в секунду (RPS/TPS)	3700	22000	30000	40000	100000
Количество HTTP-соединений в секунду (CPS)	3000	15000	21000	28000	70000
Максимальные значения параметров производительности для HTTPS-трафика					
Количество HTTPS-запросов/транзакций в секунду (RPS/TPS)	2300	13000	16000	20000	55000
Количество HTTPS-соединений в секунду (CPS)	1000	7000	9000	13000	30000

Если трафик превышает указанную пропускную способность, то работоспособность PT AF не нарушается, но часть трафика будет передаваться без обработки.

2.1. SSL-ускоритель (SSL accelerator)

Для увеличения производительности системы и снижения потребления процессорных ресурсов протоколом SSL к комплексу PT AF можно подключить SSL-ускоритель. Использование SSL-ускорителя на младших моделях PT AF позволяет значительно повысить производительность. Для HTTPS-трафика прирост производительности составляет до 25%.

SSL accelerator представляет собой PCI-карту с поддержкой qat (Intel® QuickAssist Technology), которая вместо CPU сервера выполняет задачи криптования, тем самым разгружает CPU при большом количестве https-запросов под высокой нагрузкой.

Установка SSL-ускорителя опциональна и предлагается в расширенной конфигурации.

Интеграция SSL-ускорителя происходит автоматически при установке PT AF. Установщик самостоятельно добавляет необходимые драйвера, модули и библиотеки. Дополнительно появляется служба waf-nginx-qat.

Таблица 2. Максимальные значения параметров производительности при использовании SSL-акселератора

	PTAF-CH100	PTAF-CH200-N1	PTAF-CH200-N2	PTAF-CH300
Количество HTTPS-запросов/ транзакций в секунду (RPS/TPS)	15000	20000	25000	65000
Количество HTTPS-соединений в секунду (CPS)	10000	13000	18000	40000

3. Использование системы в виртуальной среде

Система защиты веб-приложений может использоваться в виртуальной среде с привязкой к электронным ключам Guardant.

Использование ключа Guardant обеспечивает лицензионную защиту компонентов PT AF при работе в виртуальном окружении. На текущий момент такая возможность официально реализована только для виртуальных машин VMware ESXI 4.1 и Workstation версии 6.0 и выше.

При использовании других виртуальных машин эта возможность реализуется с помощью программного или аппаратного обеспечения типа USB over IP Network.

Компания ЗАО «Позитив Текнолоджиз» обеспечивает техническую поддержку решений, которые базируются только на виртуальных машинах VMware ESXI 4.1 и Workstation версии 6.0 и выше. Если вы используете решения типа USB over IP Network, по всем вопросам, связанным с работой Guardant, необходимо обращаться в службу технической поддержки компании-разработчика этого программного обеспечения.

Наличие постоянно подключенного ключа Guardant является обязательным условием для работы PT AF на виртуальной машине.

Допускается использование только ключей Guardant, выданных ЗАО «Позитив Текнолоджиз».

3.1. Порядок использования PT AF с Guardant

Для использования системы PT AF в виртуальной среде необходимо запросить виртуальную машину и соответствующую лицензию. Предоставление права использования PT AF осуществляется по лицензионным договорам. Одновременно с передачей права использования PT AF клиенту передается ключ Guardant.

Ключ Guardant передается клиенту в безвозмездное пользование. В случае утраты ключа Guardant по запросу клиента выпускается дополнительный ключ Guardant на безвозмездной основе. В этом случае доставка ключа Guardant клиенту осуществляется за его счет и с использованием его ресурсов.

В случае повторной и последующей утраты ключа Guardant клиент приобретает дополнительный ключ Guardant самостоятельно у авторизованных партнеров (за контактами необходимо обратиться в службу технической поддержки) и передает разработчику для его инициализации. Получение ключа Guardant после инициализации осуществляется за счет клиента и с использованием его ресурсов.

Клиенты, которые ранее приобрели систему PT AF и у которых есть необходимость перейти на использование PT AF в виртуальной среде, обращаются в техническую поддержку ЗАО «Позитив Текнолоджиз» с запросом на переоформление действующей лицензии. Одновременно с оформлением соответствующей лицензии формируется ключ Guardant. Формирование ключа осуществляется бесплатно. Доставка ключа Guardant клиенту осуществляется курьером компании ЗАО «Позитив Текнолоджиз» или, по договоренности, курьером компании клиента.

3.2. Импорт виртуальной машины

Подключите VMware vSphere клиент и разверните виртуальную машину, выбрав пункт меню *File - Deploy OVF Template...* (см. Рис. 6).

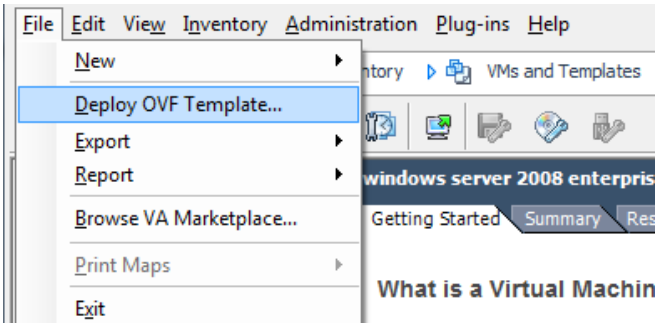


Рис. 6 – Импорт виртуальной машины

Затем укажите путь к шаблону, который был выдан компанией ЗАО «Позитив Текнолоджиз» и нажмите кнопку *Next*.

Выберите параметры, соответствующие вашей виртуальной инфраструктуре.

3.3. Установка компонентов системы с использованием Guardant

После получения дистрибутива и ключа Guardant можно приступить к развертыванию PT AF в виртуальном окружении. Алгоритм развертывания следующий:

1. Подготовьте виртуальную машину на базе VMware ESXI 4.1 или Workstation версии 6.0 и выше.
2. Установите PT AF из полученного дистрибутива.

   Примечание: процесс установки необходимо проводить при отключенном ключе Guardant.
3. Драйверы для ключа Guardant интегрированы в дистрибутив и будут установлены автоматически.
4. Подключите полученный ранее ключ Guardant к USB-порту сервера VMware.
5. В разделе *Settings* созданной виртуальной машины последовательно подключите USB controller и USB device. Если к VMware подключено несколько электронных ключей, выберите ключ, полученный от ЗАО «Позитив Текнолоджиз».

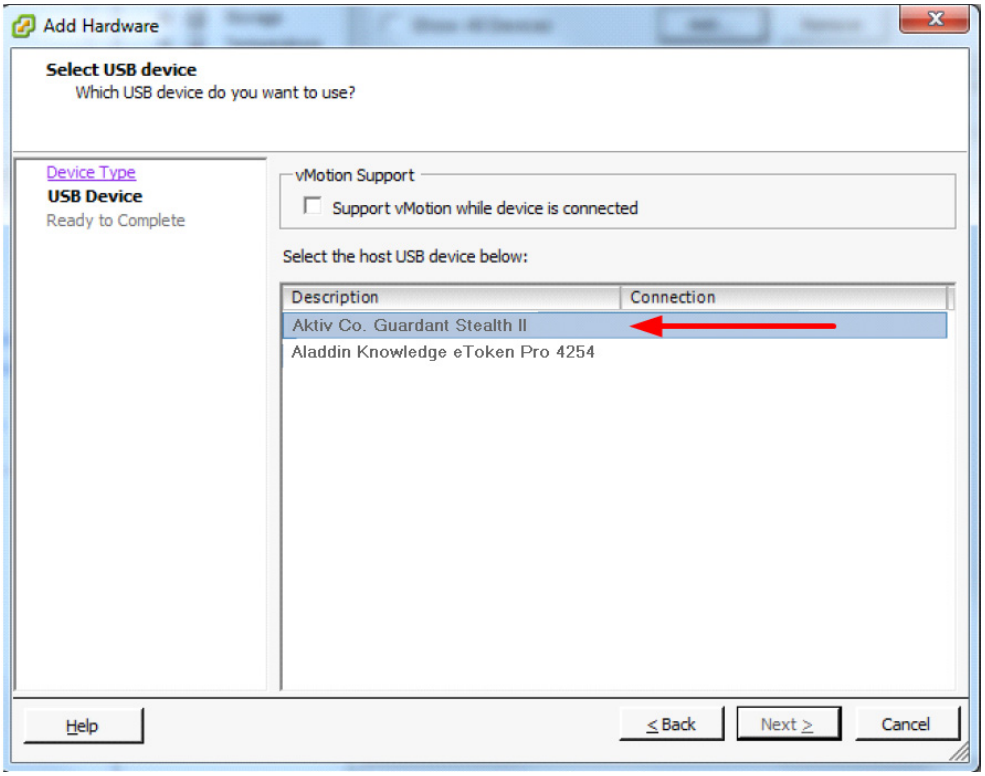


Рис. 7 – Выбор электронного ключа

В процессе активации проверяется наличие подключенного ключа Guardant и действительной лицензии. Для сохранения работоспособности PT AF требуется наличие постоянно подключенного ключа Guardant. В случае отключения ключа Guardant PT AF блокирует возможность работы с системой до подключения ключа Guardant.

4. Настройка сети

Для базовой настройки сетевых интерфейсов следует использовать конфигурационный скрипт. После его запуска открывается интуитивно понятная оболочка для базовой конфигурации системы - CLI (Command Line Interface), имеющая название wsc в системе PT AF. Запуск производится таким образом: sudo wsc

Настройки в плоскости локальных изменений хранятся в SQLite базе данных, файл базы данных расположен по следующему пути: /opt/waf/conf/local_config.sqlite3

Минимально необходимый набор действий, который следует совершить для базового конфигурирования:

- назначение интерфейсам подходящего режима работы;
- установка правильных IP-адресов и масок подсети;
- интерфейсы отмечаются командой mark, после чего становятся доступны в UI[2];
- применение конфигурации.

После базовой настройки возможно подключение к веб-интерфейсу PT AF, откуда следует производить дальнейшую конфигурацию. Для этого с помощью протокола HTTP необходимо обратиться на назначенный на интерфейс IP-адрес по 80 порту. Если подключение не происходит, остановите встроенный «firewall» (service ferm stop). После чего укажите в UI, какой интерфейс будет использоваться для целей менеджмента, а затем включите «firewall».

4.1. Команды

Команды в CLI делятся на несколько типов: служебные, информационно-редактирующие и секционные.

- Служебные команды выводят техническую информацию и позволяют управлять самим CLI: info, help, exit, output, config, snapshot. Служебные команды help, exit, output доступны из любой секции.
- Информационно-редактирующие команды позволяют просмотреть информацию и отредактировать ее. Сюда относятся команды dns, hostname и timezone.
- Секционные команды позволяют как попадать в отдельные конфигурационные секции, если задана только команда, так и просматривать и редактировать настройки секции, если задана команда с аргументами. К этой категории относятся все остальные команды.

В CLI работают сочетания клавиш Ctrl-D (выход), Ctrl-R (история команд) и Tab (автодополнение).

Некоторые полезные команды представлены в Табл. 3.

2. UI – пользовательский графический интерфейс PT Application Firewall.

Таблица 3. Команды интерактивной оболочки

Команда	Описание
hostname	Показать текущий hostname
hostname <name>	Установить hostname
host -> add	Добавить в hosts запись
host -> del	Удалить из hosts запись
if -> mode <INT> <static, manual, dhcp>	Изменить режим интерфейса
if -> vlan <INT> <tag>	Создать Саб-интерфейс 802.1q
if -> set inet_addr <addr> inet_netmask <netmask>	Назначить интерфейсу IP-адрес и маску
if -> mark <name>	Сделать интерфейс видимым в UI. Принимает список названий интерфейсов. Пример: if mark eth1 eth2 eth3
info	Показать сведения о системе
route add <network> <param1> <value1> [<paramN> <valueN>]	Добавить маршрут
route -> del <destination>	Удалить маршрут из таблицы main
route -> del <routing_table> <destination>	Удалить маршрут из таблицы
snapshot restore	Восстановить последнюю примененную конфигурацию
<section> revert	Восстановить последнюю примененную конфигурацию в пределах определенной секции. Команда может вызывать ошибки конфигурации, для полного восстановления настроек используйте команду snapshot restore
list	Показать сохраненную конфигурацию. Работает только в пределах определенной секции
config	Показать текущую конфигурацию
config commit	Применить конфигурацию. Конфигурация сохраняется после каждой команды, связанной с добавлением, удалением или изменением свойств объектов. Часть конфигурации сохраняется в MongoDB командой "config sync"
config sync	Сохранить конфигурацию для отображения в UI
help, help <command>	Помощь.

4.1.1. Пример создания кластерной конфигурации

На первом сервере запускается файл **ptaf-cl1.sync**.

```
#!/usr/bin/env wsc
# Run this from root (e.g., sudo ./ptaf-cl1.sync)

# Output defaults to none when executed from file
output changes

# Datetime configuration
timezone Europe/Moscow
ntp add 0.pool.ntp.org 1.pool.ntp.org 2.pool.ntp.org 3.pool.ntp.org

# Interface configuration
if bond 0 eth0
if set bond0 lacp_rate 1

if vlan bond0 1060 1070 1080

if set vlan1060 inet_addr 172.16.60.1 inet_netmask 255.255.255.0
if set vlan1070 inet_addr 172.16.70.1 inet_netmask 255.255.255.0
if set vlan1080 inet_addr 172.16.80.1 inet_netmask 255.255.255.0

# Hostname configuration
host add 172.16.60.1 ptaf-cl1
host add 172.16.60.2 ptaf-cl2

hostname ptaf-cl1

# Cluster configuration
cluster set elastic nodes ptaf-cl1 ptaf-cl2
cluster set elastic replset waf

# Specify nodes on master-node only.
# Arbiter is created automatically if overall quantity (including local
node) is even.
cluster set mongo local ptaf-cl1
cluster set mongo nodes ptaf-cl2
cluster set mongo replset waf

# Commit and sync to mongo
config commit
config sync
```

На втором сервере запускается файл **ptaf-cl2.sync**.

```
#!/usr/bin/env wsc
# Run this from root (e.g., sudo ./ptaf-cl2.sync)

# Output defaults to none when executed from file
output changes

# Datetime configuration
timezone Europe/Moscow
ntp add 0.pool.ntp.org 1.pool.ntp.org 2.pool.ntp.org 3.pool.ntp.org

# Interface configuration
if bond 0 eth0
if set bond0 lacp_rate 1

if vlan bond0 1060 1070 1080

if set vlan1060 inet_addr 172.16.60.2 inet_netmask 255.255.255.0
if set vlan1070 inet_addr 172.16.70.2 inet_netmask 255.255.255.0
if set vlan1080 inet_addr 172.16.80.2 inet_netmask 255.255.255.0

# Hostname configuration
host add 172.16.60.1 ptaf-cl1
host add 172.16.60.2 ptaf-cl2

hostname ptaf-cl2

# Cluster configuration
cluster set elastic nodes ptaf-cl1 ptaf-cl2
cluster set elastic replset waf

cluster set mongo local ptaf-cl2
cluster set mongo replset waf

# Commit and sync to mongo
config commit
config sync
```

4.1.2. Служебные команды (info, help, exit, output, config, snapshot)

info

Команда позволяет просмотреть информацию о системе, включая версию ядра, libc, версию UI и текущее локальное время.

```
# Command usage
info
```

help

Без аргументов команда выводит перечень доступных команд из корневой секции, если выполнена из корневой секции, или перечень команд той секции, из которой она выполнена.

Если в команде указан второй аргумент, то будет выведена справка по команде, указанной в аргументе. Если указана секция, то будет выведена справка по всем командам секции.

Применение команды help:

```
# List all help topics in current section
help

# Get help for "if" section
help if

# Get help for "dns" command in root section
help dns

# Get help for "list" command in "if" section
if help list
```

exit

Команда позволяет выйти из секции, если выполнена в ней, или из оболочки CLI, если выполнена в корневой секции. Для выхода из оболочки также можно использовать сочетание клавиш Ctrl-D.

```
# Command usage
exit
```

output

Меняет режим вывода информации на экран. Доступны следующие режимы:

- all - выводит конфигурацию секции в плоскости локальных изменений и системной плоскости;
- local - выводит конфигурацию секции в плоскости локальных изменений после каждой команды;
- changes - выводит конфигурацию секции в плоскости локальных изменений только после выполнения информационно-редактирующих команд, команд секций с

аргументом list или команд list при нахождении внутри секции отличной от корневой. Этот режим включен по умолчанию;

• none - не выводит конфигурацию секции;

Примечание: при совершении изменений (commit) информация о событиях всегда выводится на экран.

• automation - при возникновении любых ошибок происходит аварийный выход из CLI.

Применение команды output:

```
# Output local configuration after each command
output local

# Output local and system configuration after each command
output all

# Output local configuration only on "list" command
output changes

# Output nothing
output none

# Exit CLI in case of any error
output automation
```

config

Выполненная без аргументов, команда позволяет посмотреть общую текущую конфигурацию в плоскости локальных изменений. Если настройка системы позволяет, и текущая конфигурация при выводе превышает размер экрана, то используется пагинатор.

С аргументом commit команда передает конфигурацию из плоскости локальных изменений в системную плоскость, применяя конфигурацию на систему. С аргументами commit network происходит применение только сетевых настроек (затрагивает настройки, производимые в командах nameserver, host, timezone, ntp, hostname, route, if).

С аргументом sync команда сохраняет конфигурацию в формате для UI и передает в коллекцию waf.gateways текущей конфигурации MongoDB. После выполнения данной команды настройки, произведенные в wsc, появятся в UI. Все интерфейсы, которые были отмечены командой if mark, появятся в разделе *Шлюзы*.

Внимание! Во вкладке *Шлюзы* после выполнения данной команды шлюз на текущей ноде становится не активным и сбрасываются все привязки алиасов для него.

Применение команды config:

```
# Output full local configuration
config

# Commit local configuration and apply it system-wide
```

```
config commit

# Synchronize configuration with MongoDB (gateways collection)
config sync
```

snapshot

С аргументом restore команда восстанавливает всю конфигурацию, которая была сохранена после последнего применения конфигурации командой config commit. Рекомендуется использовать данную команду взамен команды <section> revert, чтобы избежать ошибок конфигурации.

Применение команды snapshot:

```
# Restore configuration
snapshot restore
```

Внимание! Выполните команду snapshot restore, если после применения команды revert возникли какие-либо ошибки конфигурации. Это позволит восстановить сохраненные настройки.

4.1.3. Информационно-редактирующие команды (DNS, hostname, timezone)

dns

При выполнении команды без аргументов выводится информация о DNS-серверах, используемых для разрешения доменных имен. При выполнении команды с аргументами (IP-адресами DNS-серверов, перечисленными через пробел (до трех серверов)) команда меняет конфигурацию плоскости локальных изменений.

Применение команды dns:

```
# Output current DNS configuration
dns list

# Add DNS server(s)
dns add 8.8.8.8 8.8.4.4

# Delete DNS server
dns del 8.8.4.4
```

hostname

При выполнении команды без аргументов выводится информация о текущем имени узла. При выполнении команды с аргументом, локальным именем, команда меняет конфигурацию плоскости локальных изменений, сохраняя предыдущее имя в истории изменений. Имя узла должно присутствовать в конфигурации секции host плоскости локальных изменений.

Применение команды hostname:

```
# Output current hostname and hostname history
hostname
```

```
# Change hostname ("host list" command output must contain "sasha-corp-ptaf-cl1" host)
hostname sasha-corp-ptaf-cl1
```

timezone

Команда выводит и устанавливает настройки часового пояса.

Применение команды timezone:

```
# Output timezone
timezone

# Set Europe/Moscow timezone
timezone Europe/Moscow
```

4.1.4. Секционные команды (cluster, ntp, host, if, route, user)

В любую из секций можно зайти и написать команды, не предваряя их именем секции, или можно из корневой секции писать полную команду с именем секции.

Пример:

Можно зайти в секцию командой if и написать команды:

```
mode eth0 dhcp
mark eth0
set eth0 inet_address 192.168.0.2 inet_netmask 255.255.255.0

# Следующая команда выходит из секции if в корневую секцию
exit
```

Или написать набор команд:

```
if mode eth0 dhcp
if mark eth0
if set eth0 inet_address 192.168.0.2 inet_netmask 255.255.255.0
```

4.1.4.1. Настройка сетевых интерфейсов (if)

if

Секция if позволяет управлять сетевыми интерфейсами. Без аргументов - команда заходит в секцию настройки интерфейса, а затем позволяет выполнять команды, не предваряя их командой if.

Далее представлены команды, предваренные командой секции.

if list

Команда выводит перечень всех интерфейсов или их свойств.

Применение команды if list:

```
# Output interface configuration
if list
```

```
# Output eth0 configuration
if list eth0
```

if set

Команда устанавливает и удаляет атрибуты, секции.

Применение команды if set:

```
# Set address and netmask of eth0
if set eth0 inet_address 192.168.0.2 inet_netmask 255.255.255.0

# Set gateway to interface eth0
if set eth0 inet_gateway 192.168.0.1

# Set the value to None
if set eth0 inet_gateway none
```

if del

Команда удаляет интерфейсы.

Применение команды if del:

```
# Delete interface bond0
if del bond0
```

if revert

Команда возвращает конфигурацию сети к состоянию, которое было до совершения изменений (в системную плоскость конфигурации).

Применение команды if revert:

```
# Revert changes
if revert
```

if mark / if unmark

Включить/выключить видимость интерфейса из UI. По умолчанию все интерфейсы невидимы (кроме интерфейса с ролью DB, который автоматически определяется по IP-адресу имени узла, содержащегося в секции роли local секции mongo конфигурации cluster).

Применение команды if mark / if unmark:

```
# Mark interface eth0 as visible
if mark eth0

# Unmark interface eth0 as visible
if unmark eth0
```

if mode

Устанавливает вид сетевой настройки интерфейса. Возможные значения: статическая конфигурация, вручную или DHCP.

Применение команды if mode:

```
# Set eth0 interface to use DHCP
if mode eth0 dhcp

# Set bond0 interface for manual start
if mode bond0 manual

# Set eth0 interface back to static configuration
if mode eth0 static
```

if vlan

Создает VLAN'ы от родительского интерфейса. Создаваемые интерфейсы именуются как vlanN, где N – тег.

Применение команды if vlan:

```
# Create VLANs tagged 1060, 1070, 1080 from eth0 interface
if vlan eth0 1060 1070 1080

# Don't forget to set VLAN configuration
if set vlan1060 inet_address 172.16.60.1 inet_netmask 255.255.255.0
if set vlan1070 inet_address 172.16.70.1 inet_netmask 255.255.255.0
if set vlan1080 inet_address 172.16.80.1 inet_netmask 255.255.255.0
```

if alias

Создает алиасы (дополнительные IP-адреса) для интерфейса вида eth0:N, где N – номер алиаса.

Применение команды if alias:

```
# Set aliases 0 and 1 to eth0 interface
if alias eth0 0 1

# Don't forget to set alias configuration
if set eth0:0 inet_addr 192.168.0.3 inet_netmask 255.255.255.0
if set eth0:1 inet_addr 192.168.0.4 inet_netmask 255.255.255.0
```

if bond

Создает бондинг минимум из одного интерфейса. Опции настраиваются с помощью if set.

```
# Create bonding bond0 from vlan1060 and vlan1070 interfaces
if bond 0 vlan1060 vlan1070

# Set bonding options
if set bond0 lacp_rate 0 mode 2
```

if bridge

Создает мост из минимум двух интерфейсов. Опции настраиваются с помощью if set.

Применение команды if bridge:

```
# Create bridge br0 from eth0 and eth1 interfaces
if bridge 0 eth0 eth1

# Set bridge options
if set br0 stp 1 waitport 0 fd 0
```

if span

Данной командой интерфейс включается в режим *promiscuous*, который необходим для работы интерфейса в Span-режиме для работы сниффера (см. главу «Сниффер»).

Применение команды if span:

```
# Set interface to promiscuous mode
if span eth3
```

4.1.4.2. Настройка маршрутизации (route)

route

Секция route позволяет управлять сетевыми маршрутами. Без аргументов - команда заходит в секцию маршрутов, а затем позволяет выполнять команды, не предваряя их командой route.

Далее представлены команды, предваренные командой секции.

route list

Команда выводит перечень всех интерфейсов или их свойств. Маршруты, протокол которых не определен, выводятся в подсекции unknown, все остальные маршруты выводятся согласно протоколу (static или kernel).

Применение команды route list:

```
# Output route configuration
route list

# List routes with "kernel" protocol
route list kernel
```

route add / route del

Команда route add добавляет маршруты в таблицу маршрутизации (подсекция static). Команда route del удаляет маршруты из таблицы.

Применение команды route add / route del:

```
# Add route to 10.0.0.0/8 network via 192.168.163.2
route add 10.0.0.0/8 via 192.168.163.2 dev eth0

# Delete route to 1.1.1.1 from the table
```

```
route del rt11 1.1.1.1
route del main 1.1.1.1

# Delete route to 10.0.0.0/8 from the main table
route del 10.0.0.0/8
```

route revert

Команда возвращает конфигурацию маршрутизации к состоянию, которое было до совершения изменений (в системную плоскость конфигурации).

Применение команды route revert:

```
# Revert changes
route revert
```

route table

Команда управляет таблицами маршрутизации.

Применение команды route table:

```
# Output routing tables
route table list

# Add routing table
route table add mytable 1

# Delete routing table
route table del mytable

# Revert uncommitted routing tables to current system settings
route table revert
```

route rule

Команда управляет правилами маршрутизации. Для правила можно указать приоритет (pref), в ином случае он будет сгенерирован автоматически.

Применение команды route rule:

```
# Output routing rules
route rule list

# Add routing rule
route rule add eth1 from 10.10.0.10/32 pref 444 table mytable

# Delete routing rule
route rule del 444
```

Внимание! Для удаления правила следует указать уникальный номер (pref). Посмотреть pref можно при выполнении команды route rule list.

```
# Revert uncommitted rules to current system settings
route rule revert
```

4.1.4.3. Настройка соответствия имен узлов и IP-адресов (host)

host

Секция host позволяет управлять таблицей соответствия IP-адресов и имен узлов. Без аргументов - команда заходит в секцию host, а затем позволяет выполнять команды, не предваряя их командой route.

Далее представлены команды, предваренные командой секции.

host list

Команда выводит таблицу соответствия IP-адресов и имен узлов.

Применение команды host list:

```
# Output host configuration
host list

# Output hostnames that belong to 127.0.0.1
host list 127.0.0.1
```

host add / host del

Команда host add добавляет имена узлов к IP-адресу и создает адрес при его отсутствии. Команда host del удаляет имена узлов или IP-адреса.

Применение команды host add / host del:

```
# Add "sasha-corp-ptaf-cl1" host to 172.16.60.1
host add 172.16.60.1 sasha-corp-ptaf-cl1
host add 172.16.60.2 sasha-corp-ptaf-cl2

# Delete "debian-minimal" hostname
host del debian-minimal

# Delete IP 127.0.1.1 from hosts and delete all the hostnames associated
with this IP
host del 127.0.1.1
```

host revert

Команда возвращает конфигурацию таблицы соответствия IP-адресов и имен узлов к состоянию, которое было до совершения изменений (в системную плоскость конфигурации).

Применение команды host revert:

```
# Revert changes
host revert
```

4.1.4.4. Настройка пула NTP-серверов (ntp)

ntp

Команда устанавливает пул NTP-серверов.

Применение команды ntp:

```
# List NTP servers
ntp list

# Add NTP server(s)
ntp add 0.pool.ntp.org 1.pool.ntp.org
ntp add 1.1.1.1

# Delete NTP server
ntp del 0.pool.ntp.org
```

ntp revert

Команда возвращает конфигурацию пула NTP-серверов к состоянию, которое было до совершения изменений (в системную плоскость конфигурации).

Применение команды ntp revert:

```
# Revert changes
ntp revert
```

Примечание: настройка часового пояса производится командой timezone.

4.1.4.5. Настройка кластерной конфигурации (cluster)

cluster

Секция cluster позволяет управлять кластерной конфигурацией. Без аргументов - команда заходит в секцию управления кластерной конфигурацией, а затем позволяет выполнять команды, не предваряя их командой cluster.

Далее представлены команды, предваренные командой секции.

cluster list

Команда выводит настройки кластера.

Применение команды cluster list:

```
# Output cluster configuration
cluster list

# Output mongo section
cluster list mongo

# Output elastic section
cluster list elastic
```

cluster set

Команда устанавливает, изменяет и удаляет свойства конфигурации и специальные опции кластера.

Применение команды cluster set:

```
# Set Elastic nodes
cluster set elastic nodes sasha-corp-ptaf-cl1 sasha-corp-ptaf-cl2

# Set MongoDB
cluster set mongo local sasha-corp-ptaf-cl1
cluster set mongo nodes sasha-corp-ptaf-cl2

# Set Elastic and MongoDB options
cluster set elastic replset waf
cluster set mongo replset waf

# Set the value to None
cluster set mongo replset none
cluster set elastic replset none
cluster set mongo nodes none
```

cluster revert

Команда возвращает конфигурацию кластера к состоянию, которое было до совершения изменений (в системную плоскость конфигурации).

Применение команды cluster revert:

```
# Revert changes
cluster revert
```

4.1.4.6. Настройка доступа к REST API (user)

Запросы к REST API могут отправлять пользователи группы Configurators. Пользователь с логином apic создается и добавляется в группу Configurators автоматически, но он нуждается в смене пароля и активации. После изменения данных можно отправлять запросы к REST API.

Применение команды user:

```
# Change user password
user password apic

# Make user active
user activate apic

# Make user inactive
user deactivate apic
```

4.2. Сопоставление запросов к REST API с командами wsc

Функционал REST API необходим для интеграции с Cisco ACI или с любой аналогичной системой. Запросы к REST API, сопоставимые с командами wsc, представлены в Табл. 4. REST API требует базовую авторизацию.

Таблица 4. Сопоставление запросов к REST API с командами wsc

Команда wsc	Запрос к REST API
cluster list	GET /api/wsc/v1/cluster
cluster revert	DELETE /api/wsc/v1/cluster
cluster set <cluster_type> <property> <value>	PATCH /api/wsc/v1/cluster/<cluster_type> {'<property>': '<value>'}
config commit	PUT /api/wsc/v1/config/commit
config sync	PUT /api/wsc/v1/config/sync
list ntp	GET /api/wsc/v1/ntp_servers
timezone	GET /api/wsc/v1/gateway ['timezone']
timezone <value>	PATCH /api/wsc/v1/gateway {'timezone': '<value>'}
hostname	GET /api/wsc/v1/gateway ['hostname']
hostname <value>	PATCH /api/wsc/v1/gateway {'hostname': '<value>'}
dns	GET /api/wsc/v1/nameservers
dns <nameserver1> <nameserverN>	POST /api/wsc/v1/nameservers {'hostname': '<nameserver1>'} POST /api/wsc/v1/nameservers {'hostname': '<nameserverN>'}
host add <ip> <hostname>	POST /api/wsc/v1/hosts {'ip': '<ip>', 'hostname': '<hostname>'}
host del <ip>	DELETE /api/wsc/v1/hosts/<ip>
host list	GET /api/wsc/v1/hosts
host list <ip>	GET /api/wsc/v1/hosts/<ip>
host revert	DELETE /api/wsc/v1/hosts
if bond <number> <interface1> <interfaceN>	POST /api/wsc/v1/interfaces/bonds {'number': '<number>', 'slaves': ['<interface1>', '<interfaceN>']}
if bridge <number> <interface1> <interfaceN>	POST /api/wsc/v1/interfaces/bridges {'number': '<number>', 'ports': ['<interface1>', '<interfaceN>']}
if list	GET /api/wsc/v1/interfaces
if list <interface>	GET /api/wsc/v1/interfaces/<interface>
if mark <interface>	PATCH /api/wsc/v1/interfaces/<interface> {'is_visible': True}
if mode <interface> <mode>	PATCH /api/wsc/v1/interfaces/<interface> {'method': '<mode>'}
if span <interface>	PATCH /api/wsc/v1/interfaces/<interface> {'is_promisc': True}

Таблица 4. Сопоставление запросов к REST API с командами wsc

Команда wsc	Запрос к REST API
if unmark <interface>	PATCH /api/wsc/v1/interfaces/<interface> {'is_visible': False}
if vlan <interface> <number>	POST /api/wsc/v1/interfaces/vlans {'interface': '<interface>', 'number': '<number>'}
route add <dst> <param1> <value1> <paramN> <valueN>	POST /api/wsc/v1/routes {'dst': '<dst>', '<param1>': '<value1>', '<paramN>': '<valueN>'}
route del <dst>	DELETE /api/wsc/v1/routes/<dst>
route list	GET /api/wsc/v1/routes
route list <dst>	GET /api/wsc/v1/routes/<dst>
route revert	DELETE /api/wsc/v1/routes

4.2.1. Примеры запросов к REST API

В приведенных ниже примерах используется HTTPie (https://github.com/jkbrzt/httpie) - CLI HTTP клиент. Вместо имени пользователя и пароля в примерах используется login:password. Информация по конфигурированию и активированию пользователя представлена в главе «Настройка доступа к REST API (user)».

4.2.1.1. WSC API

Команды данной группы примеров аналогичны командам wsc.

Hosts

```
# Get all hosts
http -a login:password --verify=no get https://node:8443/api/wsc/v1/
hosts


# Add host
http -a login:password --verify=no post https://node:8443/api/wsc/v1/
hosts ip=123.123.123.123 hostnames:='["myhostname"]'


# Delete host
http -a login:password --verify=no delete https://node:8443/api/wsc/
v1/hosts/123.123.123.123


# Cancel uncommitted changes
http -a login:password --verify=no put https://node:8443/api/wsc/v1/
hosts/revert
```

Hostname

```
# Get hostname
http -a login:password --verify=no get https://node:8443/api/wsc/v1/
gateway


# Change hostname
http -a login:password --verify=no patch https://node:8443/api/wsc/v1/
gateway hostname=myhostname
```

Nameservers

```
# Get all nameservers
http -a login:password --verify=no get https://node:8443/api/wsc/v1/
nameservers

# Add nameserver
http -a login:password --verify=no post https://node:8443/api/wsc/v1/
nameservers ip=8.8.8.8

# Delete nameserver
http -a login:password --verify=no delete https://node:8443/api/wsc/
v1/nameservers/8.8.8.8

# Delete all nameservers
http -a login:password --verify=no delete https://node:8443/api/wsc/
v1/nameservers

# Cancel uncommitted changes
http -a login:password --verify=no put https://node:8443/api/wsc/v1/
nameservers/revert
```

Timezone

```
# Get timezone
http -a login:password --verify=no get https://node:8443/api/wsc/v1/
gateway

# Change timezone
http -a login:password --verify=no patch https://node:8443/api/wsc/v1/
gateway timezone='Europe/Moscow'
```

NTP servers

```
# Get all NTP servers
http -a login:password --verify=no get https://node:8443/api/wsc/v1/
ntp_servers

# Add NTP server
http -a login:password --verify=no post https://node:8443/api/wsc/v1/
ntp_servers hostname=0.pool.ntp.org

# Delete NTP server
http -a login:password --verify=no delete https://node:8443/api/wsc/
v1/ntp_servers/0.pool.ntp.org

# Delete all NTP servers
http -a login:password --verify=no delete https://node:8443/api/wsc/
v1/ntp_servers

# Cancel uncommitted changes
http -a login:password --verify=no put https://node:8443/api/wsc/v1/
ntp_servers/revert
```

Interfaces

```
# Get all interfaces
http -a login:password --verify=no get https://node:8443/api/wsc/v1/
interfaces

# Make interface visible for UI
http -a login:password --verify=no patch https://node:8443/api/wsc/v1/
interfaces/eth0 is_visible:=true

# Make interface invisible for UI
http -a login:password --verify=no patch https://node:8443/api/wsc/v1/
interfaces/eth0 is_visible:=false

# Add alias
http -a login:password --verify=no post https://node:8443/api/wsc/v1/
interfaces/aliases parent=eth0 number=1

# Add vlan
http -a login:password --verify=no post https://node:8443/api/wsc/v1/
interfaces/vlans parent=eth0 tag=1

# Add bond
http -a login:password --verify=no post https://node:8443/api/wsc/v1/
interfaces/bonds number=1 slaves:='["eth1", "eth2"]'

# Add bridge
http -a login:password --verify=no post https://node:8443/api/wsc/v1/
interfaces/bridges number=1 ports:='["eth1", "eth2"]'

# Delete interface
http -a login:password --verify=no delete https://node:8443/api/wsc/
v1/interfaces/br1

# Delete all non-physical interfaces
http -a login:password --verify=no delete https://node:8443/api/wsc/
v1/interfaces

# Change interface properties
http -a login:password --verify=no patch https://node:8443/api/wsc/v1/
interfaces/eth0 inet_address=192.168.0.2 inet_netmask=255.255.255.0

# Cancel uncommitted changes
http -a login:password --verify=no put https://node:8443/api/wsc/v1/
interfaces/revert
```

Routes

```
# Get all routes
http -a login:password --verify=no get https://node:8443/api/wsc/v1/
routes
```

```
# Get static route
http -a login:password --verify=no get https://node:8443/api/wsc/v1/
routes/default

# Add route
http -a login:password --verify=no post https://node:8443/api/wsc/v1/
routes destination=default via=192.168.163.2 dev=eth0

# Delete static route
http -a login:password --verify=no delete https://node:8443/api/wsc/
v1/routes/default

# Delete all static routes
http -a login:password --verify=no delete https://node:8443/api/wsc/
v1/routes

# Cancel uncommitted changes
http -a login:password --verify=no put https://node:8443/api/wsc/v1/
routes/revert
```

Mongo

```
# Get mongo configuration
http -a login:password --verify=no put https://node:8443/api/wsc/v1/
cluster/mongo

# Change mongo configuration
http -a login:password --verify=no patch https://node:8443/api/wsc/v1/
cluster/mongo local:='["node1"]' nodes:='["node2"]' replset=waf
```

Elastic

```
# Get elastic configuration
http -a login:password --verify=no put https://node:8443/api/wsc/v1/
cluster/elastic

# Change elastic configuration
http -a login:password --verify=no patch https://node:8443/api/wsc/v1/
cluster/elastic nodes:='["node1", "node2"]' replset=waf
```

Config

```
# Commit changes
http -a login:password --verify=no put https://node:8443/api/wsc/v1/
config/commit

# Sync changes
http -a login:password --verify=no put https://node:8443/api/wsc/v1/
config/sync
```

4.2.1.2. WAF API

Команды данной группы примеров используются для конфигурирования в UI.

Interface aliases

```
# Get all interface aliases
http -a login:password --verify=no get https://node:8443/api/waf/v1/
interface_aliases

# Get interface alias
http -a login:password --verify=no get https://node:8443/api/waf/v1/
interface_aliases/mgmt

# Add interface aliases
http -a login:password --verify=no post https://node:8443/api/waf/v1/
interface_aliases name=mgmt type=MGMT
http -a login:password --verify=no post https://node:8443/api/waf/v1/
interface_aliases name=wan type=WAN
http -a login:password --verify=no post https://node:8443/api/waf/v1/
interface_aliases name=lan type=LAN

# Replace or add interface aliases
http -a login:password --verify=no put https://node:8443/api/waf/v1/
interface_aliases/mgmt type=MGMT
http -a login:password --verify=no put https://node:8443/api/waf/v1/
interface_aliases/wan type=WAN
http -a login:password --verify=no put https://node:8443/api/waf/v1/
interface_aliases/lan type=LAN

# Delete interface alias
http -a login:password --verify=no delete https://node:8443/api/waf/
v1/interface_aliases/lan
```

Gateways

```
# Get all gateways
http -a login:password --verify=no get https://node:8443/api/waf/v1/
gateways

# Get gateway
http -a login:password --verify=no get https://node:8443/api/waf/v1/
gateways/node

# Assign interface alias
http -a login:password --verify=no patch https://node:8443/api/waf/v1/
gateways/localhost/interfaces/eth0 aliases:='["mgmt"]'
```

Upstreams

```
# Get all upstreams
http -a login:password --verify=no get https://node:8443/api/waf/v1/
upstreams

# Get upstream
```

```
http -a login:password --verify=no get https://node:8443/api/waf/v1/
upstreams/something


# Add upstream
http -a login:password --verify=no post https://node:8443/api/waf/v1/
upstreams name=something backends:='[{"address": "123.123.123.123"}]


# Add or replace upstream
http -a login:password --verify=no put https://node:8443/api/waf/v1/
upstreams/something backends:='[{"address": "123.123.123.123"}]


# Delete upstream
http -a login:password --verify=no delete https://node:8443/api/waf/
v1/upstreams/something
```

Profiles

```
# Get all profiles
http -a login:password --verify=no get https://node:8443/api/waf/v1/
profiles


# Get profile
http -a login:password --verify=no get https://node:8443/api/waf/v1/
profiles/myprofile


# Add profile
http -a login:password --verify=no post https://node:8443/api/waf/v1/
profiles name=myprofile


# Configure profile
http -a login:password --verify=no patch https://node:8443/api/waf/v1/
profiles/myprofile proxy:='{"servers": [{"listen_ip":
"myinterfacealias1", "upstream": "myupstream"}, {"listen_ip":
"myinterfacealias2", "upstream": "myupstream"}]}'


# Add or replace profile
http -a login:password --verify=no put https://node:8443/api/waf/v1/
profiles/myprofile proxy:='{"servers": [{"listen_ip":
"myinterfacealias", "upstream": "myupstream"}]}'


# Delete profile
http -a login:password --verify=no delete https://node:8443/api/waf/
v1/profiles/myprofile
```

Firewall

```
# List IPs for Arbor
http -a login:password --verify=no get https://node:8443/api/waf/v1/
firewall/arbor.txt
```

5. Развертывание системы

PT AF может работать с сетевым трафиком в нескольких режимах: в режиме мониторинга (PT AF принимает трафик со SPAN-порта коммутатора и регистрирует ошибки) и в режиме блокирования атак (обратный прокси-сервер). Кроме того, PT AF может осуществлять оффлайн-анализ журналов веб-сервера, сетевых дампов на предмет обнаружения в них следов атак, т.е. режим Forensic.

5.1. Режим мониторинга

5.1.1. Настройка PT AF

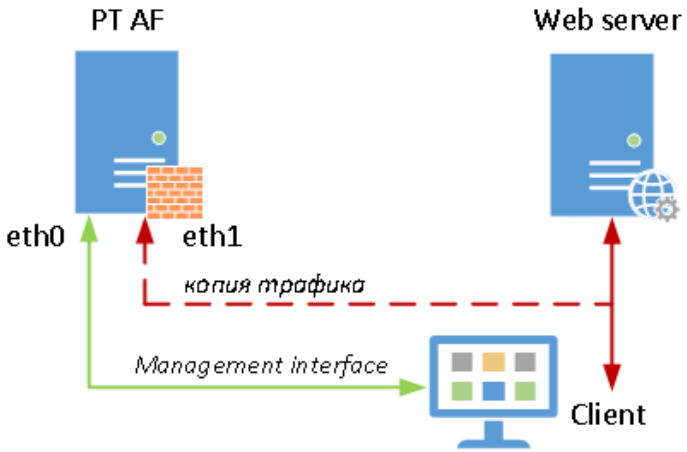


Рис. 8 – Схема работы в режиме мониторинга

1. Подключиться к PT AF браузером: http://172.16.9.9/.
2. Авторизоваться под стандартной учетной записью: admin / p0s1t1v3.

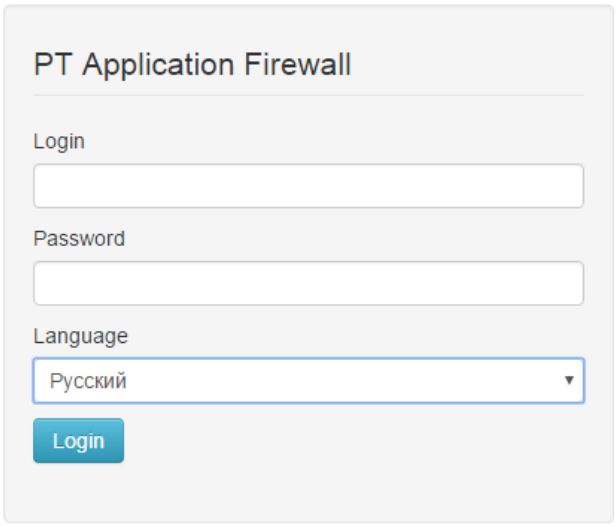


Рис. 9 – Окно авторизации

Примечание: рекомендуется сменить стандартный пароль при первом входе в систему.

3. Перейти на вкладку *Конфигурация -> Политики безопасности -> Профили*. Войти в режим редактирования профиля *Default* и отметить опцию *Сниффер подключен* во вкладке *Основные*, сохранить настройки. Убедиться, что процесс запущен.

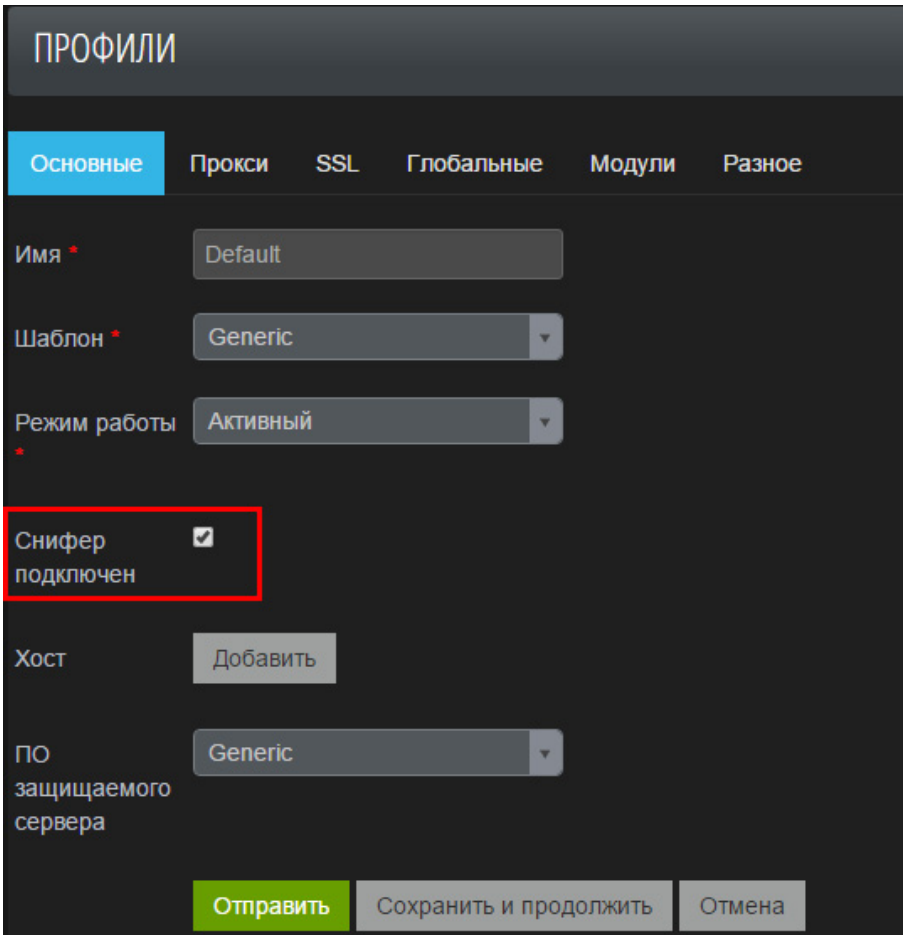


Рис. 10 – Настройки профиля

4. Подключиться браузером к Web-серверу http://172.16.9.20/.

5. Выполнить какой-нибудь подозрительный запрос, например http://172.16.9.20/sysuser/docmgr/search.stm?query=<script>alert(document.cookie)</script>

Примечание: для генерации подобных запросов можно использовать какой-нибудь сканер уязвимостей, например, NIKTO, sqlmap, w3f и т.д.

6. Убедиться, что PT AF среагировал на подозрительный запрос:

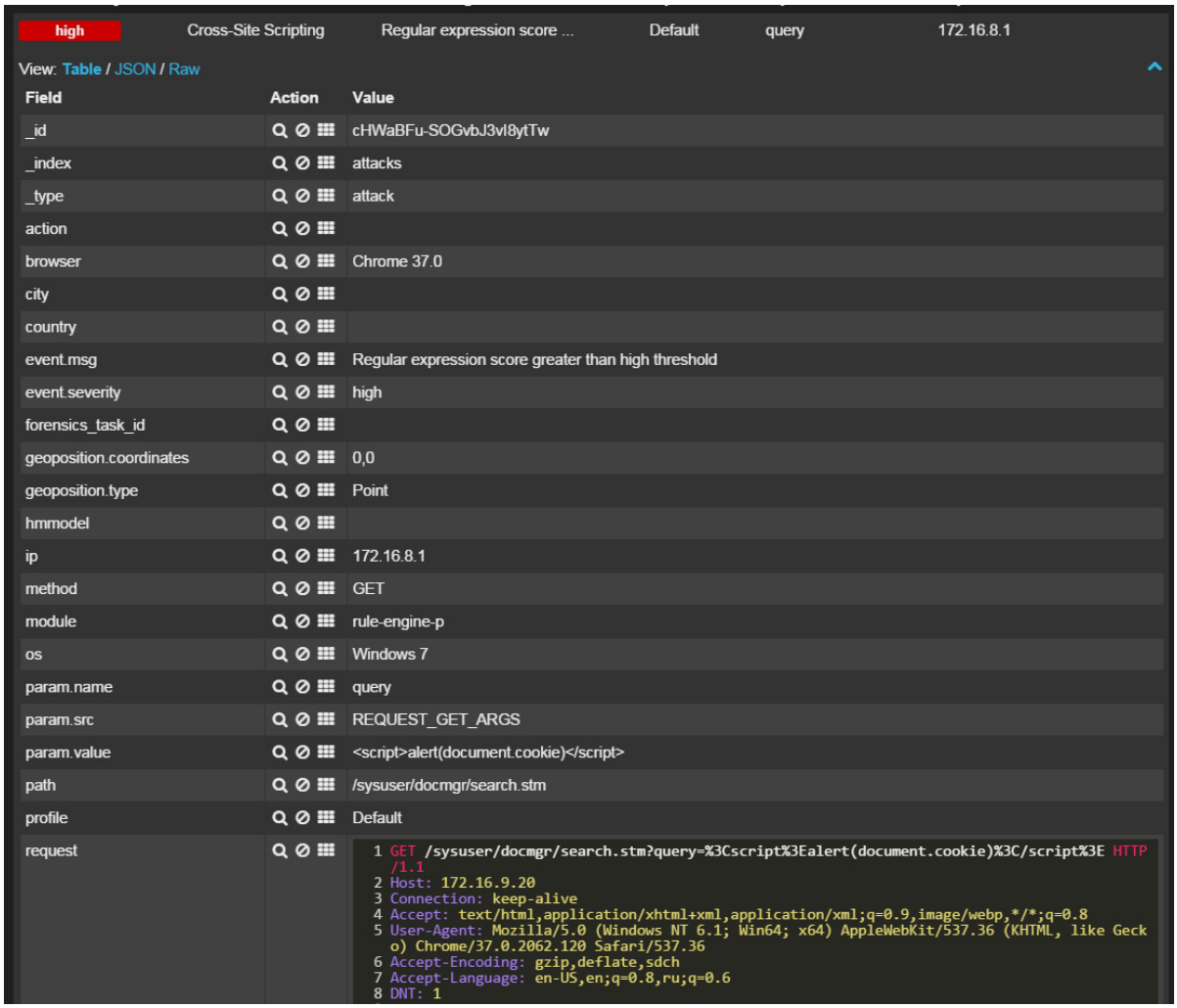


Рис. 11 – Проверка запроса

5.1.2. Настройка расшифровки SSL-трафика

SSL-трафик приложения может быть расшифрован, если загрузить приватный ключ сертификата на PT AF. Для включения этой опции понадобится выполнить следующие шаги:

1. Перейти на вкладку *Конфигурация -> SSL сертификаты и ключи*;

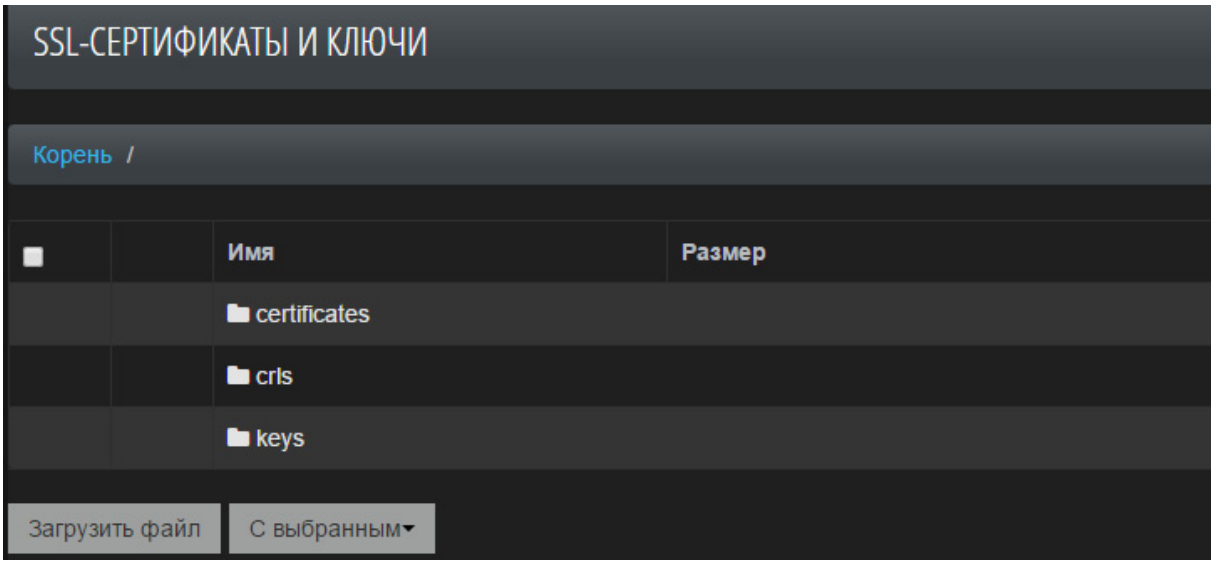


Рис. 12 – SSL-сертификаты и ключи

2. Загрузить в каталог keys файл приватного ключа;
3. Перейти на вкладку *Конфигурация –> Сеть -> Сниффер*;
4. Нажать кнопку *Добавить*, чтобы добавить новый сервер;

Рис. 13 – Добавление сервера

5. Указать настройки сервера приложения:
   - *Имя* – имя приложения myapp.ru. Если на одном IP-адресе находятся несколько приложений с разными ключами, можно дифференцировать их по именам приложений;
   - IP-адрес – IP-адрес приложения;
   - *Порт* – SSL Listen Port - 433;
   - *Приватный SSL-ключ* – указать приватный ключ, загруженный на шаге 2;
   - *Пароль для приватного SSL-ключа* – указать пароль, если такой был задан при создании приватного ключа.

Примечание: расшифровка сниффером SSL-трафика возможна только в то случае, когда на стороне защищаемого приложения в качестве протокола обмена ключами используются следующие наборы шифров:
TLS_RSA_WITH_RC4_128_MD5,
TLS_RSA_WITH_RC4_128_SHA,
TLS_RSA_WITH_AES_128_CBC_SHA,
TLS_RSA_WITH_AES_128_CBC_SHA256,
TLS_RSA_WITH_AES_128_GCM_SHA256,
TLS_RSA_WITH_3DES_EDE_CBC_SHA,
TLS_RSA_WITH_AES_256_CBC_SHA,
TLS_RSA_WITH_AES_256_CBC_SHA256,
TLS_RSA_WITH_AES_256_GCM_SHA384
Наименование указанных шифров соответствует классификации openssl: https://www.openssl.org/docs/manmaster/apps/ciphers.html

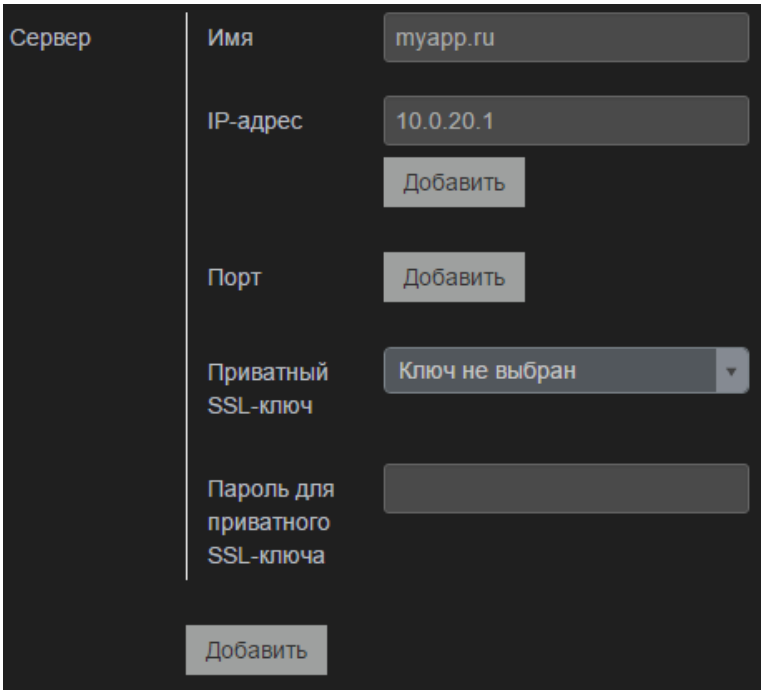


Рис. 14 – Настройки сервера

6. Сохранить настройки сниффера.

5.2. Режим блокирования атак (обратный прокси)

5.2.1. Настройка PT AF

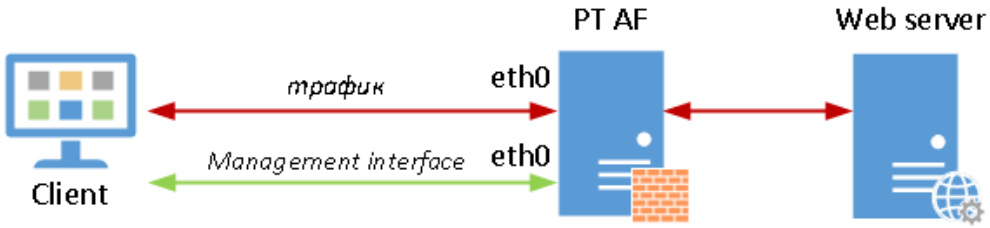


Рис. 15 – Схема работы в режиме блокировки атак

1. Подключиться к PT AF браузером: http://172.16.9.9/;
2. Перейти на вкладку *Конфигурация -> Сеть -> Группа серверов*;
3. Создать новую группу;

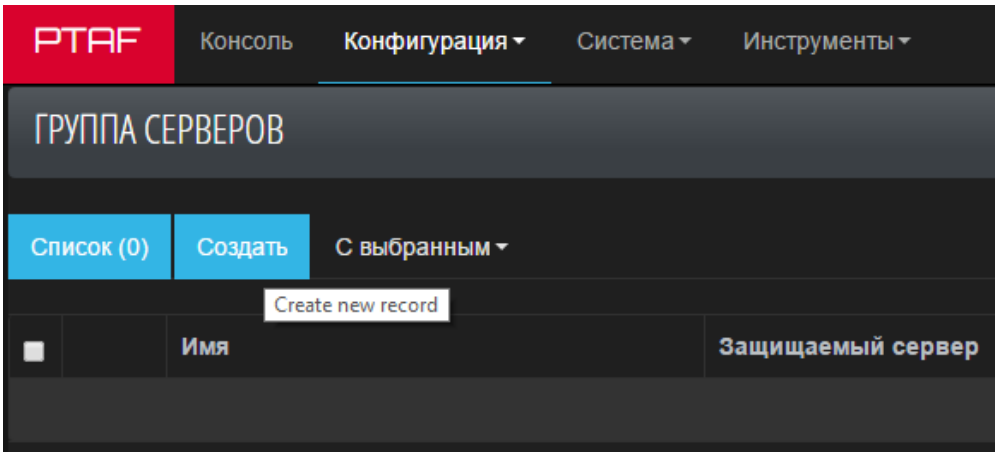


Рис. 16 – Создание новой группы серверов

4. В открывшейся вкладке заполнить следующие поля:

   • *Имя* – DNS-имя для сайтов, на которые будет проходить переадресация. Например, «backend»;

   • *Защищаемый сервер, IP-адрес* – IP-адрес сервера с веб-приложением, например «172.16.9.20».

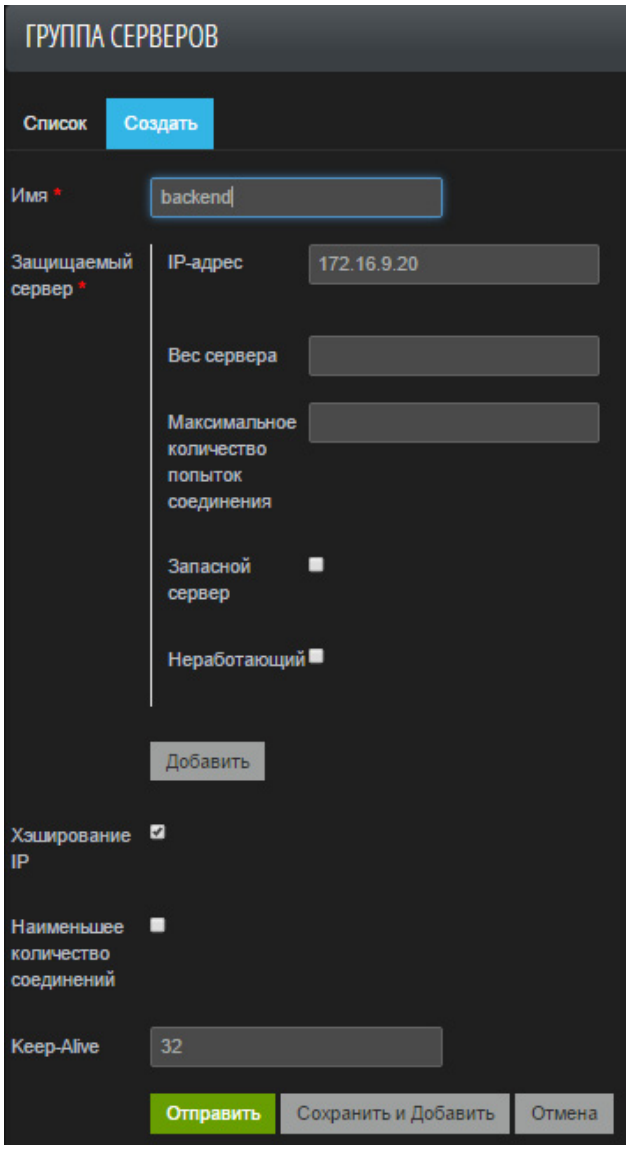


Рис. 17 – Настройки группы серверов

5. Сохранить изменения, нажав *Отправить*;
6. Перейти на вкладку *Конфигурация -> Политики безопасности -> Профили*;
7. Отредактировать стандартный профиль *Default*;

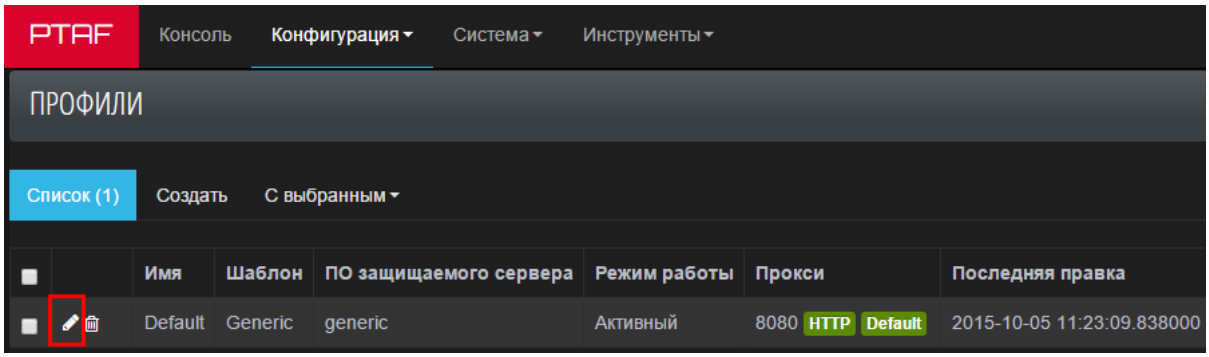


Рис. 18 – Редактирование профиля

8. На вкладке *Прокси* включить режим обратный-прокси, выбрав опцию *Сервер по умолчанию*:

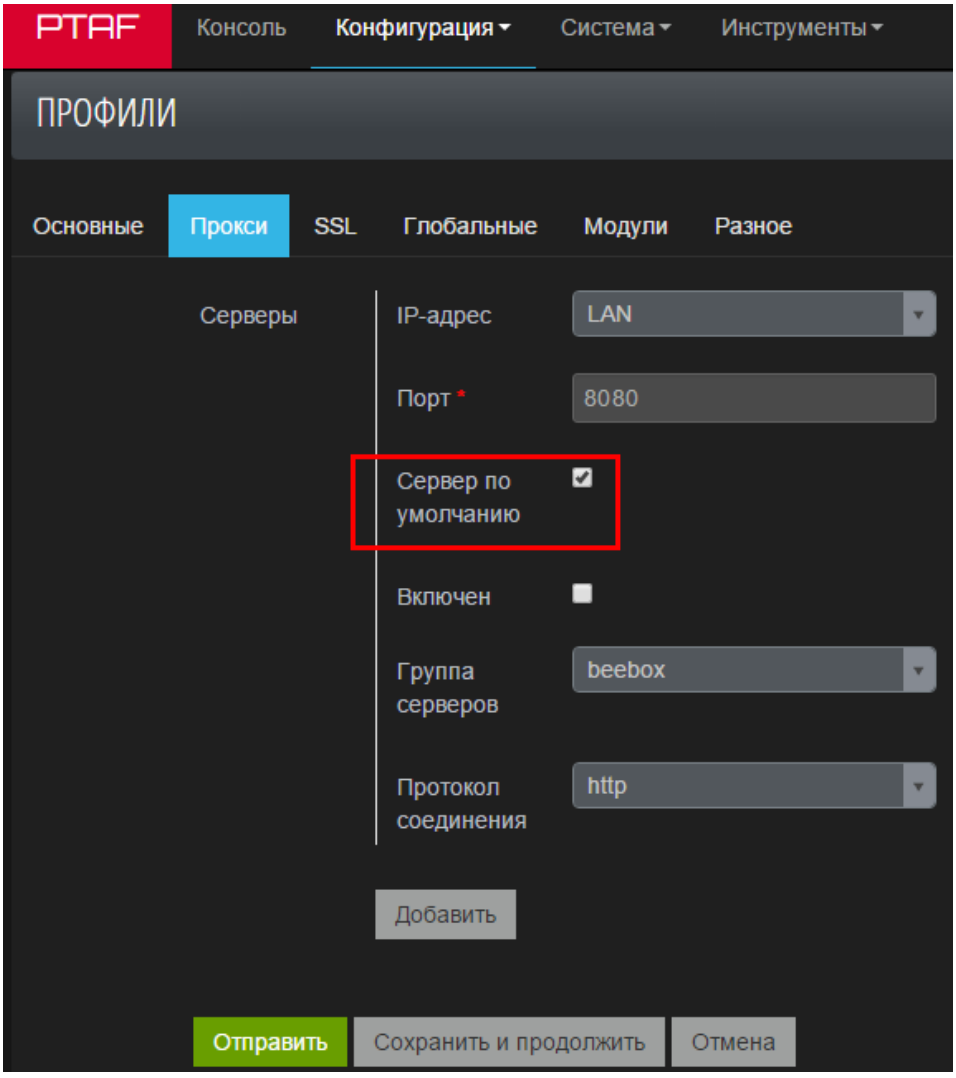


Рис. 19 – Настройки Прокси

9. Сохранить изменения, нажав *Отправить*;

10. Проверить, что Web-сервер теперь открывается по адресу: http://172.16.9.9:8080/;
11. Выполнить подозрительный запрос, например: http://172.16.9.9:8080/rpc.php?q=\%22%3e%3cscript%3ealert(document.cookie)%3c/script>;
12. Убедиться, что PT AF заблокировал запрос и отобразил результат в консоли:

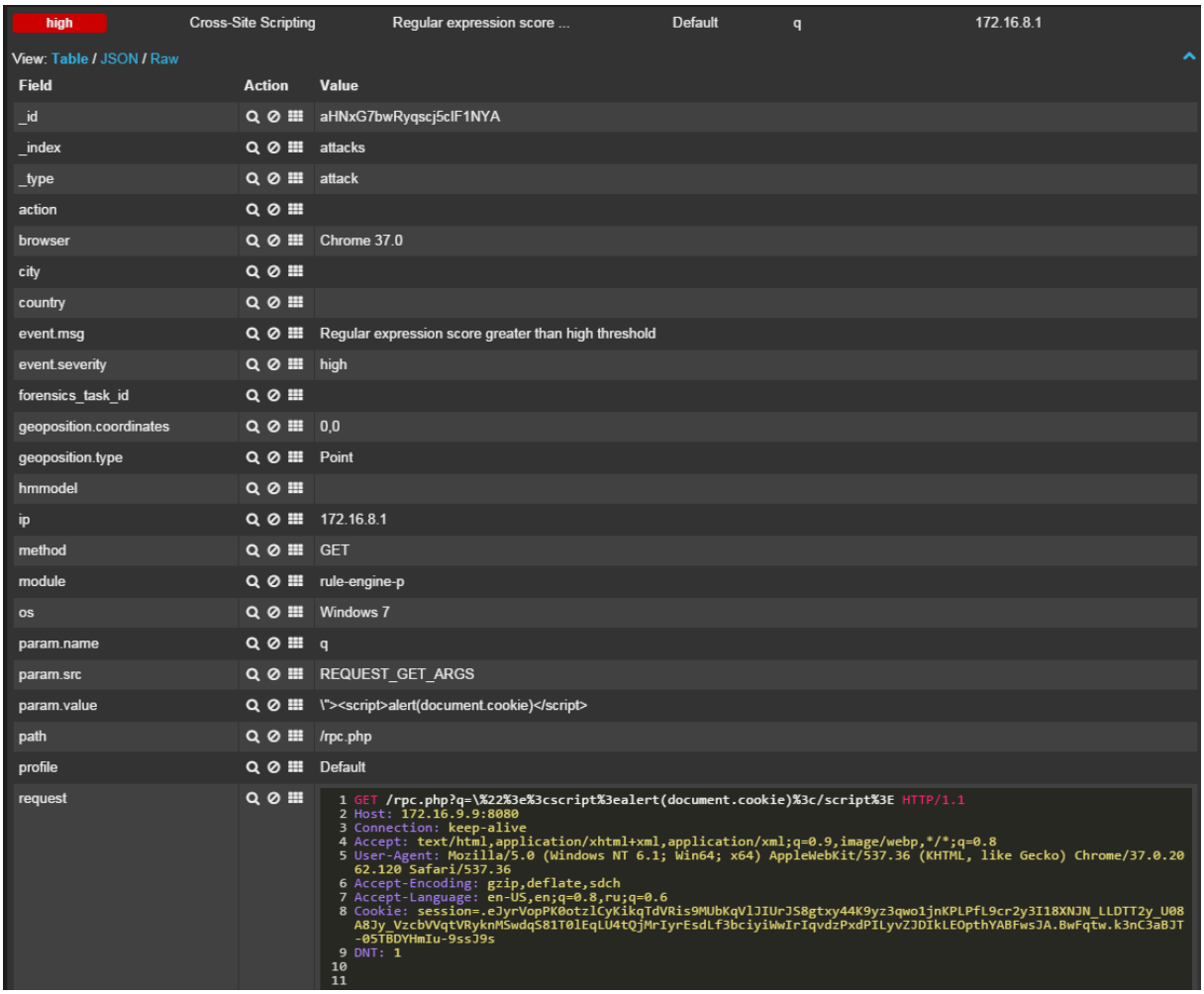


Рис. 20 – Блокировка запроса

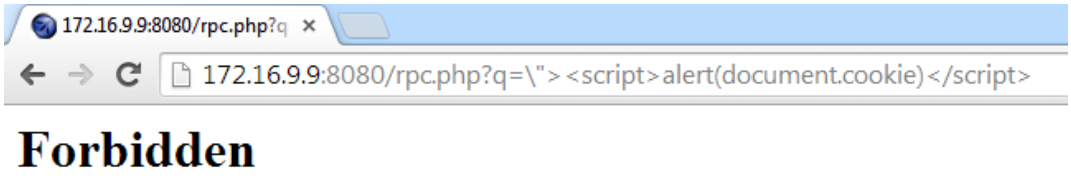


Рис. 21 – Результат блокировки запроса

5.2.2. Настройка расшифровки SSL-трафика

Для настройки PT AF предполагается наличие сертификата и приватного ключа. На официальном сайте openssl можно найти документацию по сертификатам, ключам и сертификатам прокси:

https://www.openssl.org/docs/HOWTO/certificates.txt

https://www.openssl.org/docs/HOWTO/keys.txt

https://www.openssl.org/docs/HOWTO/proxy_certificates.txt

Структура сертификатов x509 описана в RFC-5280 (http://tools.ietf.org/html/rfc5280).

Например, для выпуска самоподписанного сертификата и приватного ключа можно выполнить следующую команду:

```
# openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout
privateKey.key -out certificate.crt
Country Name (2 letter code) [AU]:RU
State or Province Name (full name) [Some-State]:Moscow
Locality Name (eg, city) []:Moscow
Organization Name (eg, company) [Internet Widgits Pty Ltd]:Demo
Organizational Unit Name (eg, section) []:MyApp
Common Name (e.g. server FQDN or YOUR name) []:myapp.ru
Email Address []:admin@myapp.ru
```

Для расшифровки SSL-трафика в PT AF необходимо выполнить следующие шаги:

1. Перейти на вкладку *Конфигурация -> SSL-сертификаты и ключи*;
2. Загрузить в каталог certificates – файл сертификата, а в каталог keys – файл приватного ключа;
3. Перейти на вкладку *Конфигурация -> Политики безопасности -> Профили*;
4. Отредактировать профиль, для которого будет выполняться расшифровка SSL-трафика (например, Default);

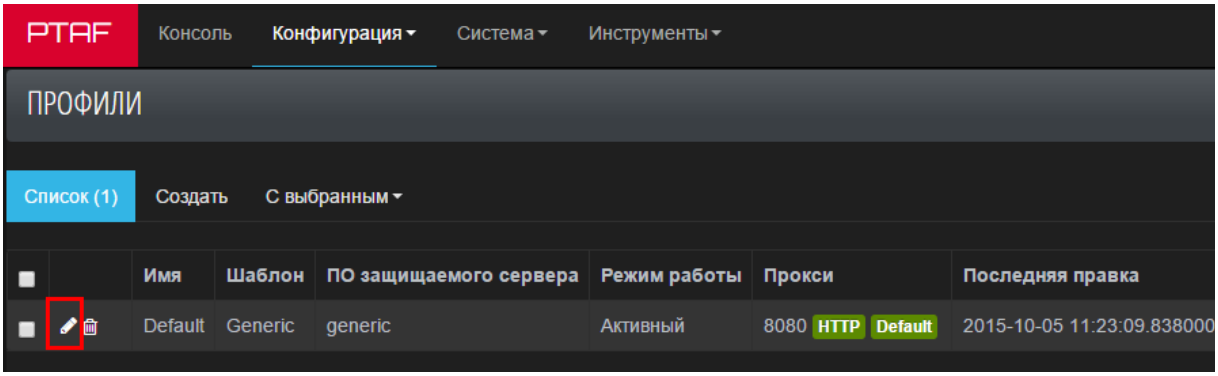


Рис. 22 – Редактирование профиля

5. Открыть вкладку SSL и заполнить необходимые поля (см. Рис. 23):
   - *SSL-сертификат* – указать сертификат, загруженный на шаге 2;
   - *Приватный SSL-ключ* – указать файл ключа, загруженный на шаге 2;

• *SSL-шифры* – список алгоритмов, поддерживаемых библиотекой openssl для соединений. Этот список задается на моменте компиляции, и по умолчанию для openssl версии 1.0.0 соответствует: ALL:!aNULL:!eNULL

• *SSL-протоколы* – выбор протоколов SSL из списка: SSLv3, TLSv1, TLSv1.1, TLSv1.2. Для примера укажем все возможные варианты.

6. Сохранить профиль.

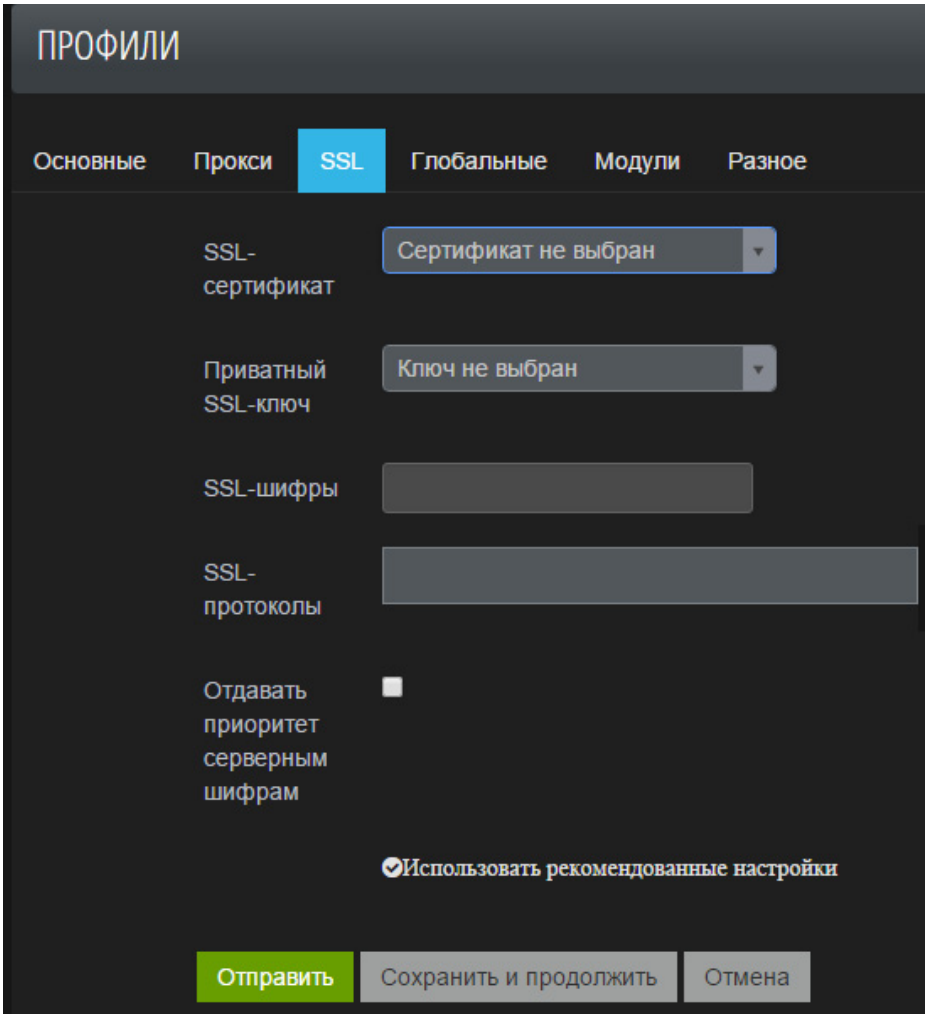


Рис. 23 – Настройки SSL

5.3. Отказоустойчивая конфигурация

Если для защиты приложения PT AF будет использоваться в качестве межсетевого экрана (режим реверс-прокси) и выйдет из строя, то приложение будет недоступно, что может привести к репутационным и финансовым потерям.

Избежать таких ситуаций можно, обеспечив высокую доступность приложения (high availability). Существуют несколько путей решения, например, построить кластер из нескольких PT AF. В этом случае при выходе из строя одного из PT AF, пользовательский трафик будет переключен балансировщиком нагрузки на резервный. Информация по

построению кластера подробно представлена в документе «Построение отказоустойчивого решения на базе Positive Technologies Application Firewall».

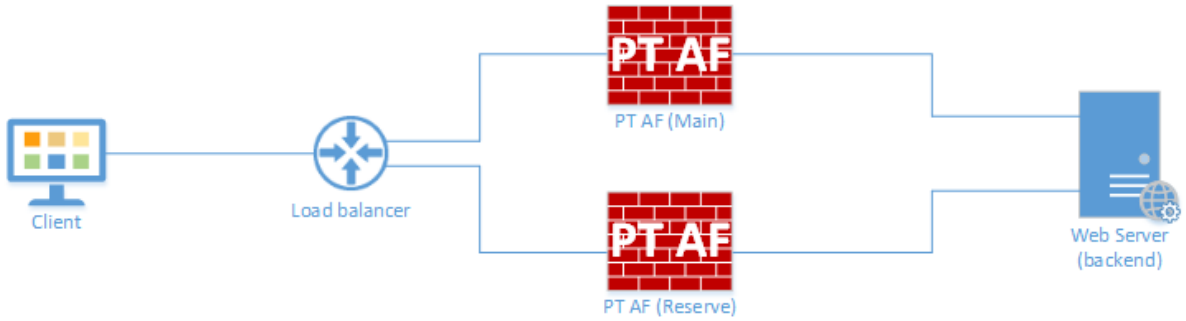


Рис. 24 – Схема работы с балансировщиком нагрузки

Аналогично можно обеспечить резервирование канала, когда при отказе PT AF трафик будет направляться напрямую к приложению.

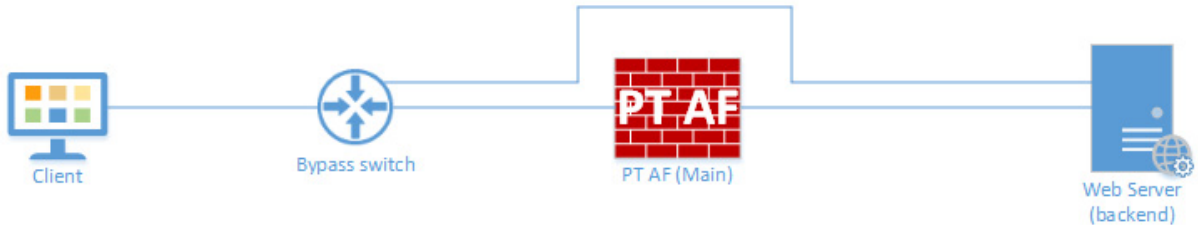


Рис. 25 – Схема с резервированием канала

Этот механизм реализуется двумя следующими способами:

- bypass-switch – внешние устройство подключается к серверу с PT AF;
- bypass network adapter – сетевая карта устанавливается в сервер с PT AF.

Более подробная информацию по механизму bypass представлена в документе «Построение отказоустойчивого решения на базе Positive Technologies Application Firewall».

5.3.1. Построение кластера

При объединении нескольких PTAF в кластер выделяется мастер-нода, которая будет ответственна за синхронизацию БД. Но это не значит, что данная нода будет является единственной активной в кластере, так как распределение трафика по кластеру управляется “VRRP” группами, которые могут быть гибко растянуты. Так, например, в кластере из двух нод, может быть две “VRRP” группы, для каждой из которых мастер-нода является разной. Таким способом можно организовать, как Active-Active, так и Active-Passive кластер. При попадании ноды в кластер, она сохраняет свою возможность доступности через административный интерфейс, при этом все события безопасности и конфигурации синхронизируются.

Создание кластера происходит в два этапа:

- Создание Арбитра на мастер-ноде, синхронизация БД Mongo на нодах;
- Синхронизация Elasticsearch.

Внимание! Перед началом конфигурирования, важно убедится в правильности записей доменных имен в “/etc/”.

6. Интерфейс и работа с системой

При входе в систему пользователь попадает на главную страницу.

В системе пользователю доступны следующие разделы (Рис. 26):

- главная страница (см. главу «Консоль»);
- конфигурация (см. главу «Конфигурация»);
- система (см. главу «Система»);
- инструменты (см. главу «Инструменты»).

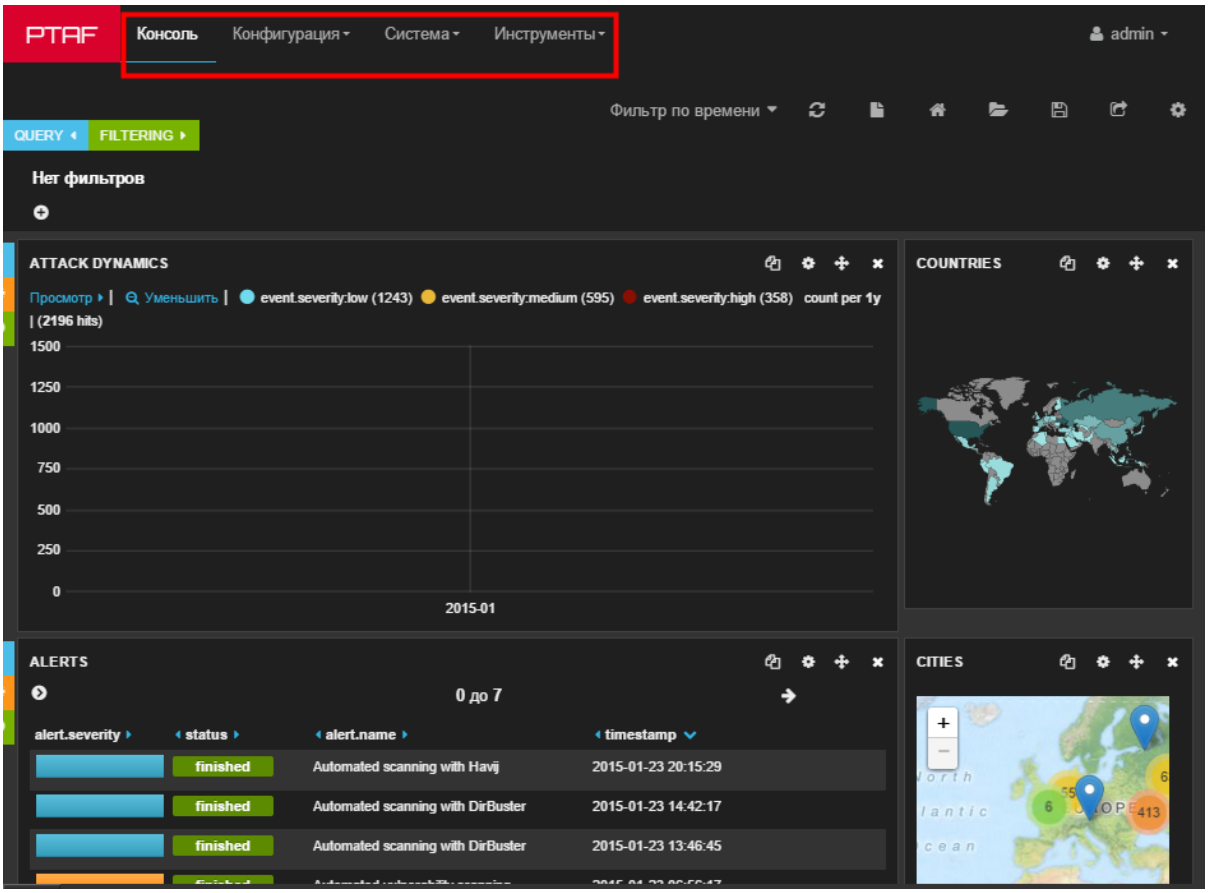


Рис. 26 – Разделы системы

6.1. Объектная модель атак в PT AF

Для того, чтобы лучше понять интерфейсные объекты, описанные в данном разделе, рассмотрим объектную модель, используемую в PT AF для событий.

Когда PT AF регистрирует некоторое событие, связанное с отправкой одного запроса от клиента к серверу, может оказаться, что ему соответствует несколько разных типов атак, которые обнаруживаются разными способами. Например, при сканировании автоматическим сканером, который размещает свою сигнатуру в HTTP-заголовке User-Agent, в PT AF на один запрос может сработать несколько правил:

1. будет зарегистрирован факт обращения от автоматического сканера;
2. будет зарегистрирован факт атаки XSS (или другой атаки, которую будет проводить этот сканер).

В свою очередь комплексная атака может состоять последовательно из нескольких запросов. Так BruteForce атака состоит из нескольких последовательных неудачных попыток входа за определенный промежуток времени, а SQL-инъекция может состоять из нескольких последовательных действий:

1. SQL injection probing – обнаружение уязвимости;
2. SQL Injection column number guessing – определения количества столбцов в таблице;
3. SQL Injection data extraction – инъекция.

Итак, каждому HTTP-запросу, отображаемому в консоли PT AF, ставится в соответствие один или более тегов, поясняющий суть произошедшего. Эти HTTP-запросы отображаются на вкладке *Attacks* в консоли. Назовем такие HTTP-запросы событиями. Несколько событий могут быть объединены при помощи правил корреляции в скоррелированное событие – событие, означающее целую атаку злоумышленника, состоявшую из нескольких последовательных действий. Скоррелированные события отображаются на панели *Alerts*.

Правила корреляции, на основе которых несколько отдельных событий могут быть скоррелированы, могут состоять из одного простого события, часто повторявшегося в течение некоторого промежутка времени, или последовательности из нескольких событий разного типа. Например, правило корреляции Verified "UNION" SQL injection Exploitation, предназначенное для обнаружения SQL-инъекции, будет состоять из трех последовательных потенциальных событий: SQL injection probing, SQL Injection column number guessing, SQL Injection data extraction.

Подробнее создание потенциальных событий и правил корреляции на их основе рассмотрено ниже.

6.2. Консоль

Основной экран предназначен для исследования трафика. На данной странице представлен график атак, что позволяет изучать события, происходящие в сети. Карта предназначена для исследования географии атак.

Консоль может работать в двух режимах – базовом и расширенном. Базовый режим дает возможность только настраивать фильтры, расширенный режим предоставляет широкие возможности по настройке представлений и выводимых данных. Более подробную информацию см. в главе «Общие настройки консоли».

Выбор данных из базы событий, попадающих в *Консоль* (Рис. 27), осуществляется при помощи одного или нескольких запросов, которые вводятся на панели *Query* (см. главу «Запросы (панель Query)»). Для фильтрации данных предназначены инструменты на панели *Filtering* (см. главу «Фильтры (панель Filtering)»).

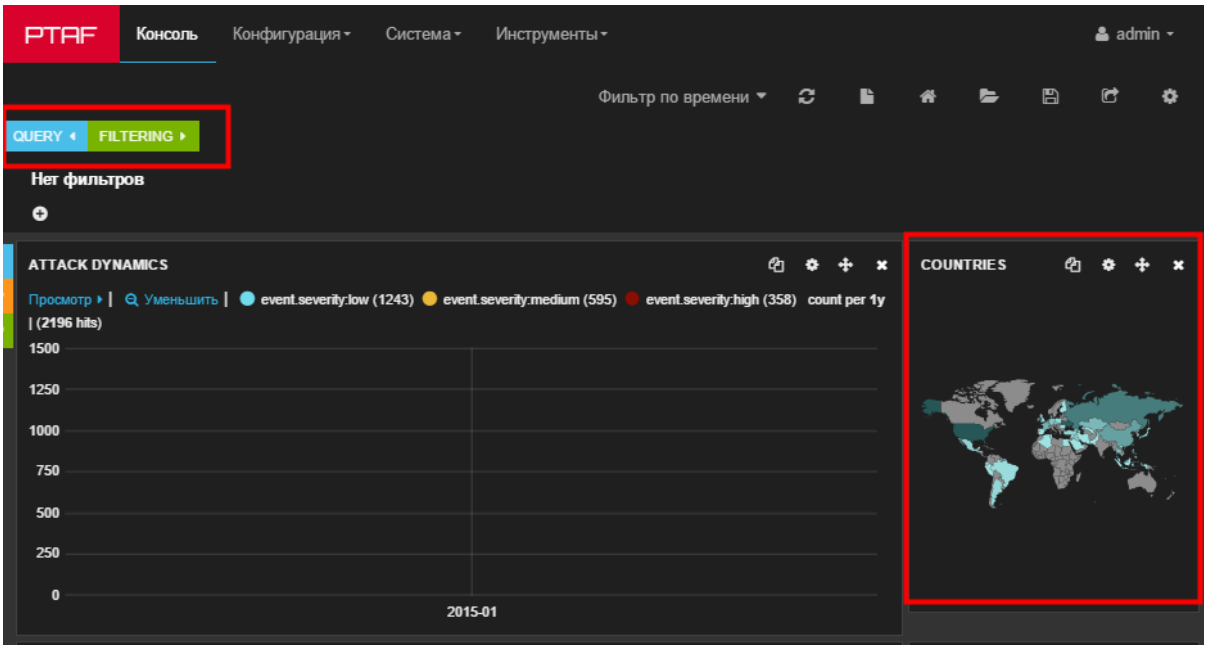


Рис. 27 – Консоль

6.2.1. Панель управления

Панель управления раздела *Консоль* - это удобное средство для пользователей, которое позволяет быстро выполнить часто используемые команды.

Рис. 28 – Панель управления

Панель управления находится в верхней части страницы и содержит команды, указанные в Табл. 5.

Таблица 5. Команды панели управления

Кнопка	Команда	Описание	Примечание
Фильтр по времени	Установить фильтр по времени	Установленный временной фильтр	См. главу «Панель Attack Dynamics»
	Обновить	Принудительное обновление данных расположенных на информационных панелях	
	Отчеты	Открытие окна генерации отчетов	См. главу «Генератор отчетов»
	Домашняя страница	Сброс состояния панелей к домашнему варианту	Кнопка доступна только в расширенном режиме работы
	Загрузить	Загрузка одного из сохраненных состояний	Кнопка доступна только в расширенном режиме работы

Таблица 5. Команды панели управления

Кнопка	Команда	Описание	Примечание
	Сохранить	Сохранение состояния панелей или установка домашнего состояния через дополнительное выпадающее меню Сохранить как консоль по умолчанию/Сбросить консоль по умолчанию/ Экспортировать конфигурацию консоли	Кнопка доступна только в расширенном режиме работы
	Поделиться	Отправка состояния панелей в виде ссылки	Кнопка доступна только в расширенном режиме работы
	Настроить консоль	Конфигурация внешнего вида консоли	См. главу «Общие настройки консоли»
	Мой аккаунт/ Выход	Изменение данных о своем почтовом адресе, пароля/ Выход из системы.	

В правом верхнем углу каждой панели консоли в расширенном режиме работы размещены элементы управления.

Таблица 6. Команды панелей консоли

Кнопка	Команда	Описание	Примечание
	Дублировать	Сделать копию панели	
	Настроить	Настройка внешнего вида диаграммы	
	Переместить сюда	Переместить панель	
	Удалить	Скрыть панель	

6.2.1.1. Общие настройки консоли

Нажмите кнопку ⚙ (*Настроить консоль)* и перейдите на вкладку *Общие*, чтобы выбрать один из режимов работы консоли (*Базовый, Расширенный*). В конфигурации *Расширенный* добавляется по умолчанию предустановленный набор панелей.

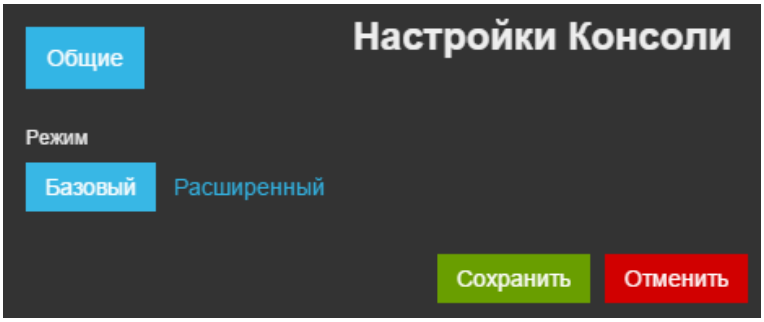


Рис. 29 – Конфигурация консоли

В расширенном режиме доступны следующие настройки консоли:

- Индекс;
- Ряды;
- Управление;
- Timepicker.

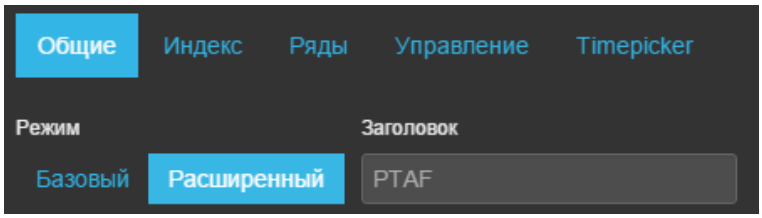


Рис. 30 – Настройки расширенно режима консоли

6.2.1.2. Генератор отчетов

Нажмите кнопку (*Отчеты)*, чтобы инициировать создание отчета, а затем настройте опции в открывшемся окне *Генератор отчетов*.

Возможные форматы итогового файла отчета – .csv, .pdf, .odt, .doc.

Внимание!

- Экспорт атак возможен только в отчет CSV.
- В отчетах формата .csv поля дат по умолчанию заполняются символами «#». Для того чтобы посмотреть содержимое такого поля, следует щелкнуть по нему мышью или растянуть столбец.

Готовые файлы отчетов можно посмотреть на вкладке *Инструменты -> Отчеты* (см. главу «Отчеты»).

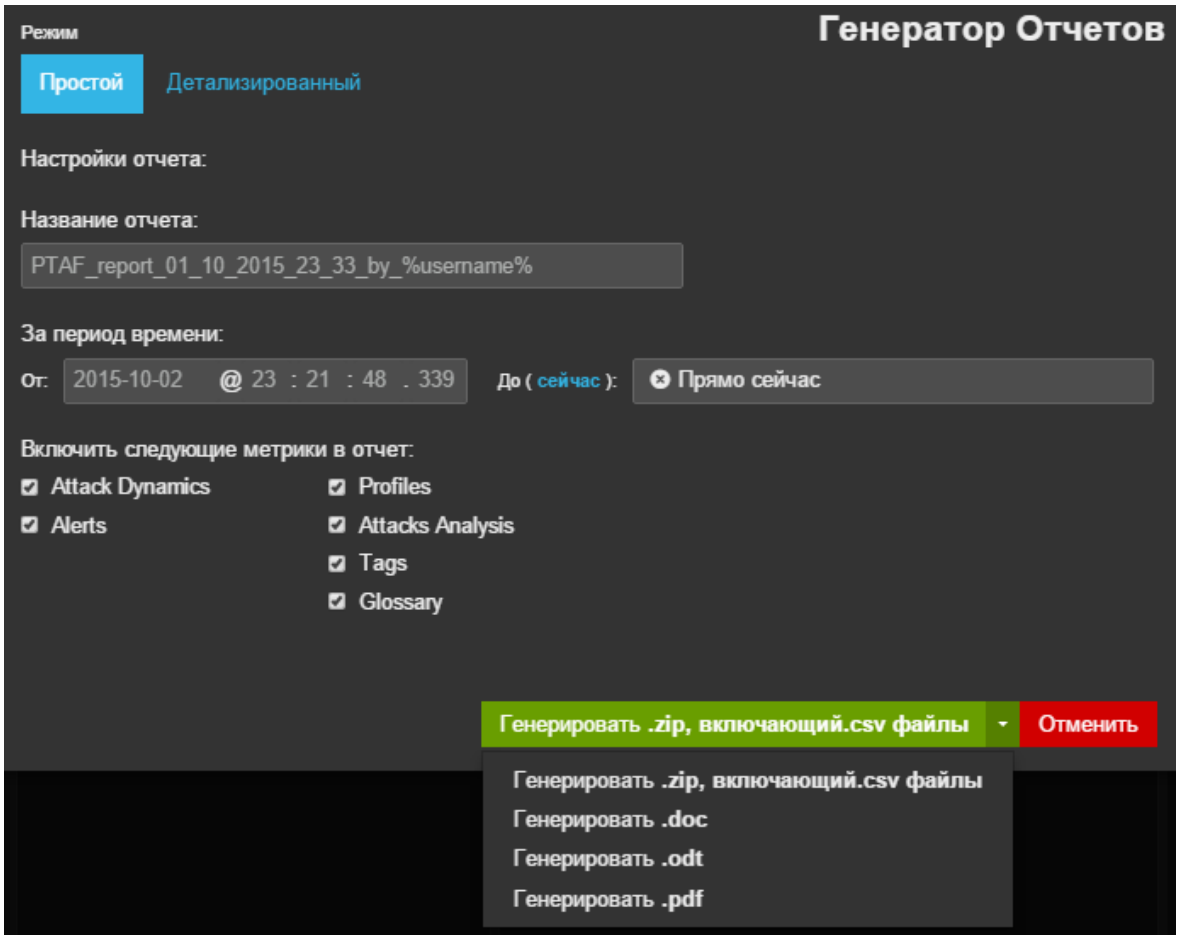


Рис. 31 – Окно генератора отчетов

6.2.2. Панели состояний

Внешний вид консоли гибко настраивается. Консоль состоит из панелей, которые размещены в виртуальной сетке состоящей из нескольких строк, в каждой из которых могут присутствовать несколько столбцов. Чтобы добавить новую строку в раздел *Консоль*, нажмите кнопку *Добавить ряд* внизу страницы. На экране откроется меню для редактирования списка рядов (Рис. 32), где вы также можете добавить новый ряд.

Внимание! Кнопка *Добавить ряд* доступна только в расширенном режиме.

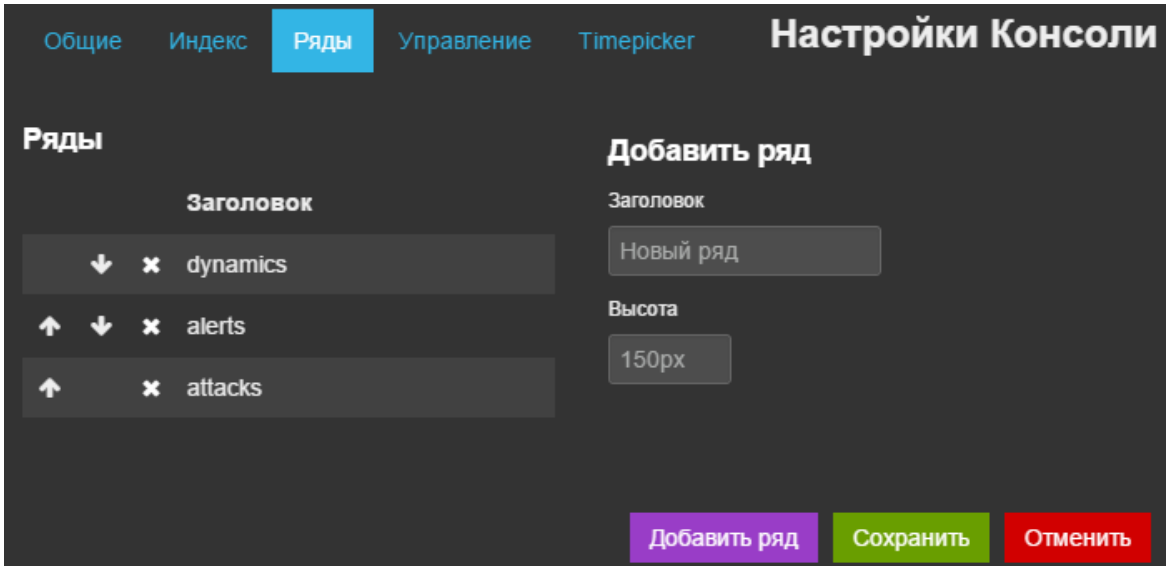


Рис. 32 – Редактирование списка рядов панелей

После добавления ряда настройте список его панелей. Для этого нажмите кнопку *Добавить панель в пустой ряд* или кнопку *Настроить*.

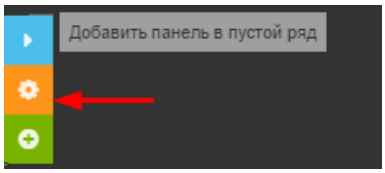


Рис. 33 – Добавление панели в пустой ряд

После этого откроется окно настройки рядов. Выберите тип панели, чтобы добавить готовую панель с настроенными параметрами. Настройки выбранной панели можно изменить в соответствии с потребностями. Если необходимо создать панель, которая не имеет предустановленные настройки, выберите тип *Пользовательский*.

В программе предусмотрены следующие типы панелей:

- Города;
- Динамика атак;
- Страны;
- Корреляции;
- Атаки;
- Браузеры;
- IP;
- Операционные системы;
- Профили;
- Теги;
- Пользовательский.

6.2.3. Запросы и фильтры

Данные для всех панелей PT AF собираются из базы данных Elasticsearch при помощи запросов панели *Query*, а затем подвергаются фильтрации при помощи фильтров с панели *Filtering*. Запросы и фильтры пишутся на языке Lucene Query String.

Так, чтобы найти слово Windows в поле os, надо ввести запрос:

```
os:Windows
```

Если в ключевом слове есть пробел, то надо взять его в кавычки:

```
os:"Windows 7"
```

Несколько последовательных запросов неявно объединяются через оператор ИЛИ. Если необходимо использовать оператор И, его следует явно указать:

```
os:"Windows 7" AND browser:Chrome
```

Допустимо использование скобок и явного оператора ИЛИ:

```
(os:"Windows 7" AND browser:Chrome) OR city:"Fort Worth"
```

Для исключения поисковых строк можно воспользоваться символом «–». Например, для поиска события, источником которого был браузер Google Chrome версии не 36.0:

```
browser:Chrome -browser:36.0
```

Допустимо использование символа «*» для указания любой последовательности символов в запросах. Например, найдем события, у которых описание (поле tag.description):

- начинается с Cross-Site:

```
tag.description:Cross-Site*
```

- начинается с Cross-Site и кончается точкой:

```
tag.description:Cross-Site*"."
```

Примечание: запрос tag.description:Cross-Site отличается от запроса tag.description:Cross-Site* тем, что в первом случае идет выборка событий, у которых в поле tag.description встречается слово Cross-Site в любом месте, а во втором случае – только в начале. То есть оператор * работает максимально жестко, и в этом отношении он не аналогичен оператору * в традиционных регулярных выражениях.

Используйте оператор ~ для того, чтобы описать как близко расположен один объект относительно другого. Так, этот запрос не может не соответствовать никаким событиям:

```
tag.description:"Cross-Site HTML"~10
```

А следующий запрос, напротив, соответствовать:

```
tag.description:"Cross-Site HTML"~100
```

Для задания диапазонов значений используйте конструкцию [1234 TO 4321]. Например:

```
timestamp:[2014-10-25T14:30:00.000Z TO 2014-10-26T00:00:00.000Z]
```

Так же диапазоны можно использовать для поиска по IP-адресам. Например, фильтр ищущий по IP-адресу источник атаки:

```
ip:[192.0.2.0 TO 192.0.2.128]
```

Запросы применяются для того, чтобы наполнить данными панели в консоли PT AF. По умолчанию применяются пять запросов:

```
event.severity:low
event.severity:medium
event.severity:high
```

```
_type:alert
_type:attack
```

Первые три применяются для создания графика Attack Dynamics (см. главу «Панель Attack Dynamics»). Запрос _type:alert применяется для создания панели Alerts, а _type:attack для панели Attacks (см. главу «Панель Attacks»).

6.2.3.1. Запросы (панель Query)

Для создания запросов в системе предусмотрена панель *Query*. Чтобы создать запрос, нажмите ✚ в последнем запросе и напишите новый запрос на языке Lucene.

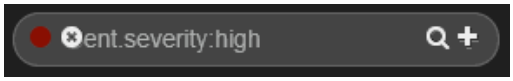


Рис. 34 – Запрос

Например, рядом с графиком *Attack Dynamics* требуется отобразить график динамики SQL-инъекций. Для этого удобно создать свой запрос и панель, отображающую результаты работы этого запроса. Выполните следующие шаги:

1. Создайте запрос tag.name:"SQL Injection".

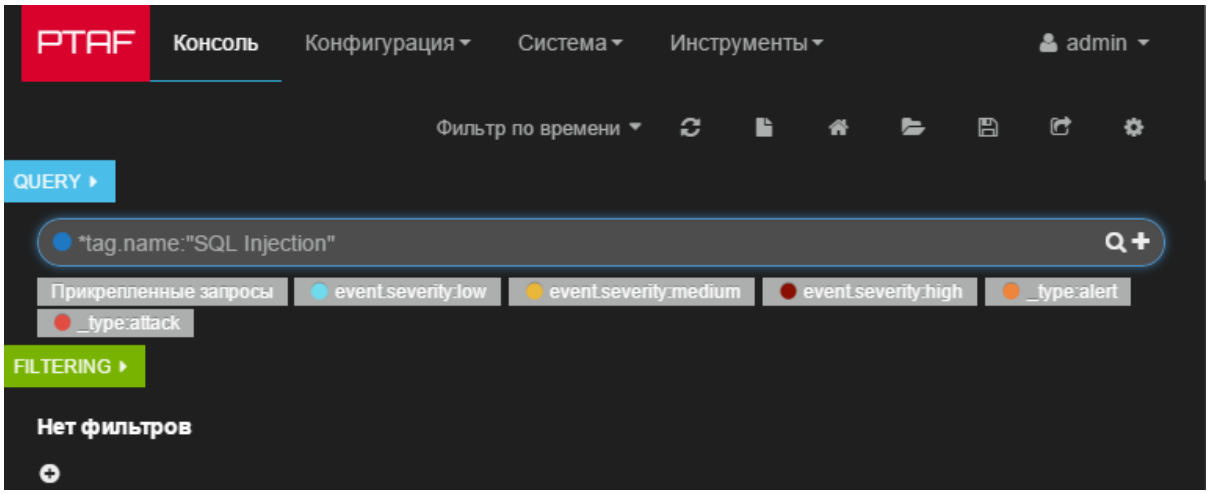


Рис. 35 – Создание запроса tag.name:"SQL Injection"

2. Создайте новую панель.

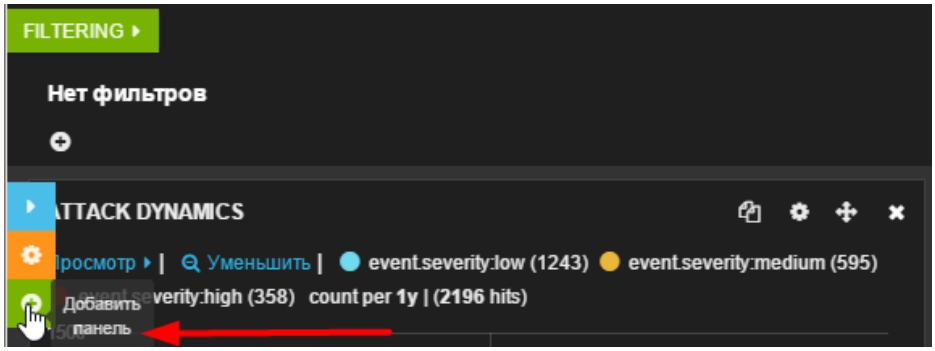


Рис. 36 – Создание новой панели

3. В появившемся диалоге выбираем тип панели *Динамика атак*. В появившемся далее меню достаточно в качестве источника данных выбрать созданный запрос, отредактировать название графика и нажать кнопку Сохранить.

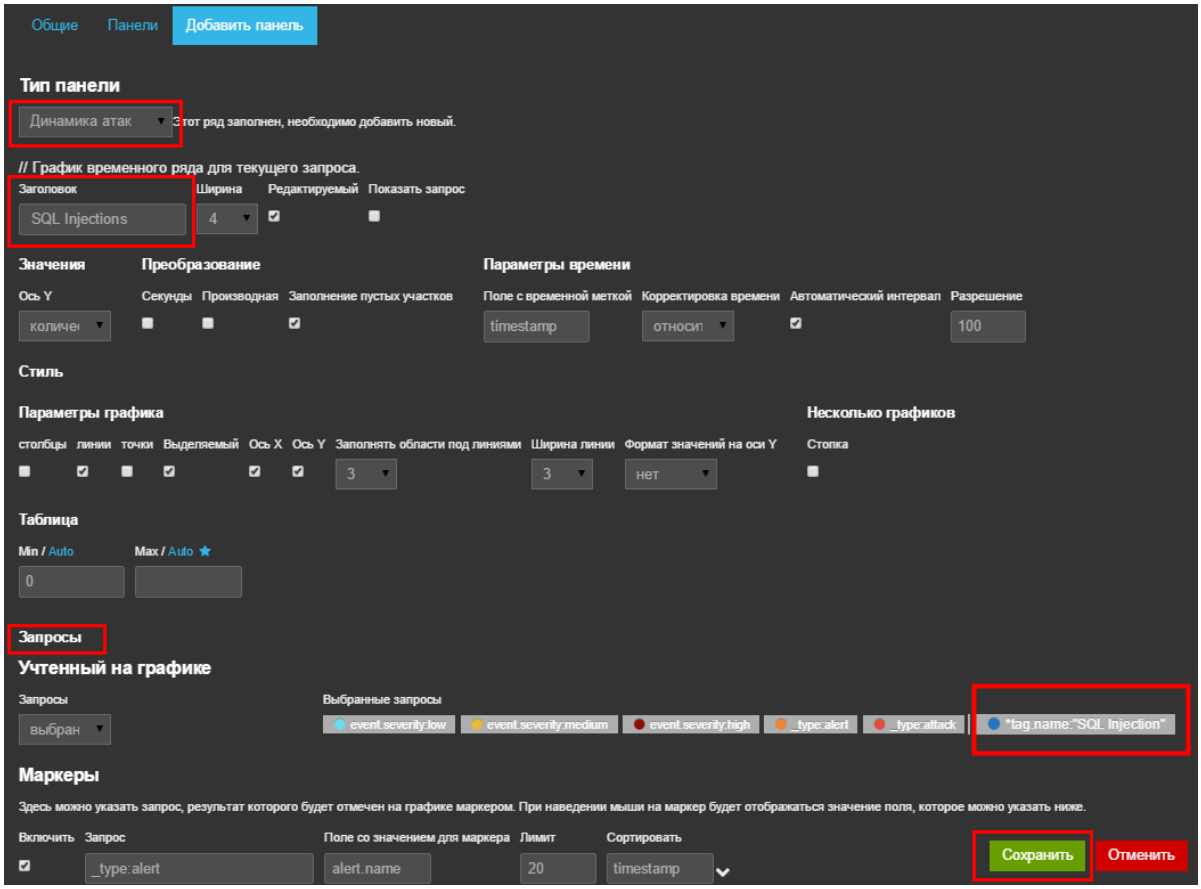


Рис. 37 – Заполнение полей

4. Теперь в консоли друг под другом расположены два графика: *Attack Dynamics* и *SQL Injections*:

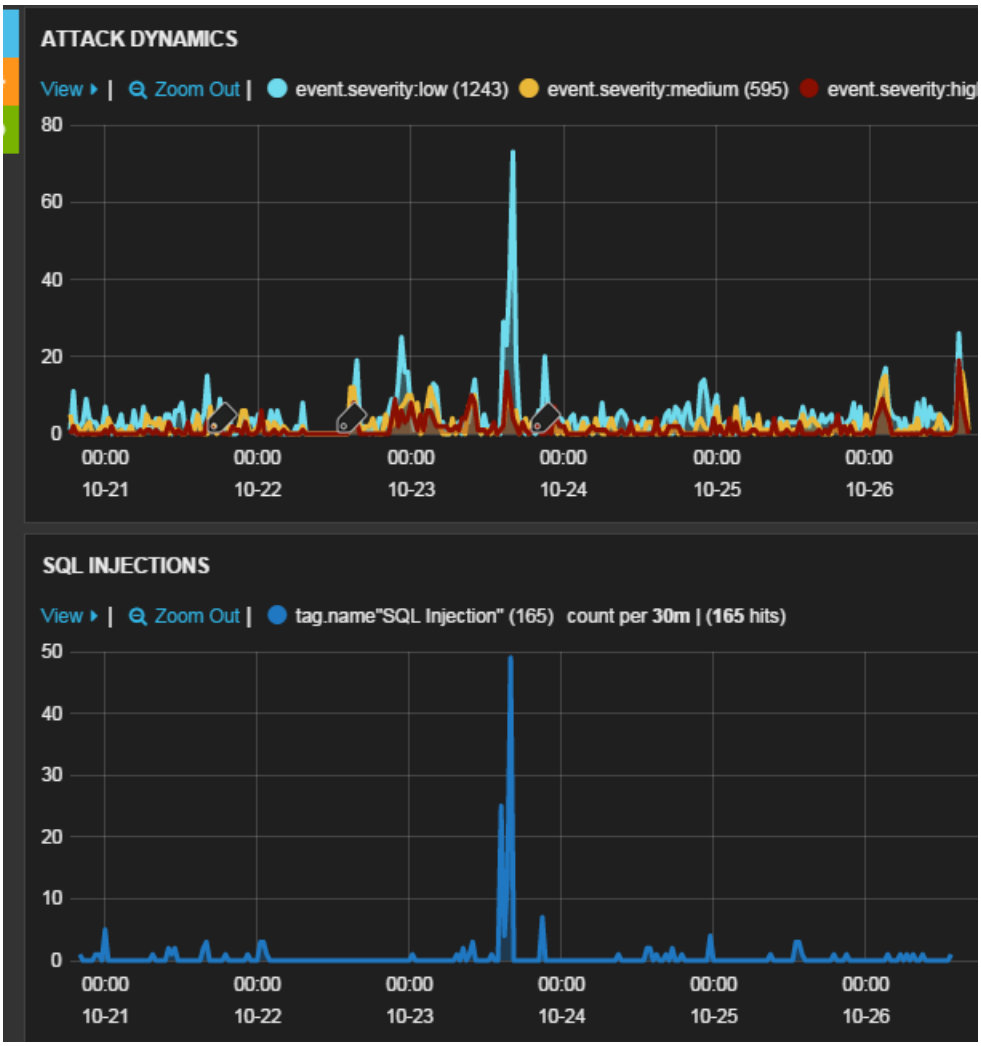


Рис. 38 – Графики Attack Dynamics и SQL Injections

Для изменения цвета, которым отображаются события на графике attack dynamics и в таблице attacks, щелкните мышью на цветном индикаторе в элементе, отображающем запрос. Чтобы удалить запрос, нажмите кнопку ✖, которая появляется при наведении мыши на поле запроса.

Для того чтобы запрос случайно не удалить и не изменить, следует выбрать пункт *Pin* Pin в меню запроса, выпадающее по нажатию цветового индикатора.

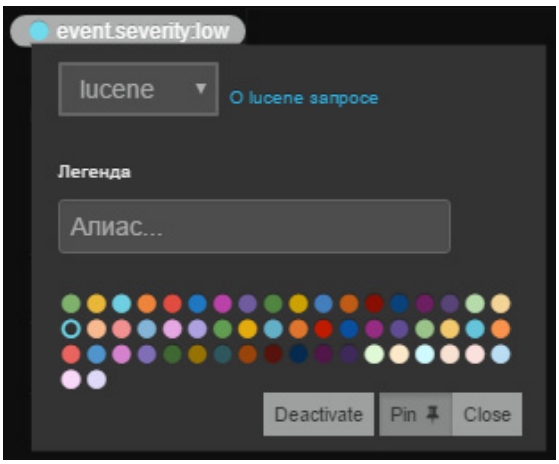


Рис. 39 – Закрепление запроса

6.2.3.2. Фильтры (панель Filtering)

Раскройте панель *Filtering* и нажмите кнопку ⊕ , чтобы создать новый фильтр. Например, для создании фильтра из выпадающего меню выберите действие фильтра *должен (mus*t), *не должен (mustNot)* или *любой (either)*, затем в запрос *query* напишите выражение tag.name:injection (на языке Lucene) и нажмите кнопку *Применить*.

Аналогично вы можете создать и другой фильтр данных, пропустив через него все события, выбранные запросом из панели *Query*.

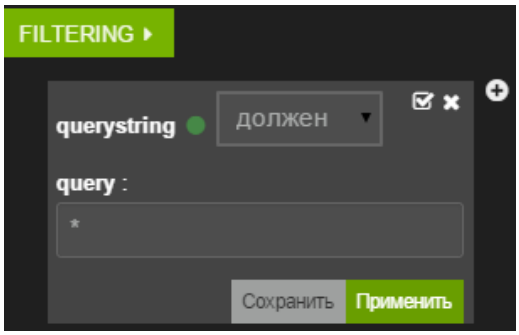


Рис. 40 – Создание фильтра

Кроме этого, наиболее универсального способа, есть и ряд более удобных, но менее универсальных способов создания фильтров с помощью интерактивных элементов в других панелях.

6.2.4. Панель Attack Dynamics

Рассмотрим подробнее панель с графиком Attack Dynamics (Рис. 41). В панели *Attack Dynamics* расположена информация о распределении случившихся событий разного уровня опасности во времени.

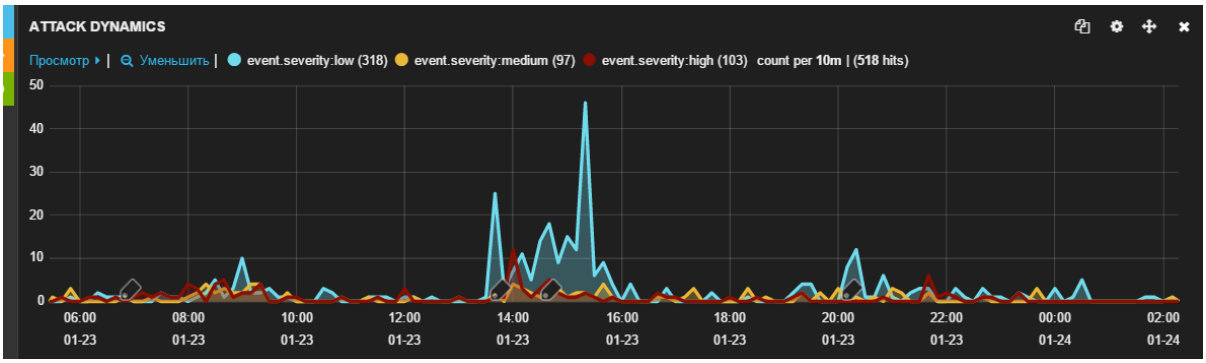


Рис. 41 – Панель Attack Dynamics

Общее количество случившихся событий указано в круглых скобках у значков легенды. Цвет линий графика можно настроить в панели *Query*.

Выделяя временной диапазон мышью прямо на графике, можно создать фильтр, который ограничит показ событий определенным промежутком времени (Рис. 42).

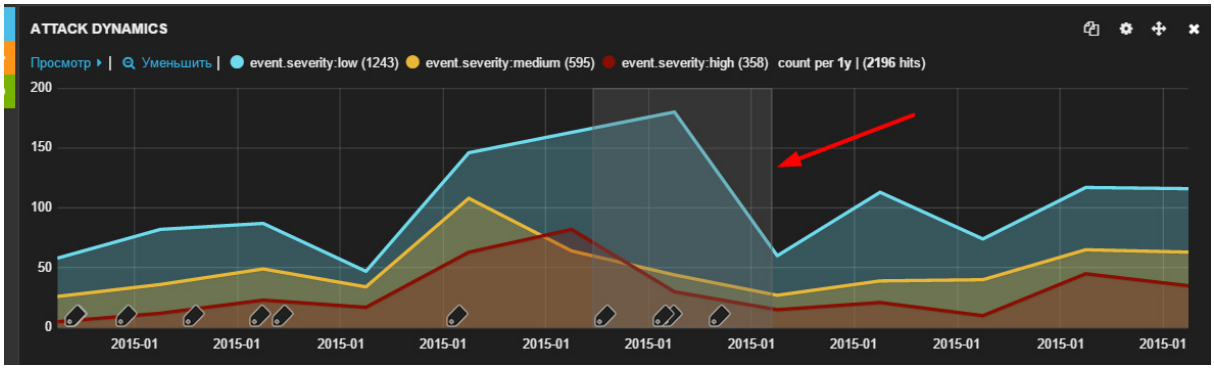


Рис. 42 – Добавление временного фильтра

Созданный фильтр отображается на выпадающем меню в верхней строке сайта, а так же на панели фильтров (Рис. 43), график при этом автоматически перерисовывается. Созданный фильтр влияет на отображение данных на всех панелях консоли, включая панель *Attacks*. Чтобы отменить созданный фильтр, разверните панель фильтров и нажмите кнопку *Удалить* около соответствующего фильтра (time must…).

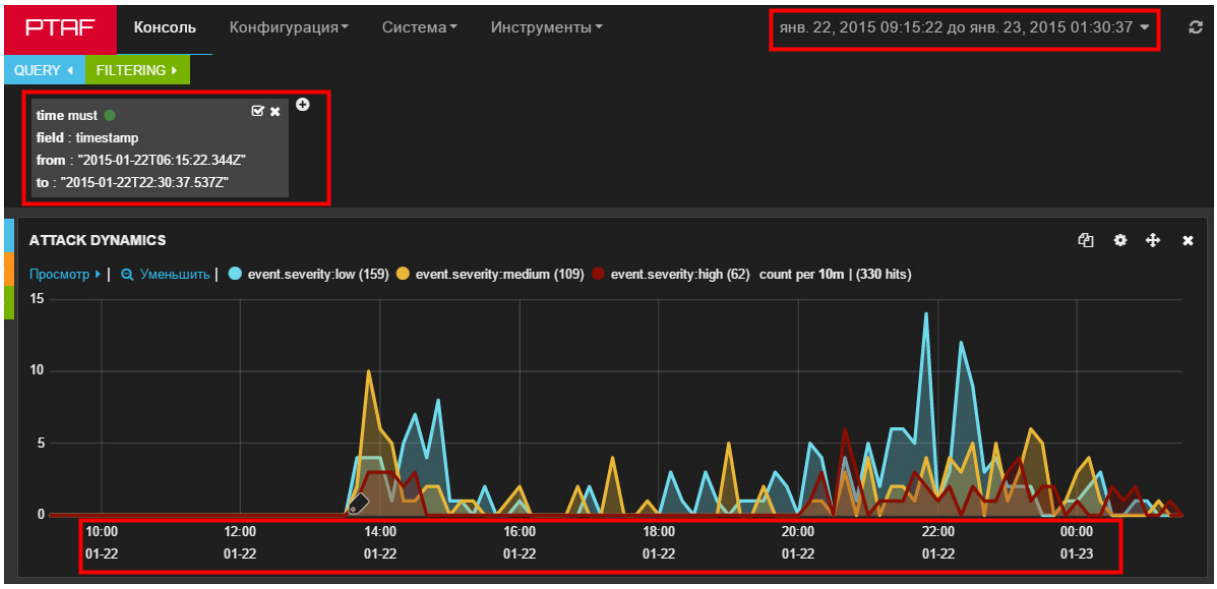


Рис. 43 – Временной фильтр

Фильтр времени можно задать и вручную используя Lucene Query Language. Например, добавить свой диапазон времени с действием *не должен* (*mustNot*).

```
timestamp:[2014-10-25T14:30:00.000Z TO 2014-10-26T00:00:00.000Z]
```

Это действие приведет к тому, что из всех панелей пропадут события из указанного временного поддиапазона.

6.2.5. Панель Alerts (корреляции)

В панели *Alerts* выводится информация о скоррелированных событиях, сгруппированных по признаку принадлежности к одной и той же атаке. За группировку отвечает поле *correlation_id*. Так, чтобы видеть только те события, которые принадлежат одному скоррелированному событию, надо сформировать фильтр correlation_id:"…", где указать идентификатор интересующей атаки. Например, на панели *Alerts* развернуть строку соответствующего события, перейти к полю *correlation_id* и нажать 🔍. Фильтр будет сформирован автоматически.

После формирования фильтра, во вкладке *Alerts* останется одна строчка (соответствующая данной атаке), во вкладке *Attacks* будут перечислены все события, связанные с данной атакой, а на остальных панелях (*Attack Dynamics, IP, Tags, Operating Systems, Browsers*) информация будет ограничена выборкой по созданному фильтру.

Alerts (уведомления) - это корреляции, которые могут включать множество событий. События могут принадлежать нескольким корреляциям. Каждый алерт описывает последовательность событий, события настраиваются во вкладке *События*. После настройки события можно использовать в алерте на вкладке *Корреляции*.

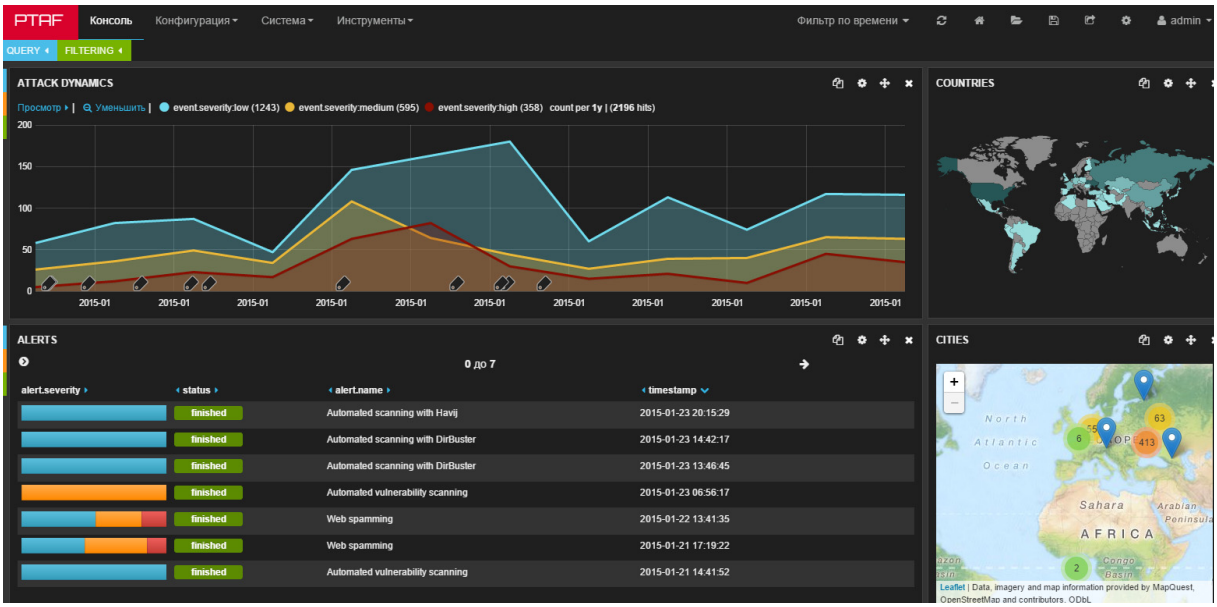


Рис. 44 – Панель Alerts

Алерты с атаками связаны по полю *correlation_id*, фильтрация по которому позволяет показать атаки, которые попали под данную корреляцию.

Алерты могут иметь статус *finished* (корреляция завершилась, таймаут истек) и *ongoing* - корреляция уже сработала, но продолжает обрабатывать входящие события, подпадающие под правило.

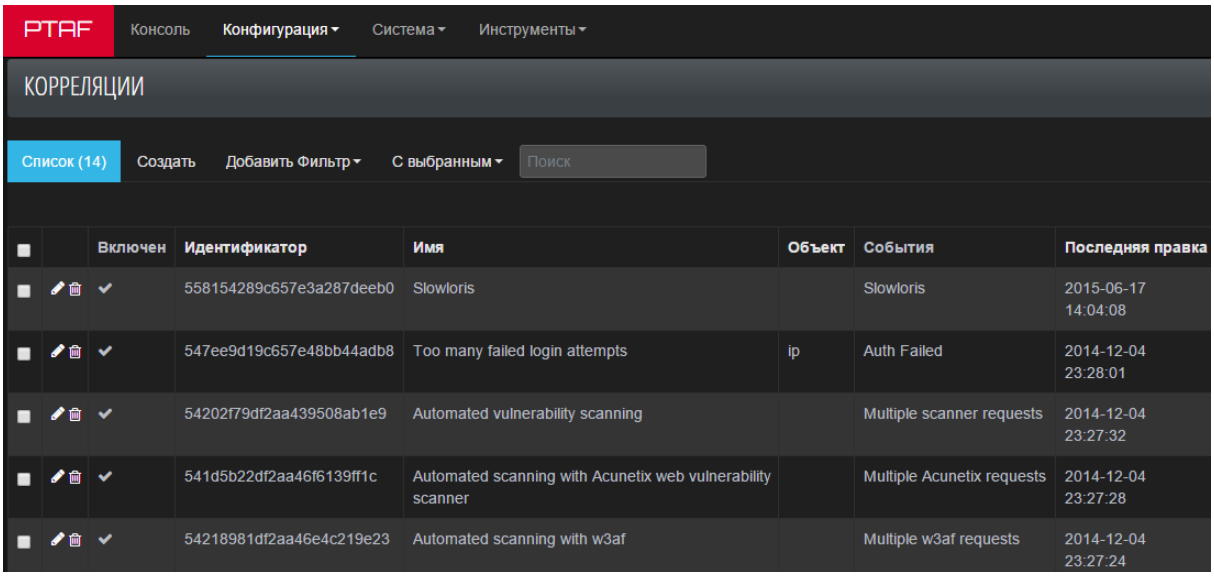


Рис. 45 – Корреляции

6.2.6. Панель Attacks

На панели *Attacks* приведена таблица, которая содержит перечень событий, выбранных из базы тремя запросами из панели *Query* (по трем уровням критичности) и отфильтрованных выражениями из панели *Filtering*. Столбцы в таблице выводятся согласно настройкам *Поля* слева от таблицы.

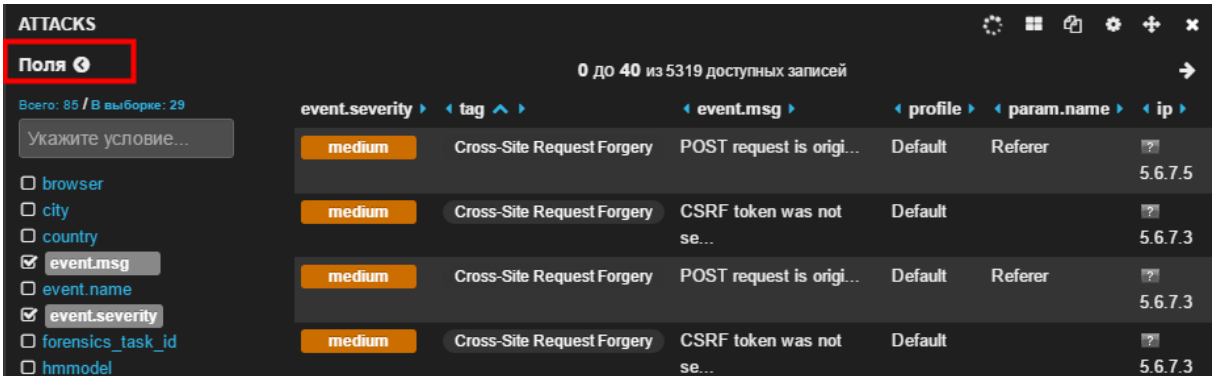


Рис. 46 – Панель Attacks

Если навести указателем мыши на одну из строк, справа появятся кнопки (см. Рис. 47) *Скан, Исключить, Блокировать*. Подробное описание кнопок представлено в главах 6.2.6.1. – 6.2.6.3.

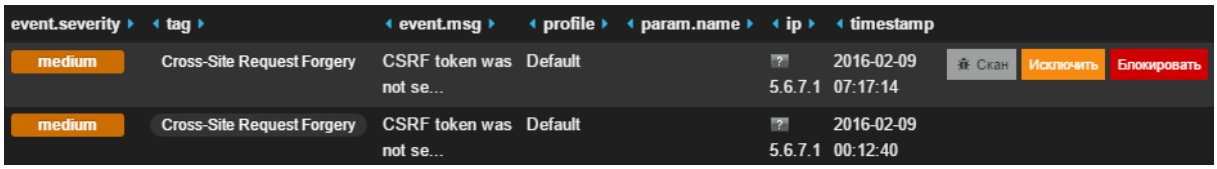


Рис. 47 – Таблица атак

Если нажать левой кнопкой мыши на любой строке в списке событий, она развернется в подробное описание события, в котором будут приведены все поля, включая те, которые не отображаются в таблице. В том числе отображается поле *request*, в котором приводится весь запрос предполагаемого злоумышленника.

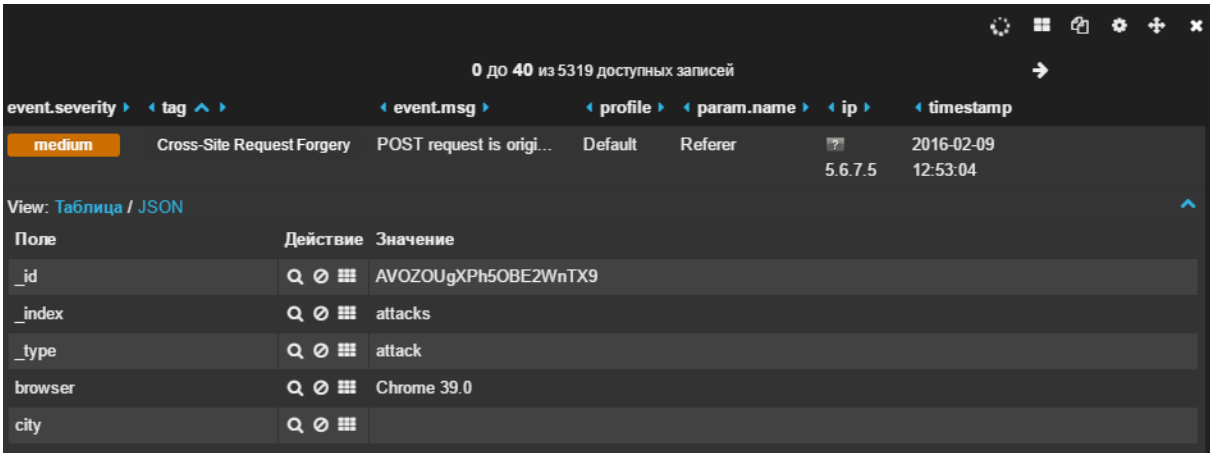


Рис. 48 – Таблица с подробным описанием события

Нажатие на кнопку 🔍 в столбце *Действия* (Рис. 49) приведет к автоматическому созданию фильтра, который ограничит вывод событий только теми событиями, в которых данное поле равно указанному значению (в строке). Например, если необходимо получить выборку из событий в которых сработали те же правила (rules) фильтрации, что и в заинтересовавшем событии, нажмите 🔍. Автоматически сформируется фильтр rules с действием *должен* (*must*). Во всех панелях останутся только те события, в которых сработали обе сигнатуры. И наоборот, нажатие на кнопку ⊘ приведет к тому, что фильтр добавится с действием *не должен* (*mustNot*), что удобно использовать для фильтрации ложных срабатываний. Нажмите кнопку ▦, если требуется показывать столбец в таблице.

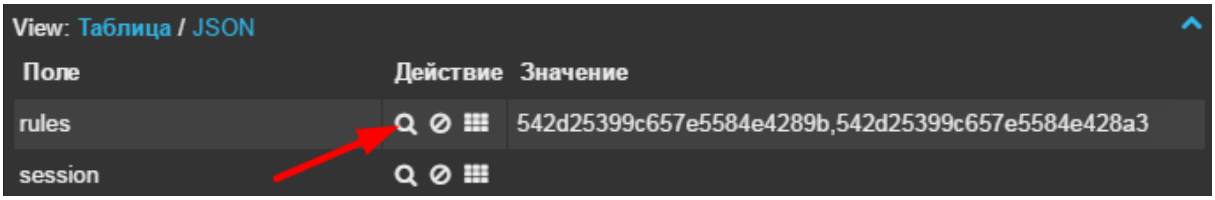


Рис. 49 – Поиск документов с указанным значением

6.2.6.1. Сканирование

При нажатии на кнопку *Скан* на панели *Attacks* или *Alerts* появляется диалоговое окно, в котором можно задать адрес сканируемого узла и затем просканировать его, чтобы убедиться в том, что по адресу атаки действительно имеется уязвимость, атака актуальна и важность данного события высока. Сканирование можно провести через прокси-сервер (HTTP/SOCKS 4/5), с аутентификацией или без.

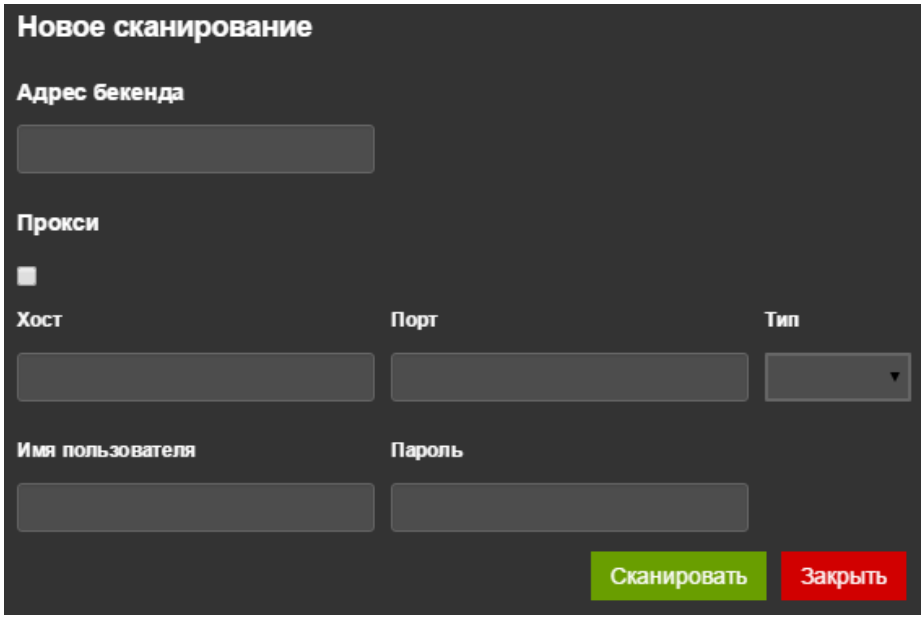


Рис. 50 – Проверка уязвимости

В поле *Адрес бекенда (Backend Address)* следует указать адрес защищаемого приложения, а в поле URL - указать проверяемый URL. Важно, чтобы в поле URL находился не IP-адрес, а доменное имя сканируемого узла, даже если в контексте PT AF это доменное имя разрешается не в адрес реально сканируемого узла, т.к. если это

условие не будет соблюдено, то сканируемый узел может не передать запросы сканера правильному виртуальному узлу.

Во время сканирования узла на месте кнопки *Скан* отображается кнопка *Стоп*, позволяющая остановить процесс сканирования. Если сканер не обнаружил проблем по указанному адресу, то событие помечается как безопасное кнопкой *Safe*. Кнопка позволяет, в том числе, выполнить повторный скан приложения. Если же уязвимость подтвердилась, то эта же кнопка будет подписана как *Alert*.

Рис. 51 – Кнопки сканирования

6.2.6.2. Исключение и восстановление атаки

С помощью кнопки *Исключить* можно отключить по определенным признакам правило, приведшее к появлению события в базе, порождающее ложные срабатывания. События так же пропадут из консоли. Кнопка исключения атаки доступна для каждой атаки.

Нажмите кнопку *Исключить*, на экране появится окно *Attack Exclude Dialog*, в котором указываются параметры исключения. Включите необходимый тип исключения атаки:

- Global Excludes – создает правила исключения на будущее, т.е. исключенные атаки не будут появляться в консоли. Эти правила исключения будут храниться в профиле в разделе *Глобальные*;
- Protector Excludes – создает правила исключения на будущее, т.е. исключенные атаки не будут появляться в консоли. Эти правила исключения будут храниться в профиле в разделе *Модули*;
- Исключить атаки по профилю – скрывает существующие атаки по профилю;
- Исключить атаки по сработавшему модулю защиты – скрывает существующие атаки по сработавшему модулю защиты;
- Исключить атаки по параметру или по регулярному выражению – скрывает существующие атаки по параметру или по регулярному выражению. Регулярное выражение указано в соответствующем поле;
- Исключить атаки по источнику параметра – скрывает существующие атаки по источнику параметра;
- Исключить навсегда – удаляет исключенные атаки физически.

Нажмите кнопку *Применить*.

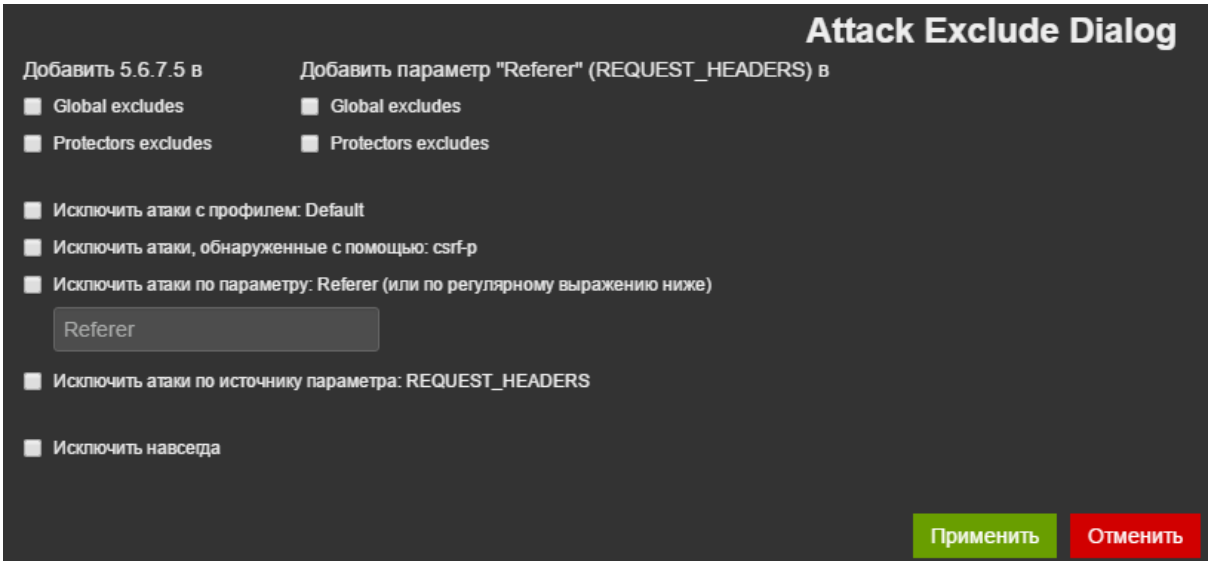


Рис. 52 – Диалоговое окно исключения

Показать исключенные атаки можно, убрав соответствующие фильтры в панели *Filtering*.

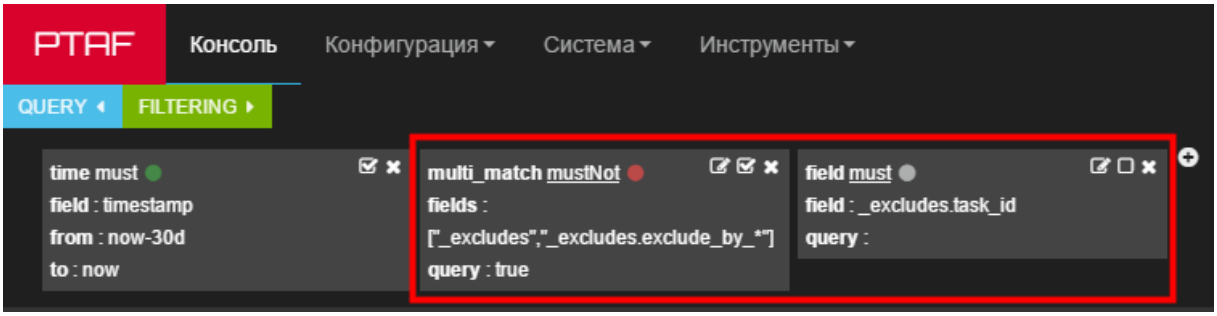


Рис. 53 – Установленные фильтры

У каждой исключенной атаки появляется кнопка *Восстановить*. С ее помощью атака может быть восстановлена.

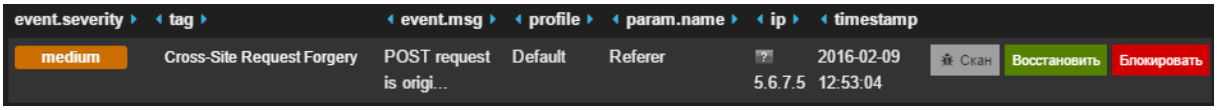


Рис. 54 – Строка атаки в таблице атак

Нажмите кнопку *Восстановить*, на экране появится окно *Attack Exclude Dialog*, в котором указываются параметры исключения. Отключите функцию исключения. Нажмите кнопку *Применить*.

6.2.6.3. Блокировка

Нажмите кнопку *Блокировать*, на экране появится окно блокировки (см. Рис. 55), где можно заблокировать одни из атрибутов атаки: *Сессию* (Cookie, сгенерированная PT AF), *Пользователя* (User Agent+IP) или *IP-адрес*. При выборе блокировки по сессии или пользователю происходит занесение сессии в список подозрительных (см. главу

«Подозрительные сессии»), при блокировке по IP происходит занесение IP-адреса в список заблокированных IP-адресов (см. главу «Файрвол»). Также в окне *Attack Block Dialog* можно указать длительность блокировки (*5 минут, 30 минут, 1 час, 3 часа, 24 часа, Навсегда*).

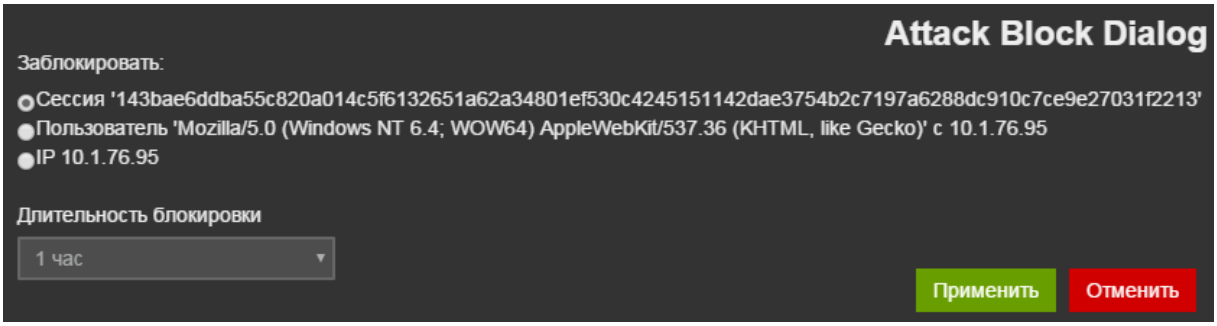


Рис. 55 – Окно блокировки атаки

6.2.6.4. Группировка

В панели *Attacks* данные могут быть представлены в сгруппированном виде. Группировка возможна только в расширенном режиме консоли. Включите расширенный режим (см. главу «Общие настройки консоли»), чтобы настроить группировку.

Пользовательская группировка предусмотрена для удобства просмотра списка атак. Группировка позволяет сворачивать в одну строку большое количество одинаковых событий на основе выбранного пользователем атрибута.

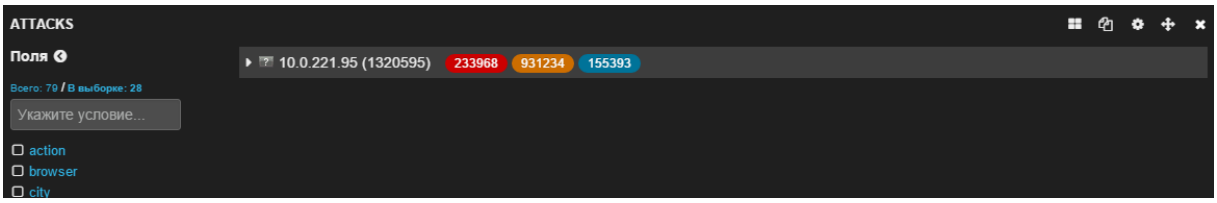


Рис. 56 – Группировка данных в панели Attacks

Нажмите на кнопку *Attack Grouping*  и выберите пункт меню *Set Grouping*, чтобы активировать группировку данных или изменить значения параметров группировки.

Внимание! Группировка возможна только в расширенном режиме консоли.

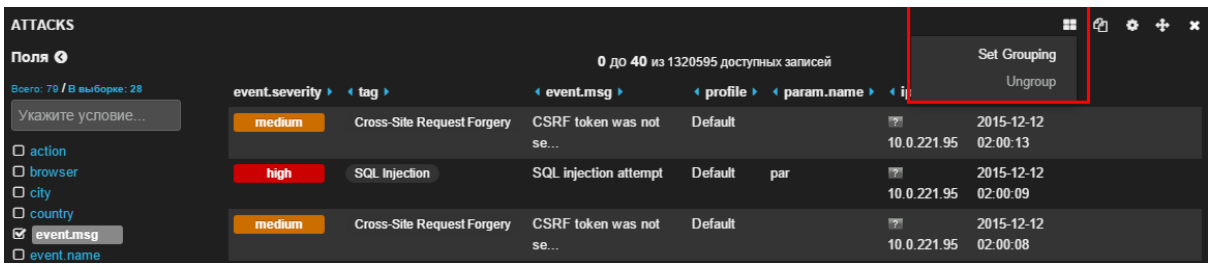


Рис. 57 – Активирование группировки

Для того чтобы начать работать со сгруппированными данными, в окне *Attack Grouping* достаточно выбрать группировку на уровне 1 (уровень 2 опционально) и нажать кнопку

Применить. Дополнительно можно настроить ограничения на количество выводимых строк и атак.

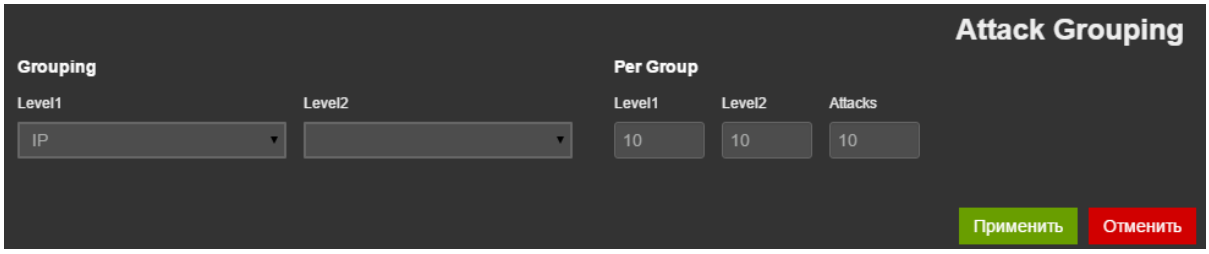


Рис. 58 – Окно настройки группировки

Если настроена группировка второго уровня, то в панели *Attacks* напротив заголовков первого уровня группировки будут отображены топ 3 значений из подгруппы. Под каждым значением размещен цветовой идентификатор в виде полоски, показывающий соотношение количества атак по параметру *Severity* («Важность»). В программе предусмотрены следующие значения параметра *Severity*:

- Высокий уровень – красный цвет;
- Средний уровень - оранжевый цвет;
- Низкий уровень – синий цвет.

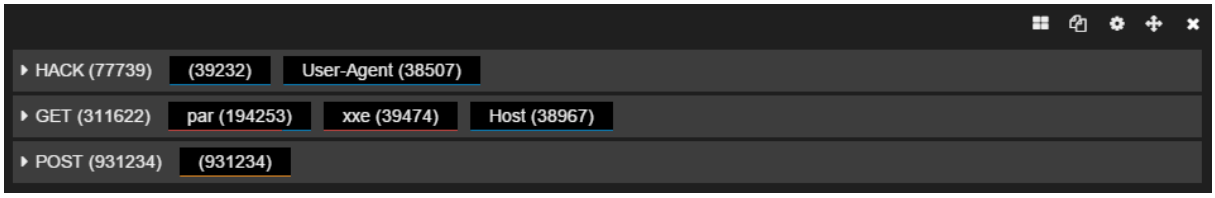


Рис. 59 – Заголовки первого и второго уровней группировки

Около заголовков второго уровня группировки отображены цветовые идентификаторы в виде кругов с цифровыми значениями количества атак с соответствующим уровнем «важности».

Если настроена группировка только одного уровня, то напротив заголовков группировки также отображены цветовые идентификаторы с цифровыми значениями количества атак с соответствующим уровнем «важности».

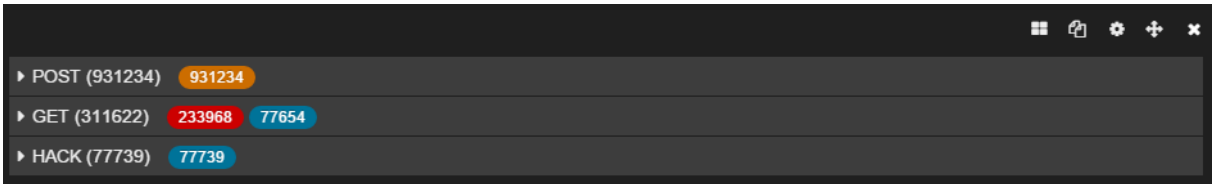


Рис. 60 – Цветовые идентификаторы

Чтобы отменить группировку данных, нажмите на кнопку *Attack Grouping* и выберите пункт меню *Ungroup*.

Для просмотра данных по группе следует раскрыть группу, щелкнув на ней левой кнопкой мыши. Данные по подгруппам раскрываются аналогичным образом.

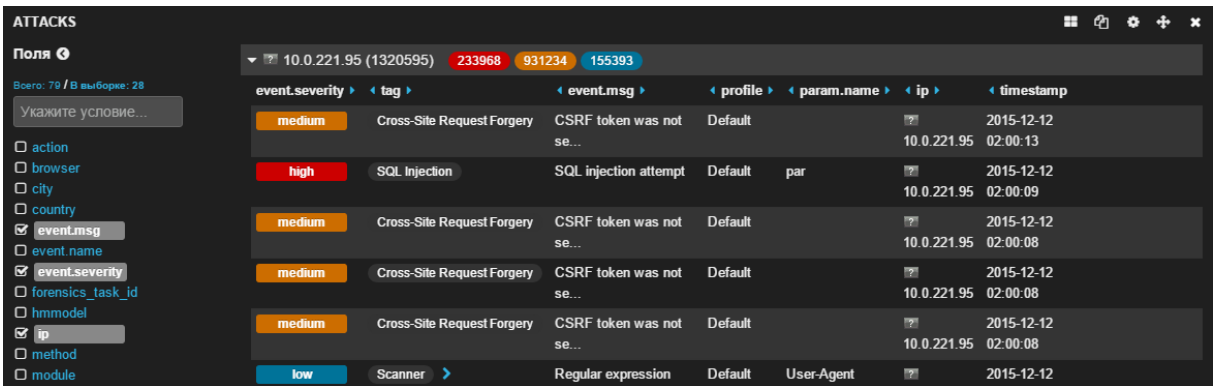


Рис. 61 – Раскрытие данных по группе

6.3. Конфигурация

Для того чтобы выполнить конфигурацию используйте следующие разделы системы:

- Политики безопасности;
- Сеть;
- Корреляция;
- Конфигурация;
- SSL-сертификаты и ключи;
- Подозрительные сессии.

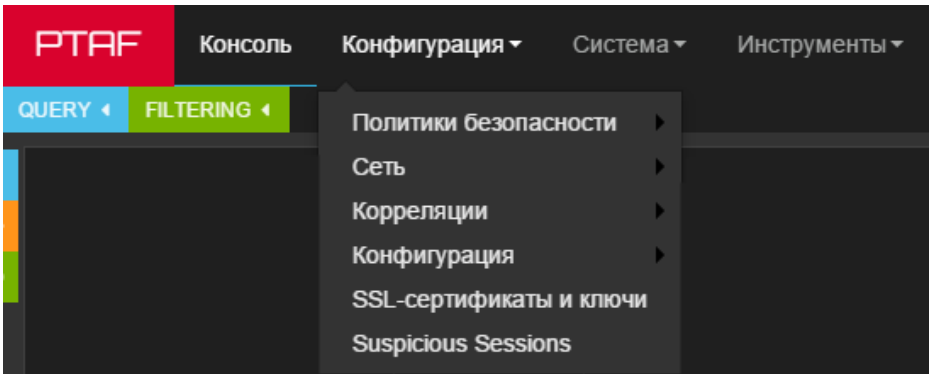


Рис. 62 – Конфигурация

В каждой вкладке, поддерживающей работу с записями, размещены элементы управления, которые могут быть следующими:

- Кнопка (*редактировать*) - нажмите, чтобы отредактировать запись;
- Кнопка (*удалить*) - нажмите, чтобы удалить запись;
- Опция *С выбранным* - выберите опцию для одной или нескольких записей, чтобы выполнить команду *С выбранным - Удалить* или *С выбранным - Включить/ выключить модель*;
- Кнопка (*активировать конфигурацию*) - нажмите, чтобы активировать настройку общего модуля.

6.3.1. Политики безопасности

Раздел *Политики безопасности* содержит следующие вкладки:

- Профили;
- Правила;
- Действия;
- Теги;
- HMM-модели;
- Content Security Policy;
- Черный список IP-адресов:
  - Черный список IP-адресов;
  - Черный список хостов;
- XML-схемы;
- Подпись форм:
  - Политики;
  - Регулярные выражения.

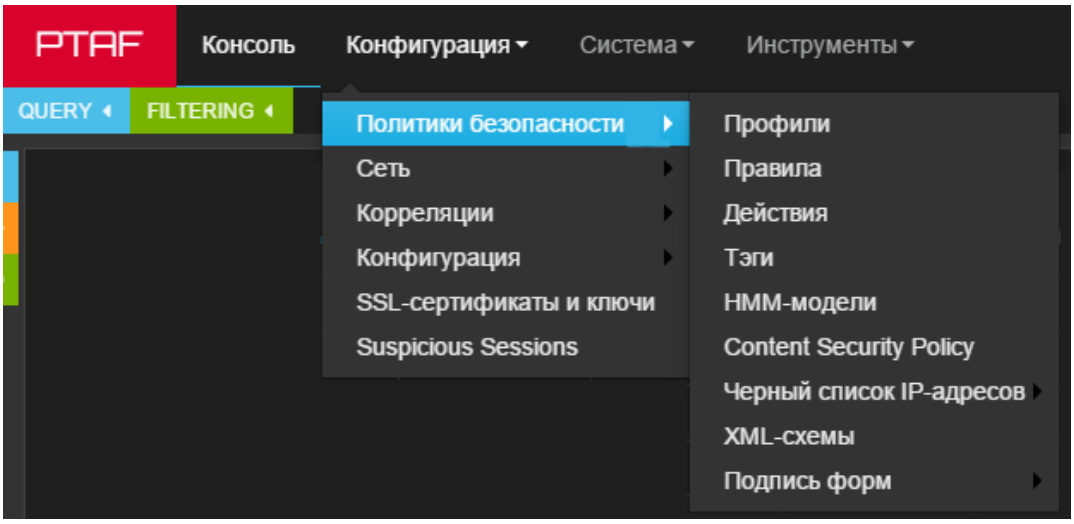


Рис. 63 – Политики безопасности

6.3.1.1. Профили

На вкладке *Профили* конфигурируются защитные профили, используемые PT AF в своей работе. Нажмите кнопку *Создать* для подготовки нового профиля.

Нажмите кнопку *Редактировать*, чтобы изменить настройки выбранного профиля. Пример редактирования профиля представлен в главе «Настройка расшифровки SSL-трафика» - «Режим блокирования атак (обратный прокси)».

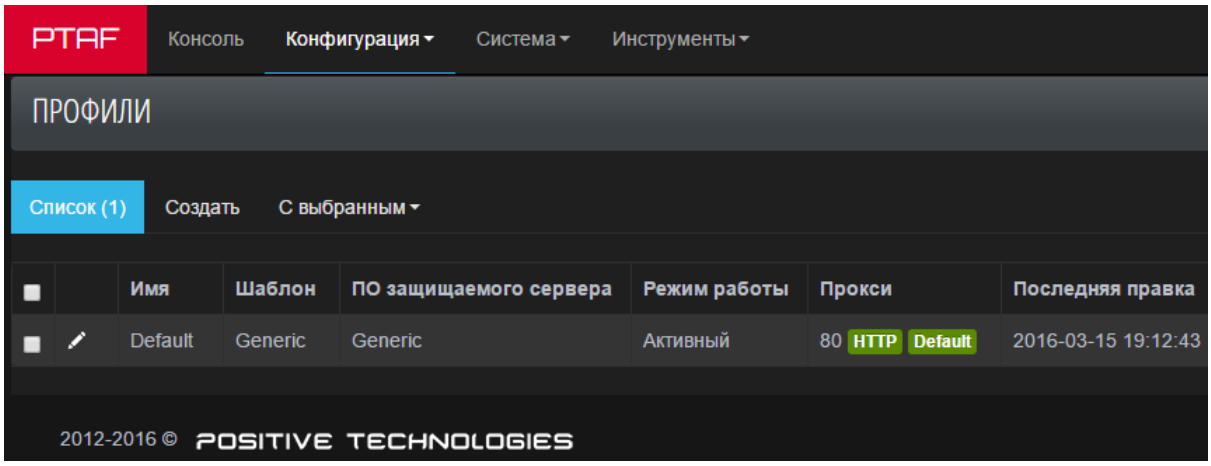


Рис. 64 – Профили

Настройки профиля разделены на шесть групп:

- Основные;
- Прокси;
- SSL;
- Модули;
- Глобальные;
- Разное.

6.3.1.1.1. Основные

На вкладке *Основные* собраны такие настройки профиля, как *Имя*, *Шаблон*, и др.

В опции *Хост* задается имя виртуального узла, защищаемого данным профилем. Одному профилю может соответствовать несколько виртуальных узлов, но нельзя указывать один виртуальный узел в нескольких профилях.

Опция *ПО защищаемого сервера* нужна для того, чтобы PT AF мог учитывать в своей работе специфику функционирования того или иного программного обеспечения. Возможные варианты: *Generic, PHP/Apache, ASP/IIS, ASP.NET/IIS, JSP/Tomcat*.

Режим работы - основная опция, которая может принимать одно из следующих значений: *Пассивный*, *Только обнаружение* и *Активный*. В случае, если PT AF работает в режиме мониторинга, *Режим работы* может принимать значение *Пассивный* или *Только обнаружение*. В первом случае система не будет собирать информацию. Во втором случае система будет собирать информацию для дальнейшего анализа на странице *Консоль* и может предотвращать некоторые атаки, отправляя сигнал закрытия соединения (TCP RST). При работе в режиме обратного прокси-сервера и в пассивном режиме работы PT AF не будет ни блокировать трафик, ни собирать данные для дальнейшего анализа. Однако при этом может работать проксирование, а также другие функции, как, например, SSL-шифрование трафика. В режиме *Только обнаружение* PT AF, работающий как обратный прокси-сервер, не будет блокировать трафик, но будет его обрабатывать. В активном режиме PT AF будет блокировать атаки злоумышленников.

Опция *Сниффер подключен* отвечает за то, работает ли сниффер с данным профилем, и имеет смысл при режиме работы *Только обнаружение*. В этом случае (см. главу

«Сниффер»), трафик, приходящий со span-порта коммутатора через службу tapered попадает в waf-nginx, при этом номер порта назначения переписывается на значение, указанное в меню *Конфигурация -> Сеть -> Сниффер*. Поскольку трафик из tapered в waf-nginx попадает через доменный сокет, нет необходимости, чтобы waf-nginx слушал порт, указанный в опции *ListenPort*. Включите опцию, чтобы указать waf-nginx'у, что он должен получать трафик из доменного сокета.

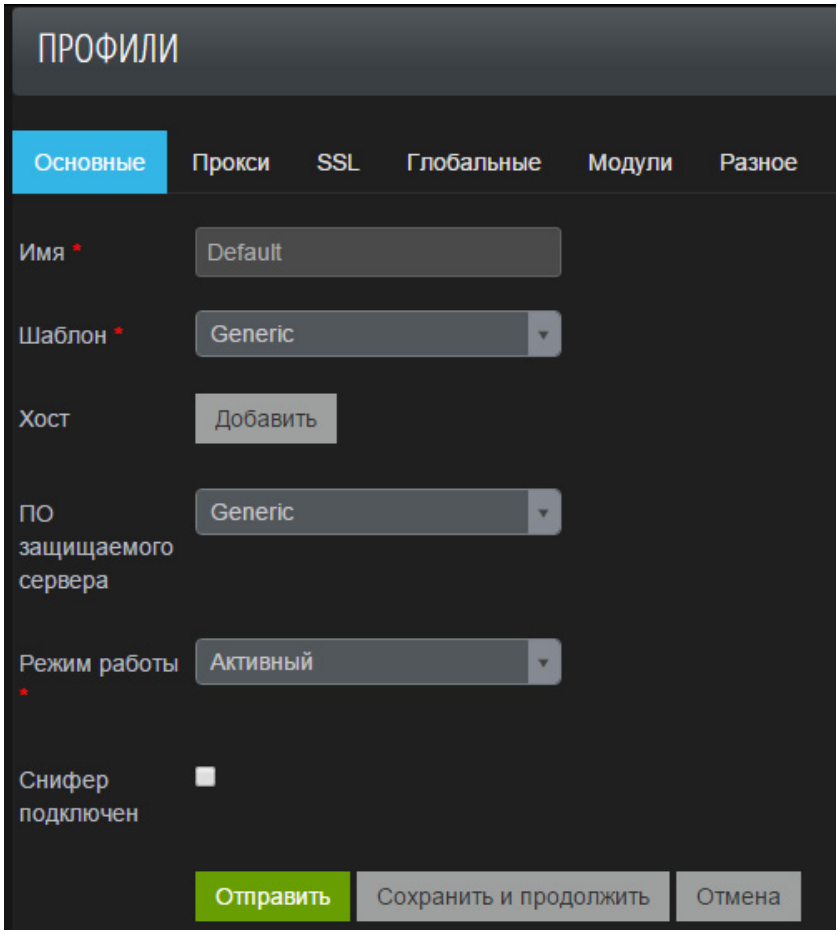


Рис. 65 – Основные настройки профиля

6.3.1.1.2. Прокси

Вкладка *Прокси* предназначена для того, чтобы пробрасывать трафик, приходящий на внешний порт PT AF, на сконфигурированную ранее группу серверов (см. главу «Группа серверов»). Здесь можно задать несколько групп настроек, которые будут одновременно применяться для текущего профиля.

В поле *IP-адрес* задается алиас, соответствующий одному или нескольким сетевым интерфейсам, а ниже указывается внешний порт, который будет слушать PT AF для того, чтобы пробрасывать пришедший туда HTTP-трафик на защищаемые сервера (см. главу «Группа серверов»).

Опция *Сервер по умолчанию* используется в том случае, если весь трафик, попадающий на данный порт, у которого имя узла (HTTP-заголовок Host) не соответствует никакому конкретному профилю, должен попасть именно в этот порт. (Ср. с настройкой Default Profile в настройках шлюза. Последняя нужна для того, чтобы в контексте шлюза

глобально указать какой профиль должен обрабатывать трафик, который в waf-nginx попал но из-за некорректного названия узла не попал ни в один явно назначенный профиль.) Дополнительные настройки можно задать при помощи опций *Включен* (при этом необходимо корректно настроить SSL на следующей вкладке), *SSL on Client cert* (требовать ли наличия SSL-сертификата у подключающегося клиента) и *Pass client info via Header*. Далее можно указать, для какой группы серверов будет работать этот профиль проксирования, а также протокол для связи с защищаемыми серверами: http или https.

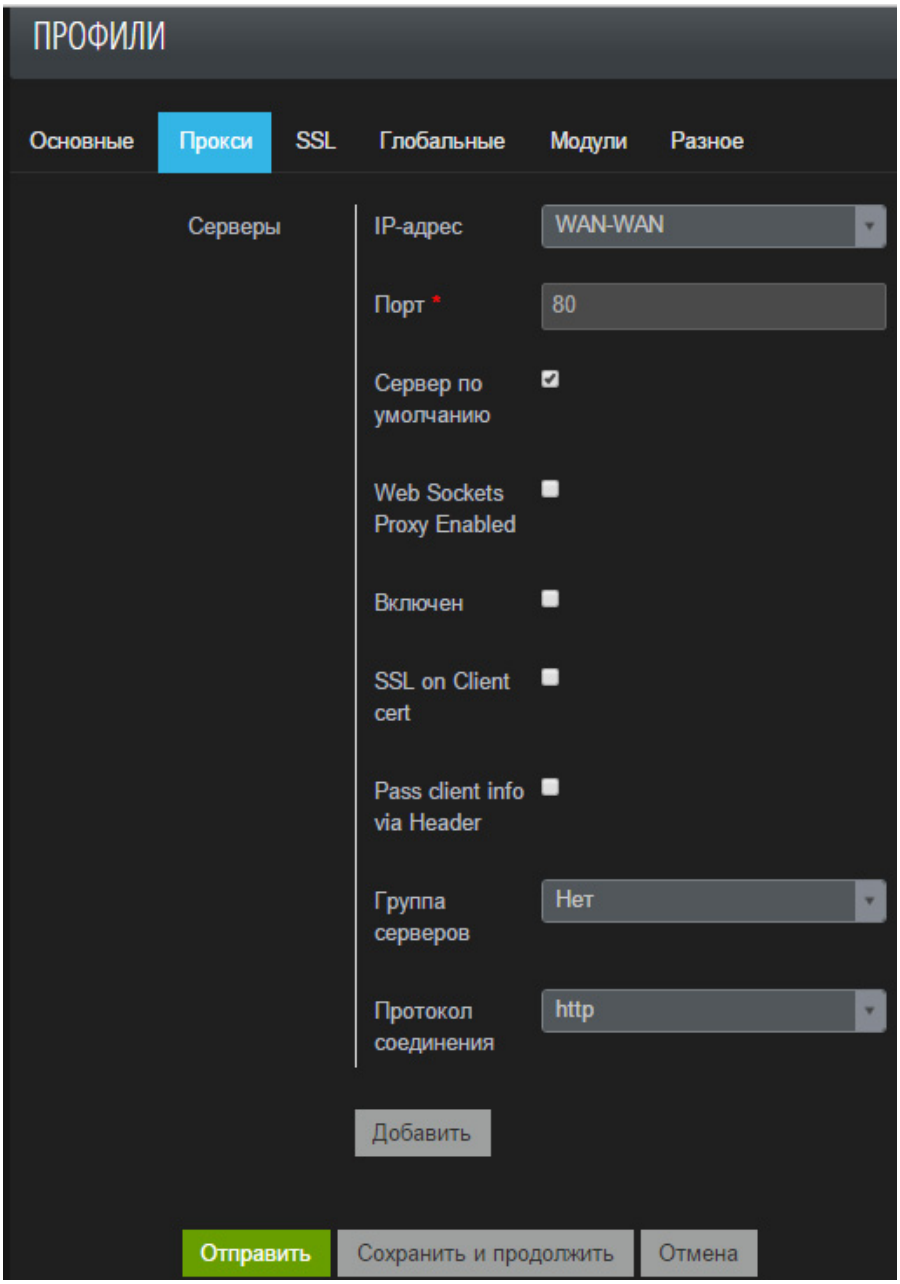


Рис. 66 – Настройки прокси

6.3.1.1.3. SSL

В случае, если PT AF защищает HTTPS-трафик, здесь следует указать настройки расшифровки трафика для его дальнейшей отправки на защищаемые сервера. Предварительно необходимо загрузить на сервер сертификат веб-сайта и приватный ключ (см. главу «SSL-сертификаты и ключи»), а так же указать необходимые криптопараметры. Для режима обратного прокси-сервера реализована поддержка алгоритмов шифрования ГОСТ.

SSL не будет работать, если не указать SSL-сертификат и приватный SSL-ключ.

Поле *SSL-шифры* необходимо заполнить в соответствии со стандартом, поддерживаемым библиотекой OpenSSL (обычно достаточно нажать кнопку *Использовать рекомендованные настройки* внизу страницы), а в списке SSL-протоколов выбрать те, поддержка которых требуется. Дополнительно укажите сертификат, используемый для подписи клиентских сертификатов (когда используется клиентская аутентификация при помощи сертификатов), а также список (CRL) недействительных клиентских сертификатов.

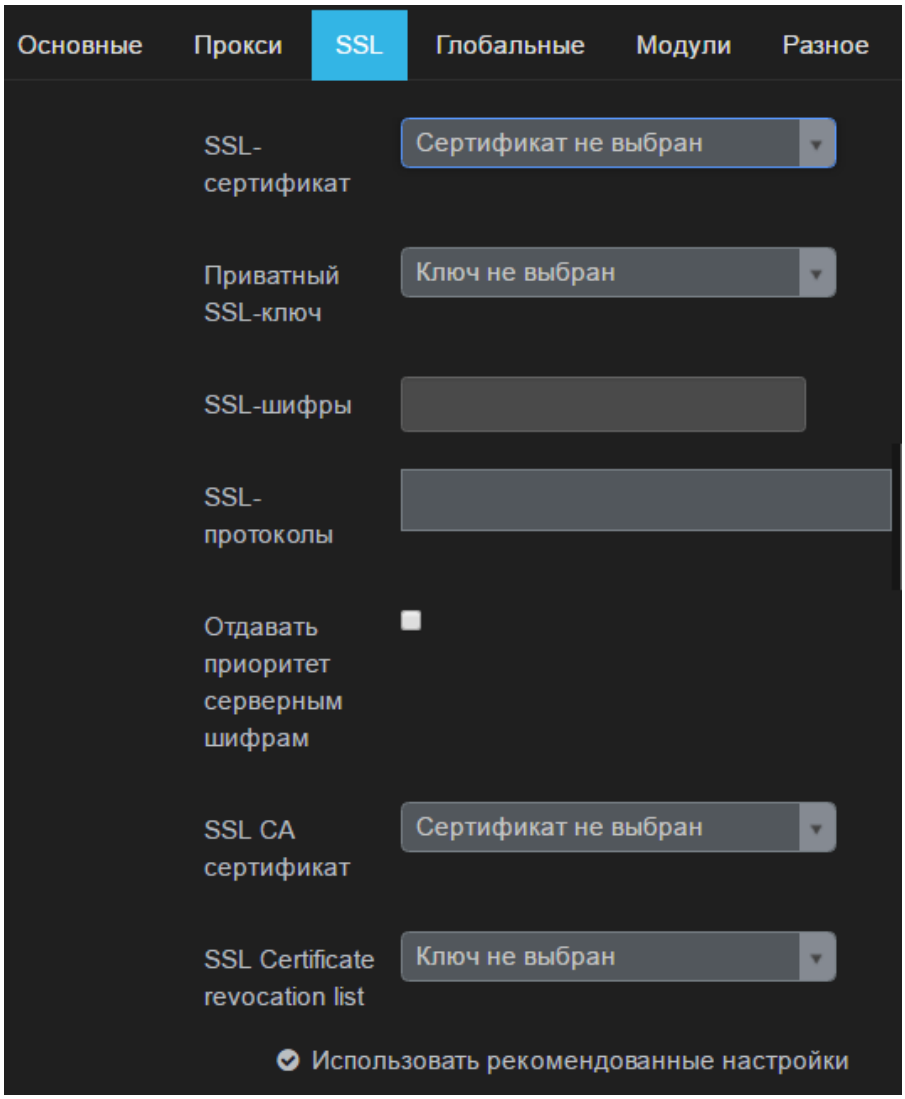


Рис. 67 – Настройки SSL

6.3.1.1.4. Глобальные

Во вкладке *Глобальные* задаются глобальные значения параметров модулей защиты (подробную информацию по модулям см. в главе «Конфигурация модулей защиты»).

В списке действий можно указать, которое из них должно применяться во всех случаях. Так, если глобально задать действие *Send to syslog*, то это действие будет выполняться при любом событии, т.е. при срабатывании любого модуля защиты. Если глобально задать действие *Block request*, то, если PT AF работает в режиме *Active Prevention* первый же сработавший модуль защиты (модули защиты срабатывают в определенном порядке, см. ниже) обязательно заблокирует трафик, что приведет к тому, что на другие модули защиты трафик уже не пойдет.

Здесь же можно глобально задать переменные, отвечающие за то, на каком трафике будут работать защитные модули. Например, если задать переменную Include /admin/*, то все защитный модули будут работать только с URL, которые содержат в адресе каталог admin.

Если наоборот, надо, чтобы защитные модули не работали на трафике содержащем определенное выражение, то соответствующий параметр надо добавить в поле *Exclude*.

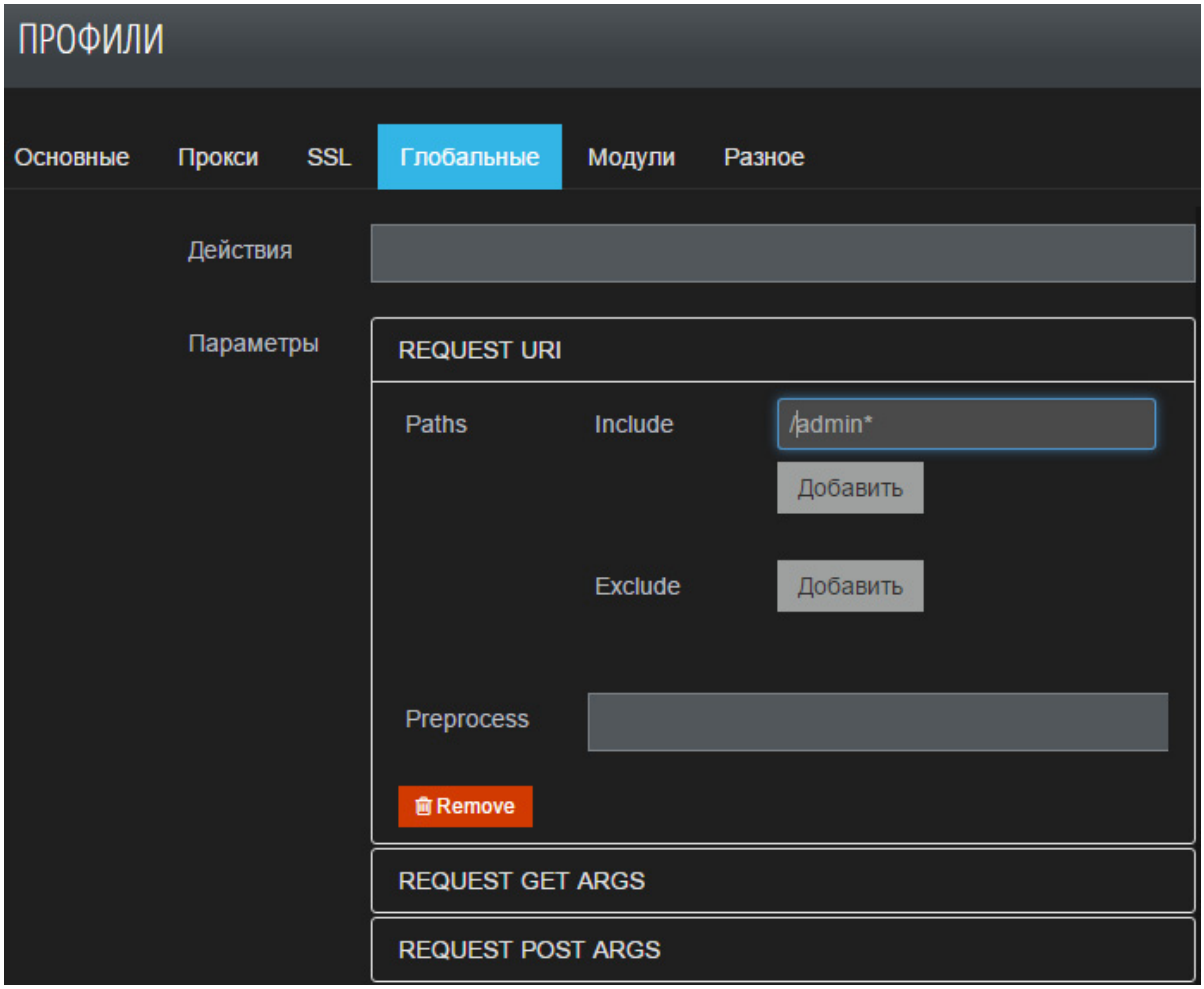


Рис. 68 – Глобальные значения параметров для защитных модулей

6.3.1.1.5. Модули

Обрабатываемый PT AF трафик проходит через набор модулей. В системе установлен следующий порядок срабатывания модулей:

- Защита HTTP (HTTP Protector),
- Модуль HMM (HMM Protector),
- Обнаружение CSRF (CSRF Protector),
- Защита от DDoS-атак (DDoS Protector),
- Обнаружение SQL-инъекций (SQL Injection Protector),
- Обнаружение XSS (XSS Protector),
- Обнаружение Open Redirect (Open Redirect Protector),
- Защита XML (XML Protector),
- ICAP-интеграция (ICAP Protector),
- Правила (Rule Engine),
- Content Security Policy (CSP Protector),
- Фильтрация ответов (Response Filter),
- Защита от роботов (Robot Protector),
- Правила доступа к ресурсам (ACL Protector),
- LDAP-авторизация (LDAP protector),
- Черные списки (Blacklist protector);
- Отслеживание сессий (Session Tracking).

При срабатывании защитного модуля могут генерироваться базовые события и вызываться различные действия (см. главу «Действия»). Некоторые действия направлены на журналирование трафика, отправку информации во внешние SIEM-системы (например, ArcSight) или выполнение модификации трафика. После срабатывания модуля с блокирующим правилом обработка запроса прекращается. Таким образом, действует логика «первое правило выигрывает».

Подробное описание модулей доступно в главе «Конфигурация модулей защиты».

6.3.1.1.6. Разное

Если перед PT AF расположен балансировщик нагрузки, который подменяет адрес источника на свой, в журналах PT AF адрес балансировщика будет все время попадать в качестве адреса источника трафика. Чтобы избежать такой ситуации, укажите заголовок HTTP, из которого PT AF будет брать адрес источника (балансировщик будет добавлять это поле в заголовки HTTP) и адрес самого балансировщика.

В поле *Получать реальный IP-адрес из HTTP-заголовка* (см. Рис. 69) на вкладке *Разное* указывается заголовок HTTP, из которого следует извлекать адрес источника трафика. Как правило, это заголовок X-Forwarded-For.

В поле *Доверять диапазону IP при получении адреса из заголовка* (см. Рис. 69) требуется указать адрес балансировщика. Адрес необходимо указывать с сетевой маской, например 192.0.2.123/32.

Включите опцию *Inject waf.js into HTTP responses*, чтобы активизировать защиту на стороне клиента от CSRF-, Clickjacking- и XSS-атак, а также активизировать обнаружение

ботов (PhantomJS, Selenium) и средств взлома (Acunetix, Burp Suite, ZAP и т.п.) путем внедрения в защищаемую страницу скрипта javascript.

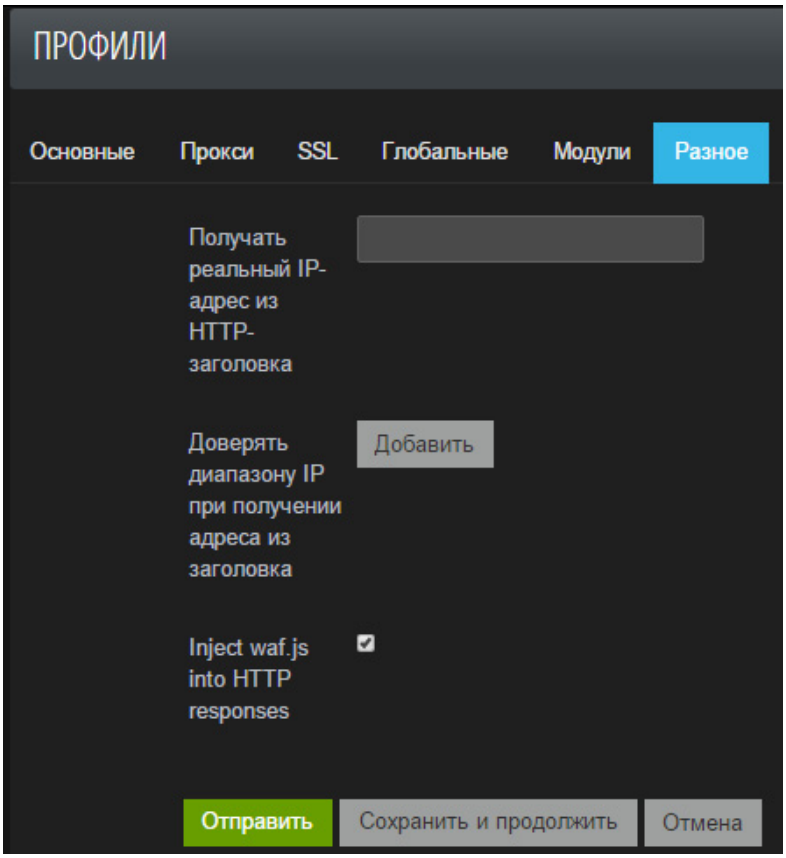


Рис. 69 – Обработка HTTP

6.3.1.2. Действия

На данной вкладке настраиваются действия, которые PT AF может производить с трафиком. Эти действия используются в модулях защиты, которые конфигурируются внутри защитного профиля.

Действия можно задать вручную, или использовать настроенные правила по умолчанию.

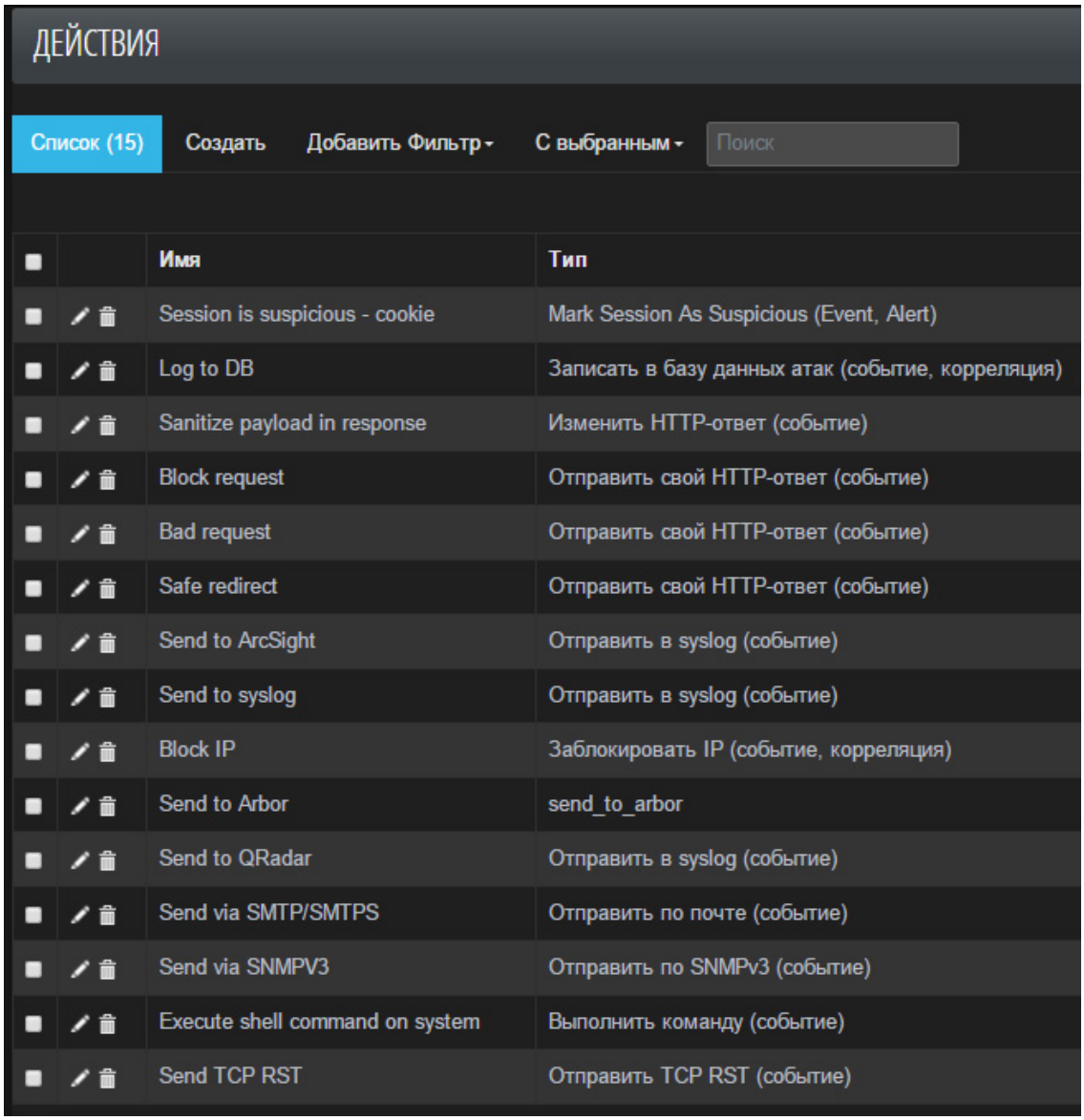


Рис. 70 – Действия

6.3.1.2.1. Session is suspicious - cookie

Назначив действие *Session is suspicious – cookie* для событий в модулях защиты, можно занести сессию в список подозрительных. Список подозрительных сессий находится во вкладке *Конфигурация -> Suspicious Sessions* (см. главу «Подозрительные сессии»).

Для данного действия предусмотрена следующая опция:

- Use whole cookie - определяет, какой тип будет присвоен сессии, при занесении ее в список подозрительных сессий. Если опция включена и в запросе содержатся Cookie, выданные защитным модулем *Отслеживание сессий*, то будет присвоен тип *Session*, в других случаях будет установлен тип *User*.

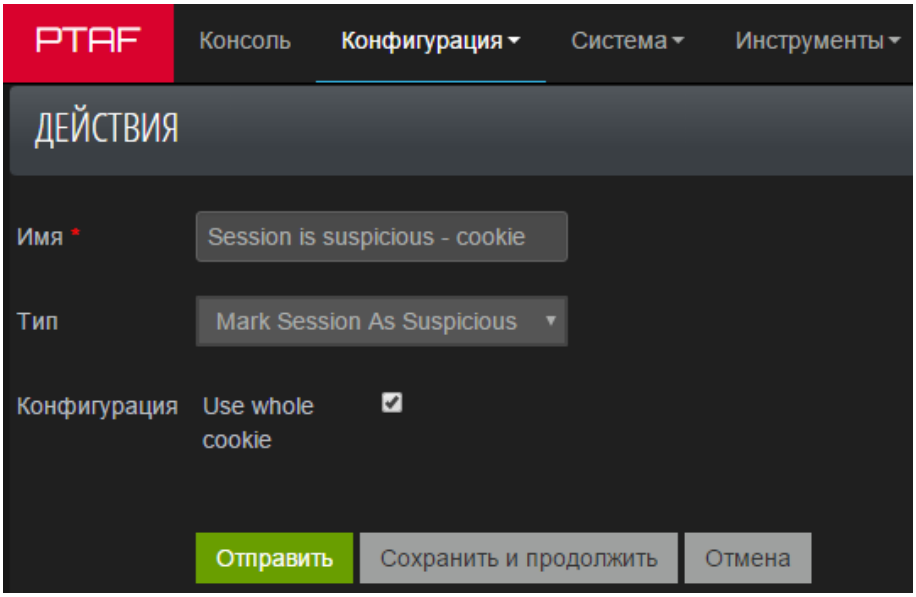


Рис. 71 – Session is suspicious – cookie

Примечание: сессию также можно заблокировать через консоль в панели атак, нажав кнопку *Блокировать* (см. главу «Панель Attacks»).

6.3.1.2.2. Log to DB

При помощи действия *Log To DB* обнаруженная атака будет занесена в базу данных Elasticsearch. Необходимо задать следующие поля:

- Хост – адрес узла с базой данных Elasticsearch (по умолчанию 127.0.0.1);

- Порт – номер порта, который слушает служба Elasticsearch (например, 9900);

- Log Response – опция для включения/отключения записи ответа в базу данных Elasticsearch;

- Max. Body Size – опция определяет размер тела ответа, который будет записан в базу данных Elasticsearch. Значение по умолчанию – 1 МБ.

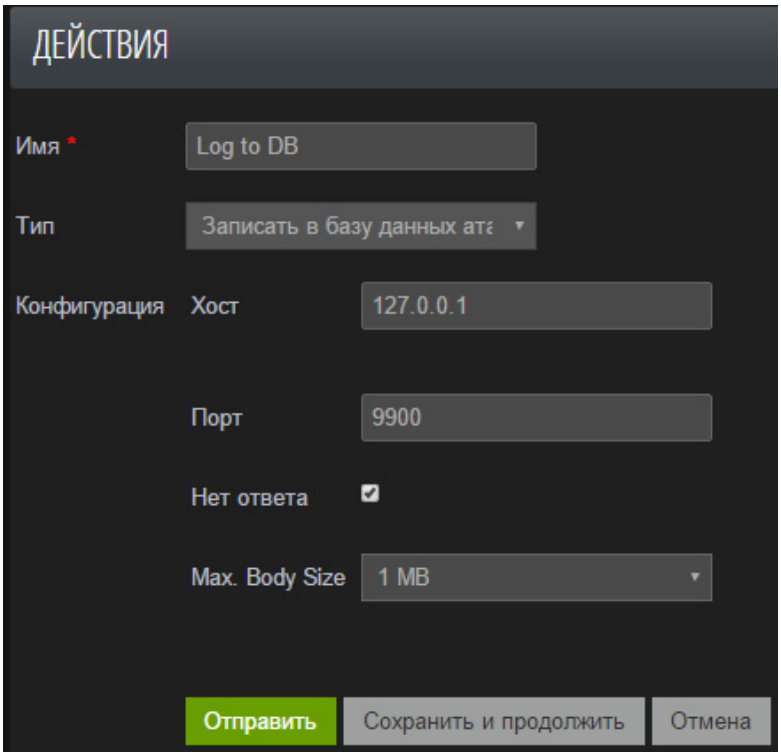


Рис. 72 – Log to DB

6.3.1.2.3. Sanitize payload in response

При срабатывании действия *Sanitize payload in response* входные данные, обнаруженные в ответе, будут экранированы или заменены заготовленной строкой. Задаются параметры:

- Тип – механизм проведения проверки: экранировать HTML-сущности или замена строкой;
- Строка для замены – заготовленный шаблон строки для замены.

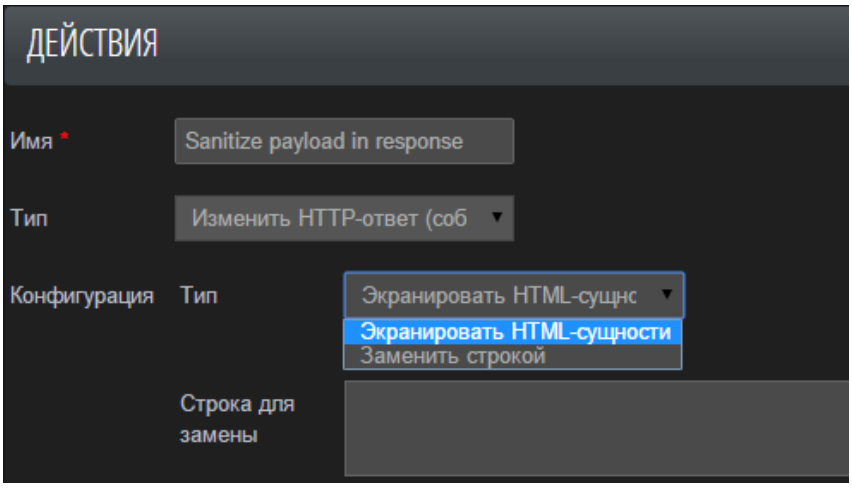


Рис. 73 – Sanitize payload in response

6.3.1.2.4. Block request

Действие *Block request* позволяет настроить шаблон страницы, которая будет отображаться при блокировании запросов. Указываются параметры:

- Код ответа – HTTP-код ответа (по умолчанию, 403);
- Заголовки ответа – заголовки HTTP-пакета, которые будут добавлены в ответ сервера;
- Тело ответа – текст HTML-страницы для отображения.

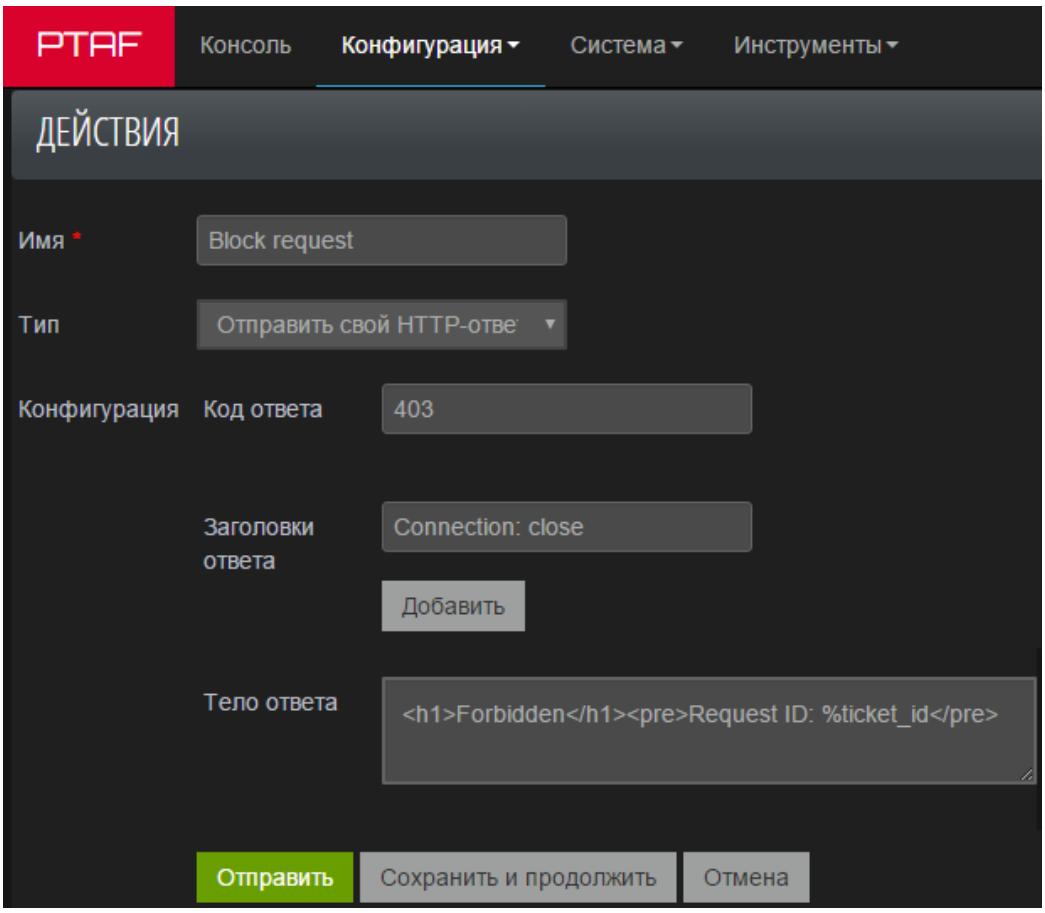


Рис. 74 – Block request

6.3.1.2.5. Bad request

Действие *Bad request*, подобно *Block request*, позволяет настроить шаблон страницы, которая будет отображаться при получении некорректно сформированных запросов. Указываются параметры:

- Код ответа – HTTP-код ответа (по умолчанию 400);
- Заголовки ответа – заголовки HTTP-пакета, которые будут добавлены в ответ сервера;
- Тело ответа – текст HTML-страницы для отображения.

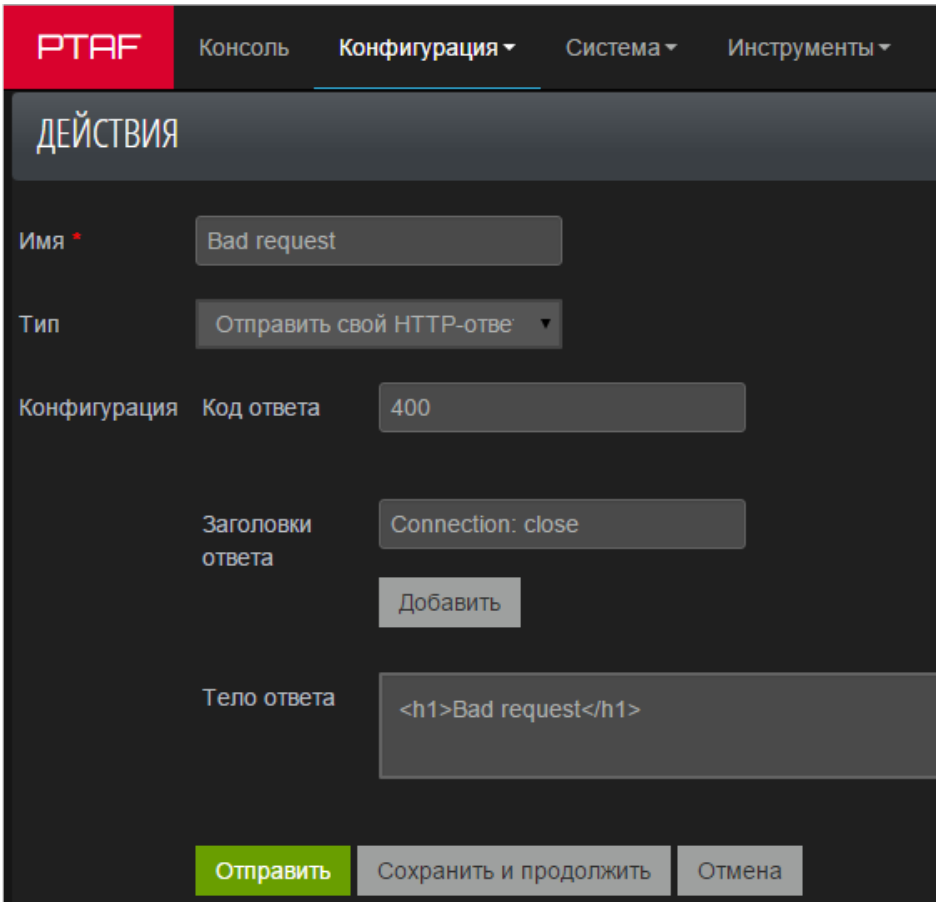


Рис. 75 – Bad request

6.3.1.2.6. Safe redirect

Действие *Safe redirect* предназначено для безопасного перенаправления клиента на заранее определенную страницу. Заполняются следующие поля:

- Код ответа – HTTP-код ответа (по умолчанию, 301);
- Заголовки ответа – заголовки HTTP-пакета, которые будут добавлены в ответ сервера. Здесь можно добавить заголовок Location, указывающий на разрешенную страницу для перенаправления;
- Тело ответа – текст HTML-страницы для отображения (поле можно оставить пустым).

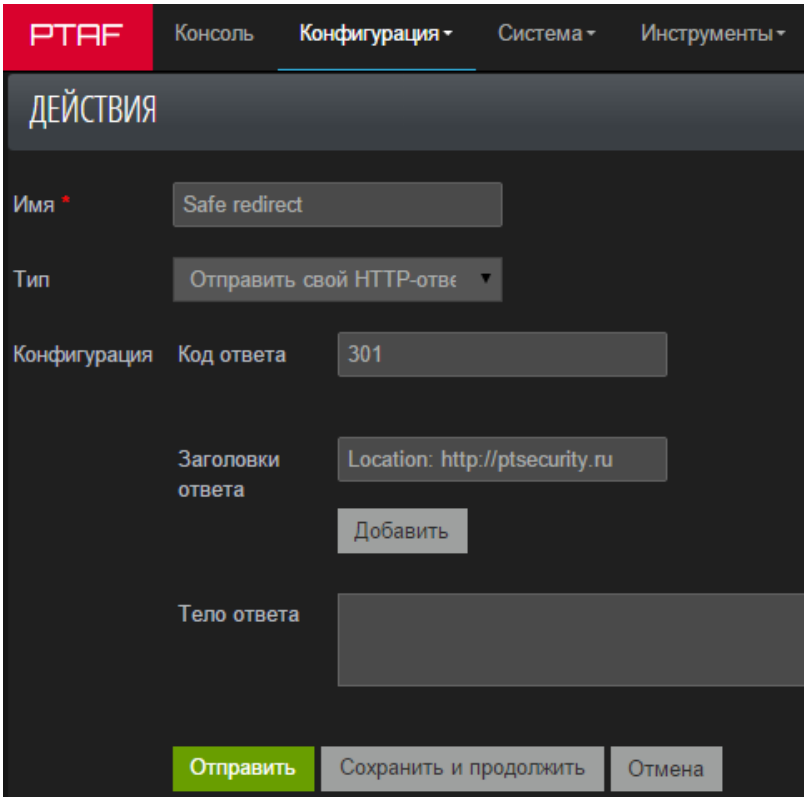


Рис. 76 – Safe redirect

6.3.1.2.7. Send to ArcSight, Send to syslog, Send to QRadar

Отправить информацию об инциденте во внешнюю систему журналирования. Заполняются следующие поля:

- Хост – узел, на котором установлен ArcSight;
- Порт – порт, на котором работает служба ArcSight (по умолчанию, 514);
- Формат – формат сообщения. Нажмите на значок *Помощь по формату*, чтобы увидеть перечень атрибутов события, которые можно поместить в текст сообщения. Например, чтобы в журнал попала запись с какого адреса шла атака, надо поместить в сообщение оператор %ip.
- Экранировать символы – список символов, которые будут экранироваться перед отправкой в журнал. Список всех доступных операторов доступен при нажатии на ссылку *Помощь по формату* в нижней части страницы.

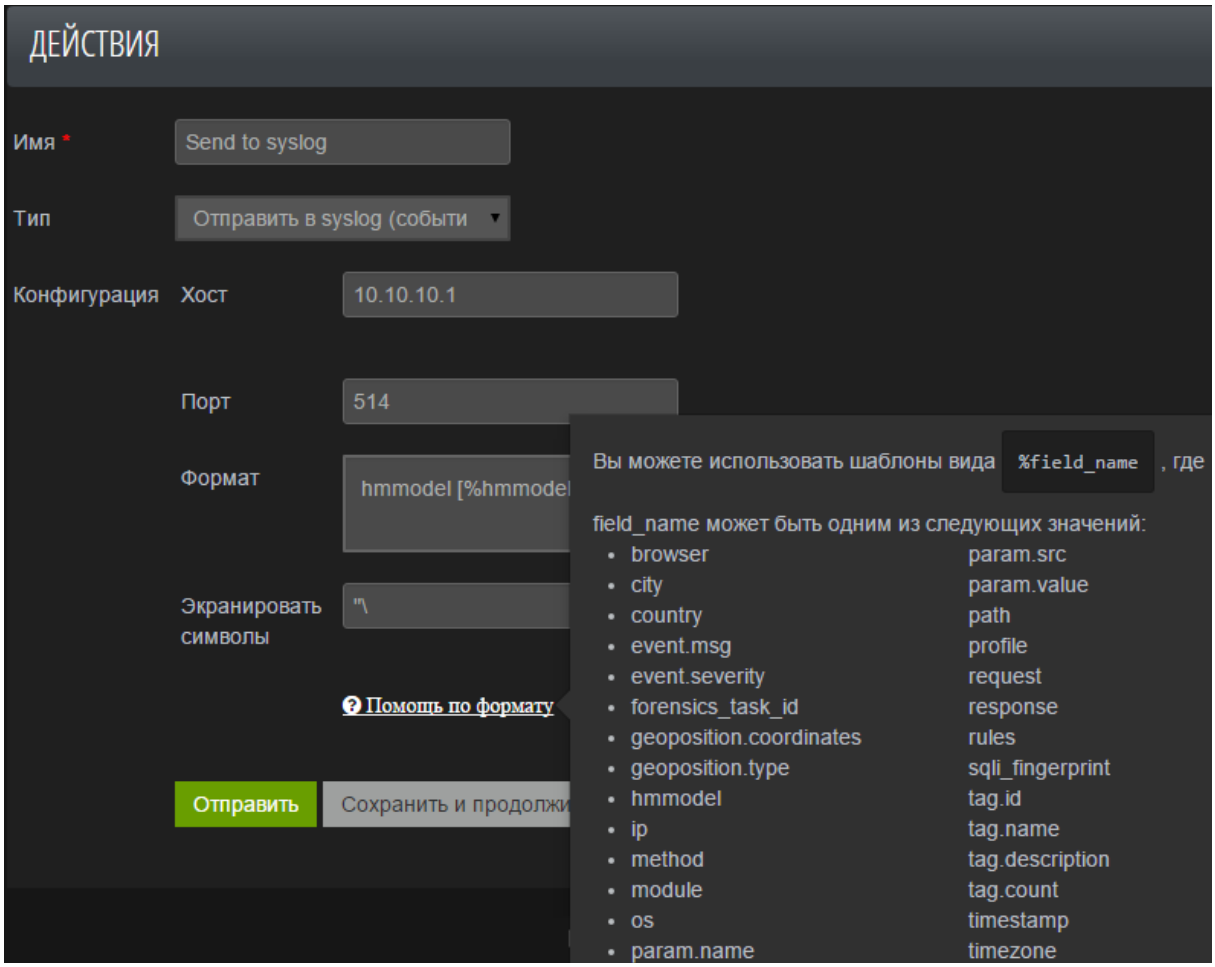


Рис. 77 – Send to syslog

6.3.1.2.8. Block IP

При срабатывании проверки блокируется IP-адрес атакующего. Укажите срок блокировки IP-адреса в поле *Длительность*.

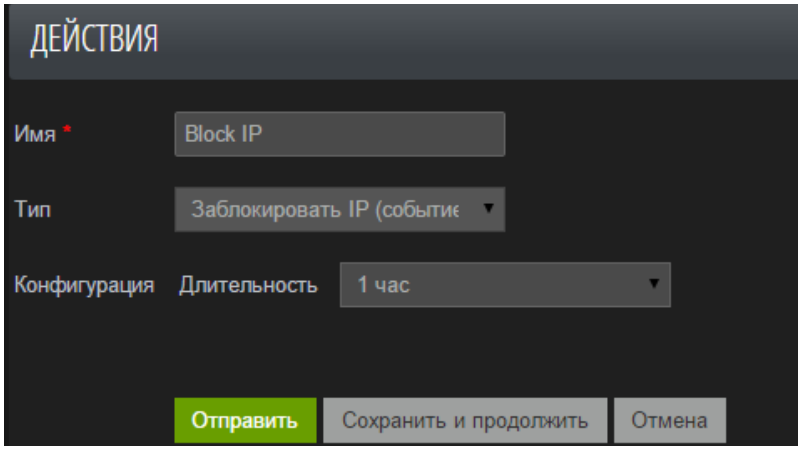


Рис. 78 – Block IP

6.3.1.2.9. Send to Arbor

Действие *Send to Arbor* обеспечивает интеграцию с Arbor Networks APS, позволяя заблокировать IP-адрес для Arbor. Продукт Arbor Networks APS предназначен для обнаружения DDoS-атак.

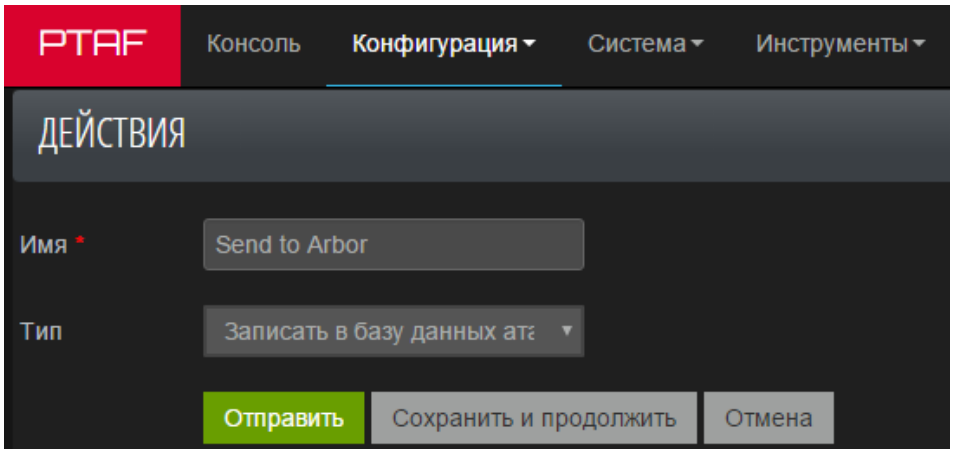


Рис. 79 – Send to Arbor

6.3.1.2.10. Send via SMTP/SMTPS

Отправить информацию об инциденте по электронной почте. Заполняются следующие группы полей:

- Хост SMTP, Порт SMTP – адрес и порт почтового сервера.
- Экранировать символы – список символов, которые будут экранироваться перед отправкой в журнал.
- Формат – Формат сообщения. При нажатии левой кнопкой мыши по значку справки можно увидеть перечень атрибутов события, которые можно поместить в текст сообщения. Например, чтобы в журнал попала запись с какого адреса шла атака, необходимо поместить в сообщение оператор %ip.
- To, From – кому и от чьего имени будет адресовано письмо.
- SASL – заполняется при условии, что почтовый сервер требует аутентификации при отправлении.
- SSL – заполняется при необходимости отправлять почту по шифрованному каналу.

6.3.1.2.11. Send via SNMPv3

Отправить информацию об инциденте на SNMP-сервер. Поля *Хост, Порт, Экранировать символы* и *Формат* имеют то же значение, что и для отправки значений во внешние журналы (см. главу «Send to ArcSight, Send to syslog, Send to QRadar» и «Send via SMTP/SMTPS»). Остальные поля относятся к специфическим параметрам аутентификации на SNMP-сервере.

6.3.1.2.12. Execute shell command on system

Действие позволяет выполнить пользовательскую команду на PT AF. Это действие можно использовать, например, если стоит задача записать активность злоумышленника в дамп-файл формата pcap. Или для любых других целей. Однако, это действие является столь же гибким, сколь и потенциально опасным.

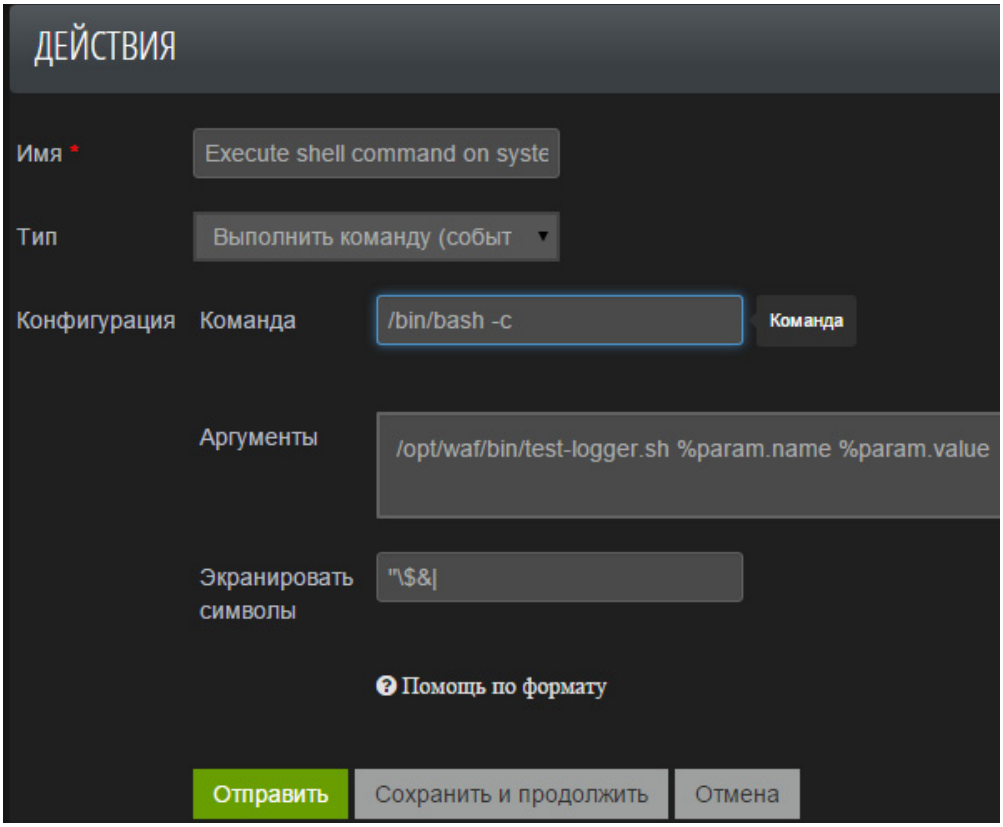


Рис. 80 – Действие «Execute shell command on system»

6.3.1.2.13. Send TCP RST

Это действие предназначено для отправки TCP-пакетов с флагом RST в режиме работы *Active Prevention* (необходимое условие: трафик пришел со SPAN-порта и в настройках *Снифф*ер выбран интерфейс для отправки TCP RST). Не требует настройки.

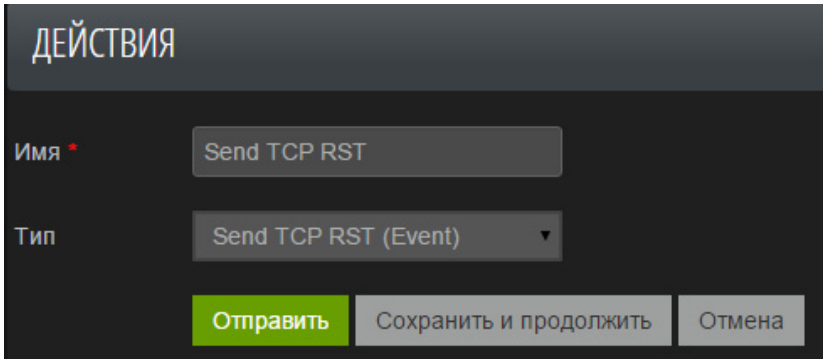


Рис. 81 – Действие «Send TCP RST»

6.3.1.2.14. Сканирование PT WebEngine

Действие для такого типа по умолчанию не создается. Оно позволяет при обнаружении атаки немедленно просканировать приложение встроенным сканером с целью узнать, насколько данная атака актуальна для приложения, т.е. уязвимо ли оно к атаке. Его можно создать, нажав на кнопку *Создать* и выбрав из списка типов действий соответствующий пункт.

> **Внимание!** Не указывайте в качестве адреса реальный backend – таким образом вы сможете только усилить воздействие злоумышленника на вашу систему. Используйте для проверки уязвимости только сервера, расположенные в тестовой зоне.

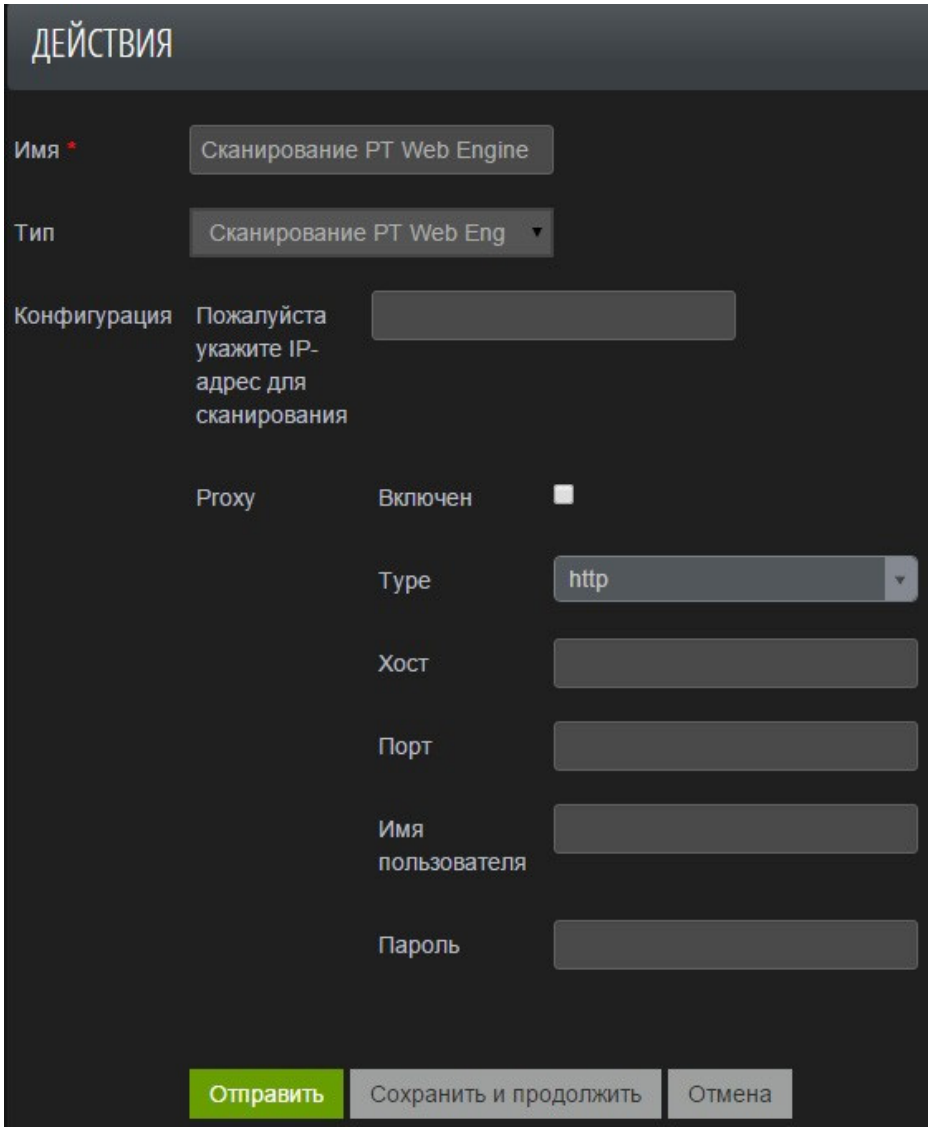


Рис. 82 – Сканирование PT Web Engine

6.3.1.3. Правила

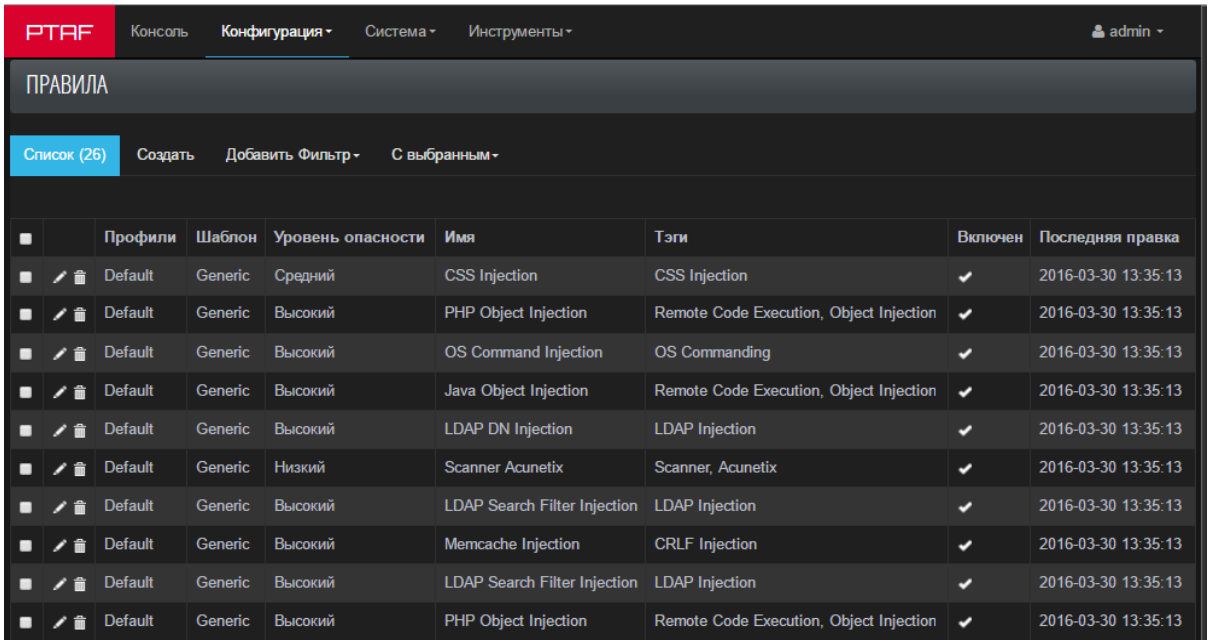
На данной вкладке представлен список правил, используемых в модуле защиты *Правила*.

Для каждого правила в списке указаны следующие параметры:

- Профили – имена профилей, для которых работает правило;
- Шаблон – применимость правила для новых профилей. Значение *Generic* указывает на то, что правило является системным, а значит будет применимо ко всем новым профилям при их создании;
- Уровень опасности – уровень важности правила;
- Имя – имя правила;
- Теги – один или несколько тегов (см. главу «Теги»), соответствующих правилу. В результате срабатывания правила генерируется экземпляр базового события с заполненными значениями полей, включая поля тега;
- Включен – статус активности правила.

Каждое правило содержит фильтры, аналогичные глобальным настройкам профиля, которые позволяют ограничить действия правила определенной областью передаваемого запроса или ответа.

Внимание! Все правила, включенные для группы серверов, применяются по очереди.



Профили	Шаблон	Уровень опасности	Имя	Тэги	Включен	Последняя правка
Default	Generic	Средний	CSS Injection	CSS Injection	✔	2016-03-30 13:35:13
Default	Generic	Высокий	PHP Object Injection	Remote Code Execution, Object Injection	✔	2016-03-30 13:35:13
Default	Generic	Высокий	OS Command Injection	OS Commanding	✔	2016-03-30 13:35:13
Default	Generic	Высокий	Java Object Injection	Remote Code Execution, Object Injection	✔	2016-03-30 13:35:13
Default	Generic	Высокий	LDAP DN Injection	LDAP Injection	✔	2016-03-30 13:35:13
Default	Generic	Низкий	Scanner Acunetix	Scanner, Acunetix	✔	2016-03-30 13:35:13
Default	Generic	Высокий	LDAP Search Filter Injection	LDAP Injection	✔	2016-03-30 13:35:13
Default	Generic	Высокий	Memcache Injection	CRLF Injection	✔	2016-03-30 13:35:13
Default	Generic	Высокий	LDAP Search Filter Injection	LDAP Injection	✔	2016-03-30 13:35:13
Default	Generic	Высокий	PHP Object Injection	Remote Code Execution, Object Injection	✔	2016-03-30 13:35:13

Рис. 83 – Список правил

6.3.1.3.1. Создание и редактирование правила

Нажмите кнопку *Создать*, чтобы настроить новое правило, нажмите кнопку *Редактировать*, чтобы изменить значение настроек.

Список настроек следующий (Рис. 84):

- Включен – опция активации правила;
- Шаблон – нередактируемая опция, указывающая на применимость правила для новых профилей. Значение *Пользовательская настройка* указывает на то, что

правило не является системным, а значит не будет применимо ко всем новым профилям при их создании;

- Имя – имя правила;
- Описание – краткая информация о правиле;
- Правила – выражения-операнды, из которых состоит правило;
- Тест – опция для тестирования правила;
- Теги – один или несколько тегов (см. главу «Теги»), соответствующих правилу;
- Действия – одно или несколько действий (см. главу «Действия»), соответствующих правилу;
- Уровень – уровень важности правила;
- Профили – имена профилей, для которых работает правило.

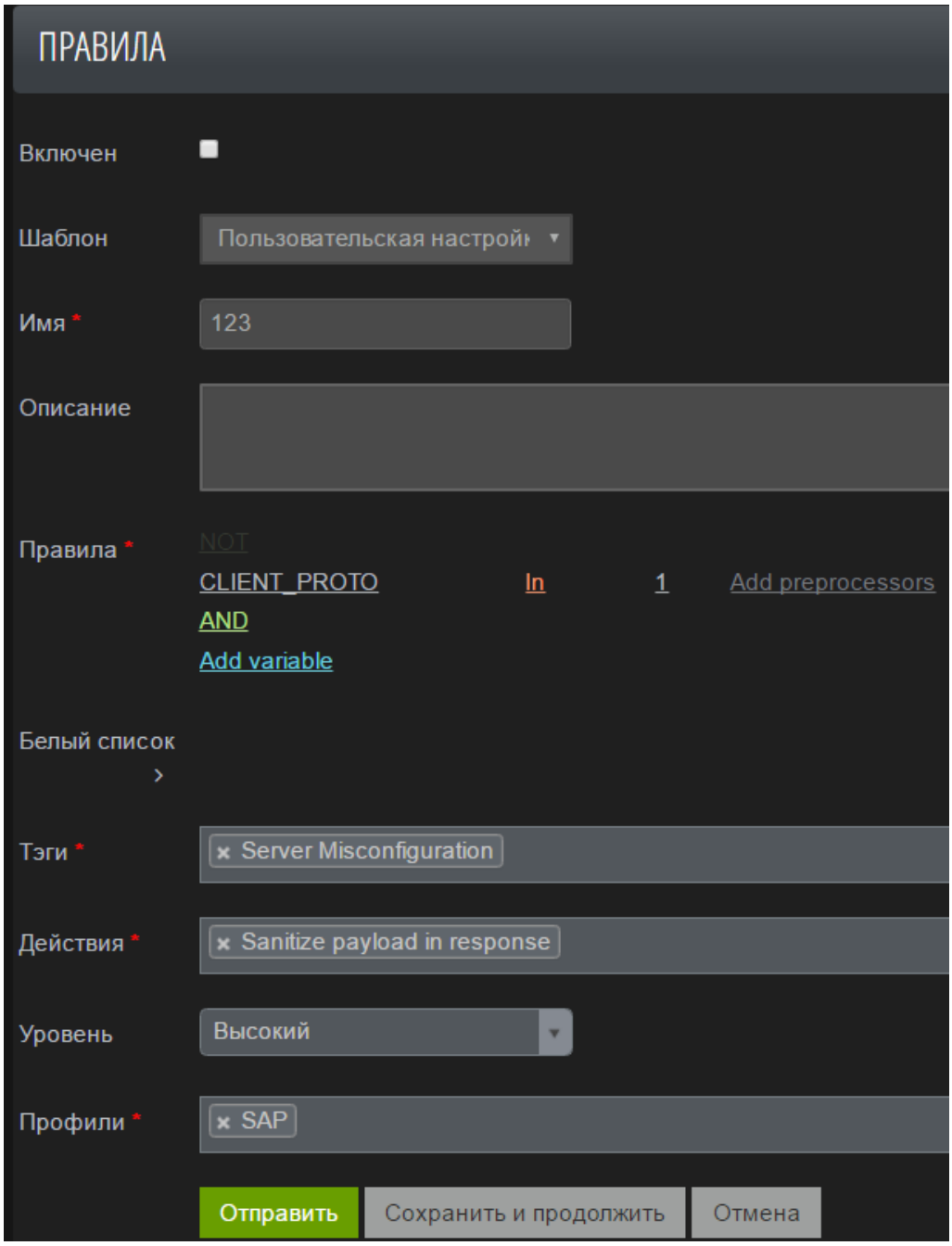


Рис. 84 – Окно редактирования правила

6.3.1.3.2. Задание выражений правила

Правило состоит из выражений-операндов, объединенных логическими операторами AND или OR.

В общем случае выражение создается заданием переменной и оператора сравнения, которые выбираются из списка, значением переменной для сравнения (текстовое поле) и списком препроцессоров (обязательное для строковых переменных, поле множественного выбора), которыми обрабатывается значение переменной запроса-ответа перед сравнением со значением переменной из выражения.

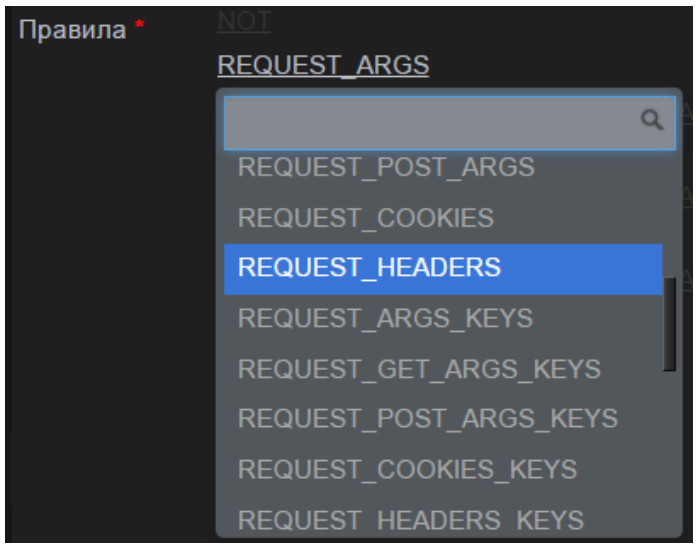


Рис. 85 – Список переменных

После заполнения очередного обязательного поля выражения становится доступным следующее поле выражения. К обязательным полям относятся поля *Add operator* и *Add value*.

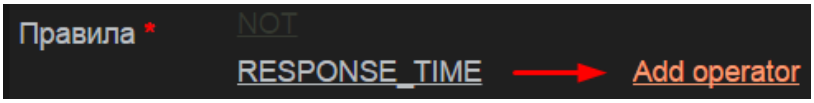


Рис. 86 – Поля выражения

После заполнения всех обязательных полей выражения становится доступным создание нового выражения правила (действие *Add variable*).

Рис. 87 – Окно с заполненными полями

Значения в поле множественного выбора препроцессоров можно удалять и перемещать, меняя тем самым порядок действия препроцессоров - от первого слева до последнего справа.

Примечание: для числовых переменных препроцессоры не выбираются.

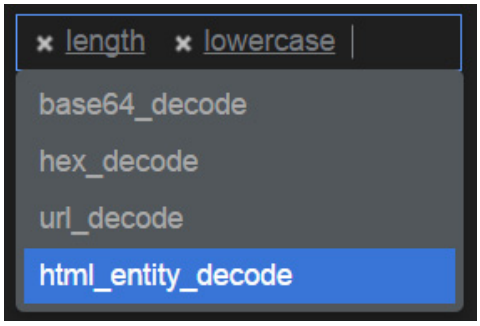


Рис. 88 – Выбор препроцессоров

Для списковых переменных выражение формируется заданием вложенного правила на параметры этой переменной. Отличиями вложенного правила от основного являются: необязательное текстовое поле для задания имени параметра вместо обязательного выбора переменной и дополнительное булевое поле-переключатель ("Aa") игнорирования регистра. Переключатель необходим, чтобы игнорировать регистр имени параметра переменной при его сравнении с выражением правила. Т.е., например, выражение правила ["param" "Aa" Equals "value"] будет удовлетворять значению value соответствующей переменной и для имени параметра param и для имени параметра PARAM.

Если текстовое поле имени параметра не заполнено, то при заполнении других обязательных полей на месте имени параметра появляется звездочка. Это означает, что данное выражение проверяется для всех параметров. Наличие звездочки не исключает возможности в любой момент ввести конкретное имя параметра в это поле.

Логические операторы в выражении меняются нажатием левой кнопкой мыши на сам оператор. При нажатии на AND он переключается на OR и наоборот. Переключение происходит в соответствии со следующей логикой: при нажатии левой кнопкой мыши на операторе выражения "слева" (сверху) и "справа" (снизу) объединяются в составные операнды и к ним применяется оператор обратный тому, на который было совершено нажатие.

Для отрицания выражения или правила служит оператор NOT. Оператор NOT, размещенный сверху всего правила (Рис. 89), предназначен для отрицания всего правила. Нажмите левой кнопкой мыши на данный оператор, чтобы активировать его. Отрицание простого выражения осуществляется с панели управления выражением (см. следующую главу – «Панель управления выражениями»).

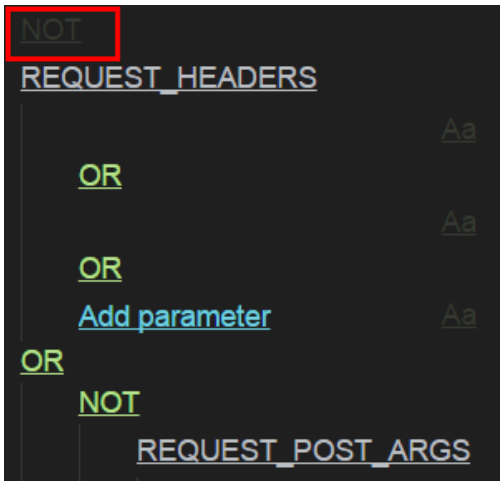


Рис. 89 – Оператор NOT

6.3.1.3.3. Панель управления выражениями

При наведении курсора мыши на выражение оно подсвечивается и справа появляется панель с кнопками управления выражением (см. Табл. 7).

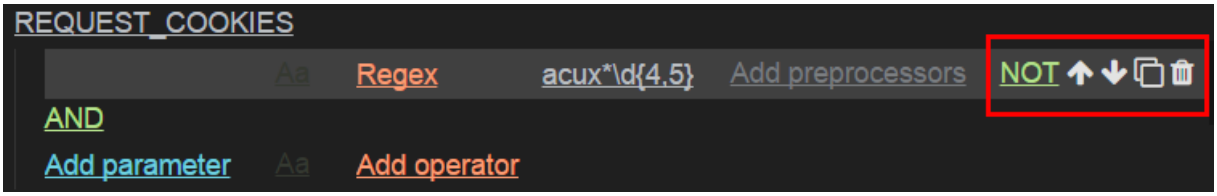


Рис. 90 – Панель управления выражениями

Для вложенных правил списковых переменных доступны аналогичные действия. При сохранении или тестировании (см. главу «Тестирование правила») происходит валидация полей. Если есть невалидные поля, то напротив них появляется всплывающая подсказка, и правило не сохраняется и не тестируется.

Таблица 7. Команды управления выражениями

Команда	Описание
NOT	Осуществляет отрицание единичного простого выражения
Move up/ down	Перемещает выражение вверх/вниз на 1 шаг. При перемещении выражения в направлении составного операнда перемещаемое выражение передвигается в составной операнд. При перемещении выражения в направлении из составного операнда выражение выходит из составного операнда
Remove	Удаляет выражение из правила
Group	Инициализирует режим группировки выражений[3]
Ungroup	Выносит выражение из составного операнда подобно тому, как работают действия перемещения на границах составных операндов, только позволяет вынести выражение, находящееся в середине составного операнда

Таблица 7. Команды управления выражениями

Команда	Описание
Clone	Создает полную копию выражения.

6.3.1.3.4. Тестирование правила

Тестирование правила происходит на наборе переменных и их значений из запросов и ответов так, как это делает PT AF при реальных запросах и ответах.

Для раскрытия области тестирования необходимо нажать на метку *Test* в окне создания/редактирования правила.

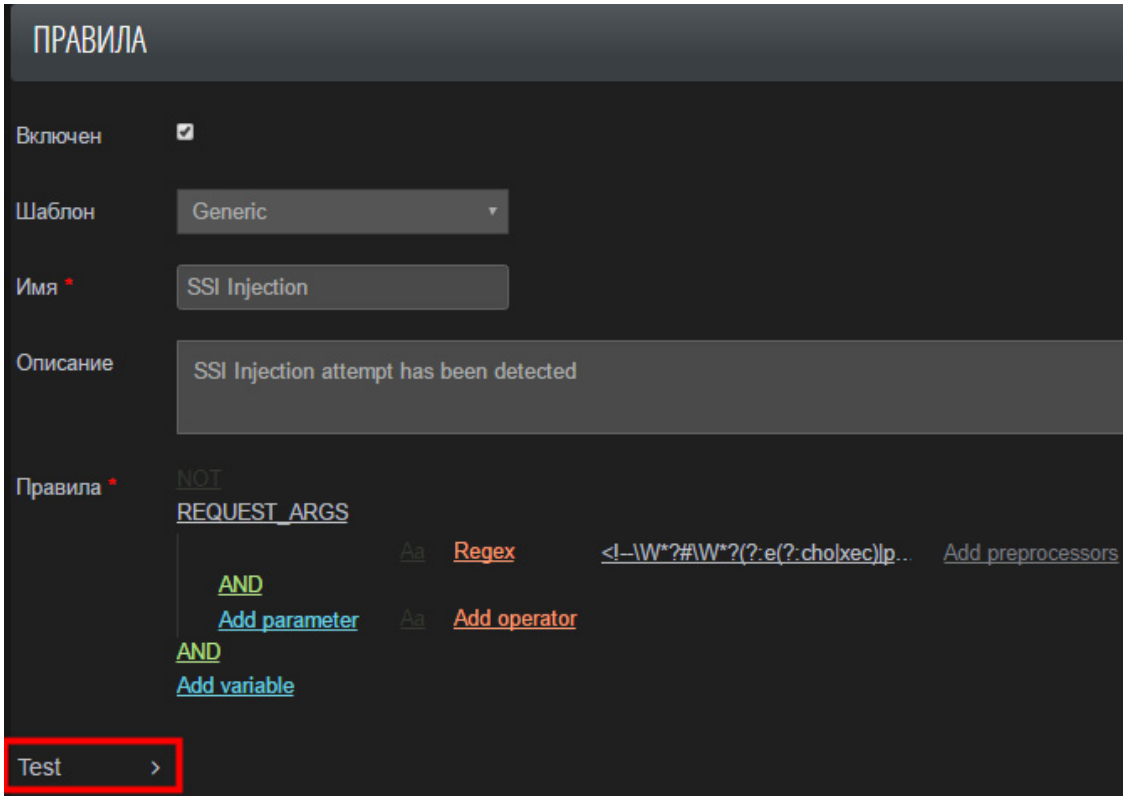


Рис. 91 – Окно редактирования правила

Область тестирования состоит из редактора переменных (*Variables*), их значений, на которых тестируется правило, и полей для ввода запросов и ответов (*Raw request* и *Raw response*), которые можно разобрать на переменные для упрощения тестирования.

3. В режиме группировки сплошной границей обозначаются члены нового составного операнда, а пунктирной границей - выражения, которые можно добавить в группу. Для добавления в группу выражения или для удаления выражения из группы нажмите на нем левой кнопкой мыши. Нажатие левой кнопкой мыши в любом другом месте страницы завершает группировку. Выражения в группе объединяются оператором обратным тому, которым объединялись выражения до группировки.

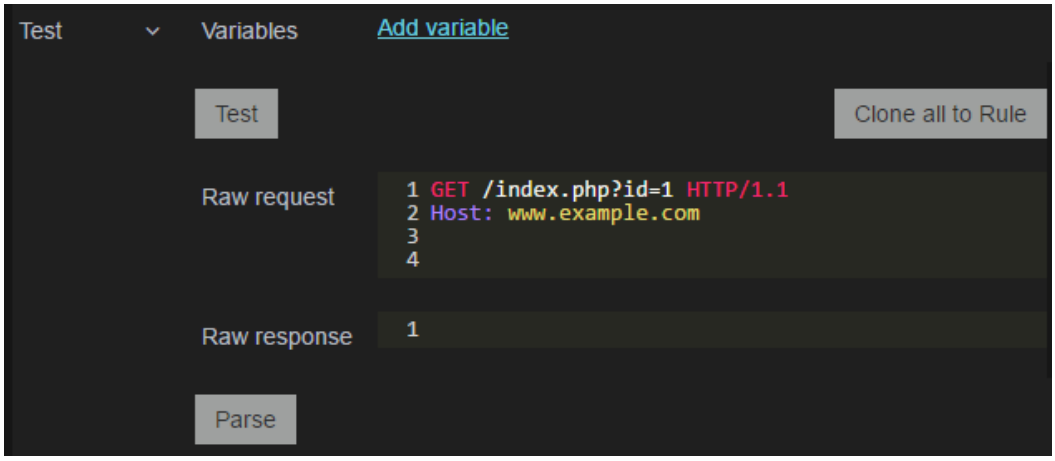


Рис. 92 – Область тестирования

Тестирование на переменных

Для тестирования правила необходимо выбрать в редакторе переменных *Variables* переменные и их значения, на которых будет тестироваться правило, а затем нажать кнопку *Test*.

Внимание!

- Правило тестируется на переменных, а не на запросах и ответах. Разбор запросов и ответов предназначен только для заполнения переменных, запрос с ответом никак не участвуют в процессе самого тестирования.
- Все переменные должны иметь нормализованные, раскодированные значения.

Если правило удовлетворяет переменным, то рядом с кнопкой *Test* появляется сообщение о том, что правило блокирует потенциальные запрос и ответ, состоящие из введенных переменных. Если правило не удовлетворяет переменным, то рядом с кнопкой *Test* появляется сообщение о том, что правило не блокирует потенциальные запрос и ответ, состоящие из введенных переменных. При наведении курсора мыши на строку с переменной и ее значением в редакторе переменных строка подсвечивается, справа появляется панель с кнопками управления выражением. Действие *Clone to Rule* копирует переменную и ее значение в правило, подставляя оператор *Equals*. Действие *Remove* удаляет переменную и ее значение из набора переменных.

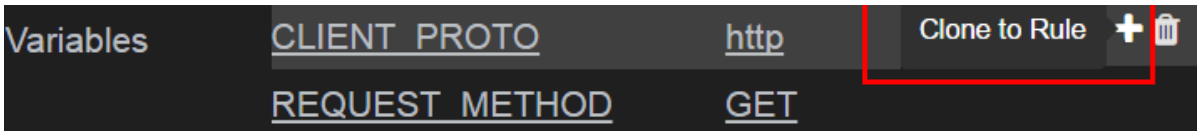


Рис. 93 – Кнопка Clone to rule

Для автоматического заполнения переменных из запросов и ответов можно заполнить поле *Raw request* запросом, а поле *Raw response* ответом, и нажать кнопку *Parse*.

Внимание! Обязательно должно быть заполнено одно из двух полей.

После обработки введенных запроса и ответа редактор переменных заполняется переменными и их значениями из запроса и ответа. На вводимые запросы и ответы накладываются ограничения. Если запрос и/или ответ не удовлетворяет ограничениям, то рядом с кнопкой *Parse* появляется соответствующее сообщение.

Примечание. Ограничением на запрос является обязательное наличие заголовка Host. Ограничением на ответ является обязательное наличие заголовка Content-Length с корректным значением. Если в запросе есть тело, то заголовок Content-Length с корректным значением является также обязательным для запроса.

6.3.1.4. Теги

На вкладке *Теги* формируются сущности, называемые тегами и представляющие собой «маячки» для работы механизмов формирования событий (*Events*) и корреляций (*Alerts*). Теги проставляются для Базовых событий, генерируемых в результате срабатывания правил и защитных модулей.

Теги имеют уникальные идентификаторы (автогенерируемые), а также имя, тип и описание, которые доступны при описании событий и корреляций.

Предусмотрены следующие типы тегов:

- acl – событие в приложении, связанное с контролем доступа (например, Guest access, Unauthorized Access, Authorization Failed и др.);
- tool – применение программного средства (например, сканирование сканером Nessus);
- weakness – обнаружение уязвимости;
- attack – осуществление атаки (например, атака ShellShock).

6.3.1.5. HMM-модели

На вкладке *HMM-модели* приведен список скрытых моделей Маркова (Hidden Markov Models), используемых в модуле защиты *Модуль HMM*. Модели создаются автоматически во время анализа действий пользователей.

Для каждой модели в списке указаны следующие параметры:

- Профиль - имя профиля, для которого создана модель;
- Путь – путь на защищаемом приложении для указанного профиля;
- Параметр – параметр указанного пути. Если путь не указан, то поле параметра будет пустым;
- Источник – канал для получения данных;
- Порог – атрибут модели;
- Количество образцов данных – общее количество собранных образцов данных на этапе обучения модели;
- Количество ошибок – общее количество ошибок на этапе обучения;
- Последняя правка – дата и время последнего изменения модели на этапе ее обучения;
- Включен – атрибут модели, который определяется настройками тренера и действием пользователя.

6.3.1.6. Content Security Policy

Content Security Policy – новый механизм защиты от XSS на стороне браузера. Он позволяет указать белый список внешних ресурсов, с которых может загружаться тот или иной контент. Если веб-сервер не умеет добавлять CSP-заголовки, их может добавить PT AF.

На вкладке *Content Security Policy* конфигурируются политики CSP, используемые в модуле защиты *Content Security Policy*. Данный модуль защиты включается внутри защитных профилей и использует конфигурации, создаваемые на этой вкладке для соответствующего профиля. CSP настраивается для страниц индивидуально и позволяет указывать допустимые источники для различных внедряемых объектов.

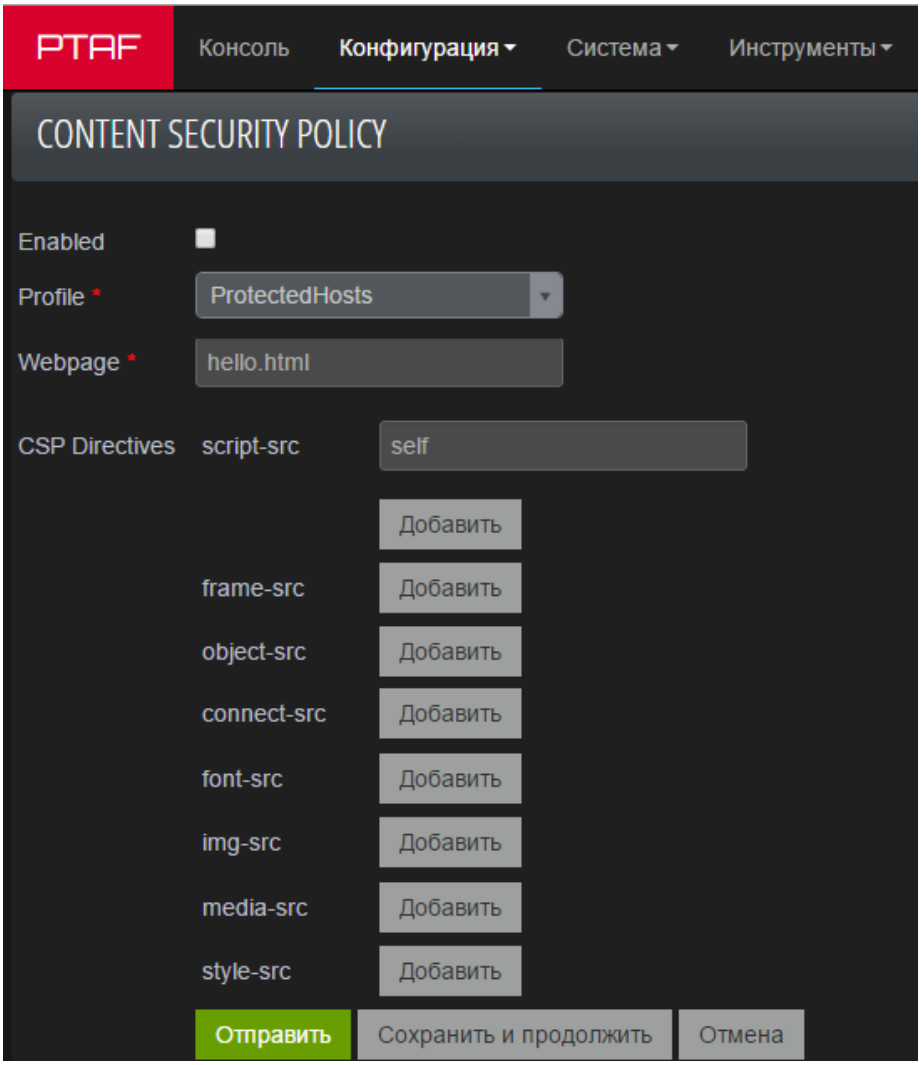


Рис. 94 – Content Security Policy

6.3.1.7. Черный список IP-адресов

На данной вкладке представлен список IP-адресов, которые следует заблокировать при обращении к приложению. Таким образом можно ограничить доступ из TOR-сетей, различных анонимайзеров и пр.

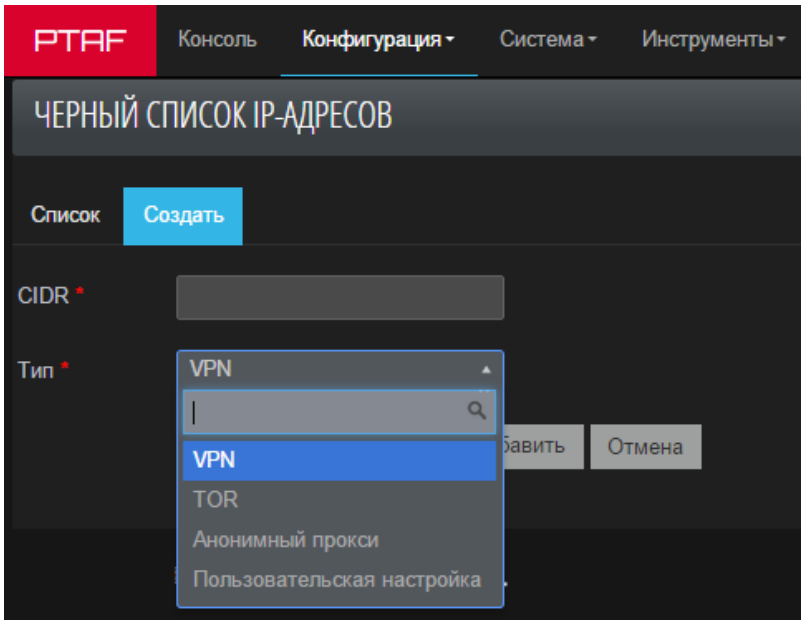


Рис. 95 – Блокировка IP-адреса

6.3.1.8. Черный список хостов

Во вкладке *Черный список IP-адресов -> Черный список хостов* размещен список доменов, которые следует заблокировать (фишинговые, вирусные и пр.). Поиск происходит при помощи регулярных выражений. Программа сообщает приходит ли пользователь с подобных сайтов, на которых скорее всего он был атакован.

6.3.1.9. XML-схемы

Настройка, позволяющая загружать пользовательские XSD-схемы, которые проверяются в модуле защиты *Защита XML*.

6.3.1.10. Подпись форм

Данный механизм предназначен для обеспечения подлинности веб-форм защищаемого приложения.

Механизм защиты веб-форм состоит в следующем: в соответствии с заданной политикой безопасности по защищаемой форме и секретному ключу вычисляется подпись (проверочное значение), которая добавляется в эту форму. При получении от клиента данных формы в виде HTTP-запроса производится повторное вычисление значения подписи по параметрам запроса и секретному ключу и его сравнение с ранее вычисленным оригинальным значением подписи. Если данные значения совпадают, то запрос разрешается. Отсутствие подписи в HTTP-запросе или ее несовпадение с оригинальной подписью приводит к генерации события безопасности.

Данный механизм может быть использован для защиты:

1. Веб-форм, сгенерированных на серверной стороне приложения и передающихся методом POST (атрибут формы method равен POST);

2. Данных в текстовых полях (поля <input> с атрибутом type равным text или поля <textarea>)) или в скрытых полях (поля <input> с атрибутом type равным hidden) веб-формы.

Данный механизм не может быть использован для защиты:

1. Веб-форм, динамически генерируемых на стороне клиента с средствами JavaScript;
2. Веб-форм, отправляемых на сервер методом GET;
3. HTTP-запросов, создаваемых на стороне клиента средствами JavaScript.

Порядок работы механизма защиты веб-форм следующий:

1. Генерация секретного ключа для вычисления подписи;
2. Создание конфигурации на основе настроек механизма защиты;
3. Для каждой HTML-страницы, сгенерированной защищаемым веб-сервером в ответ на запрос пользователя, происходит анализ HTML-кода и при наличии в нем защищаемой веб-формы выполняется вычисление подписи в соответствии с заданной конфигурацией.
4. Подпись добавляется в скрытое (hidden) поле защищаемой формы;
5. При получении от пользователя HTTP-запроса, соответствующего защищаемой форме, выполняется проверка наличия подписи в запросе, вычисление подписи по параметрам запроса и ее сравнение с оригинальным значением;
6. Если подпись в запросе отсутствует или не совпадает с оригинальной, то в зависимости от конфигурации генерируется событие безопасности.

6.3.1.10.1. Политики

Нажмите кнопку *Создать*, чтобы выбрать и настроить тип политики безопасности для генерации подписи. Настройте следующие параметры:

- Protect - параметр позволяет выбрать тип политики безопасности. Возможные значения:
  - Hidden Fields in All Forms – для всех форм подписываются значения всех скрытых полей. Политика применяется для всех URL;
  - Fields Names in Specified Forms – подписываются все имена полей формы, без учета их значений. Защищаемые формы задаются по URL;
  - Fields in All Forms – для всех форм подписываются значения заданных полей. Политика применяется для всех URL;
  - Fields in Specified Forms – для веб-форм, заданных по URL, подписываются значения заданных полей, с возможностью проверки значения:
    - на соответствие регулярному выражению;
    - на равенство значению поля в оригинальной форме от защищаемого веб-сервера.
- URL - адрес обработчика формы;
- Use session ID – опция, позволяющая включить идентификатор сессии пользователя в вычисление подписи;
- Use HTTP Method – опция, позволяющая включить HTTP-метод формы в вычисление подписи;
- Имя - имя пользовательской политики безопасности;
- Параметр.

- Field name;
- Действие. Возможные значения:
  - Control – проверять равенство значения поля в веб-форме и соответствующего значения параметра в HTTP-запросе.
  - Validate – проверять соответствие значения параметра в HTTP-запросе регулярному выражению. Если выбрано значение *Validate*, то дополнительно необходимо выбрать тип регулярного выражения (см. главу «Регулярные выражения»).

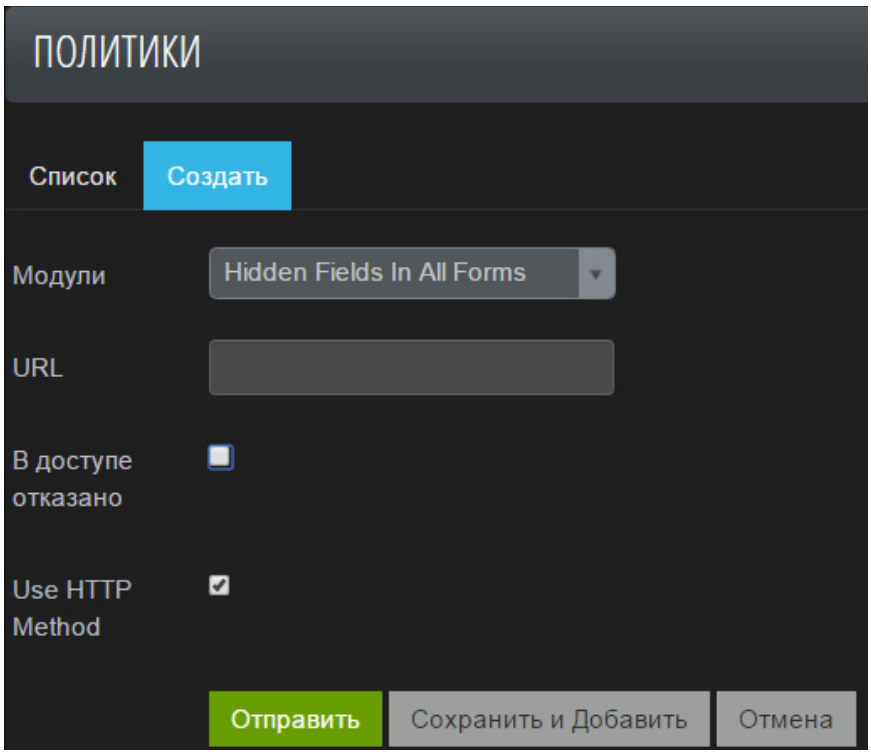


Рис. 96 – Окно настройки политики

6.3.1.10.2. Регулярные выражения

В политиках типа *Fields in Specified Forms* и *Fields in All Forms* (см. главу «Политики») можно проверять значения полей на соответствие регулярному выражению. Данные, несоответствующие регулярному выражению в HTTP-запросе, приведут к невалидной подписи.

Чтобы создать запись регулярного выражения, необходимо нажать соответствующую кнопку на вкладке *Регулярные выражения*, а затем установить имя и задать регулярное выражение.

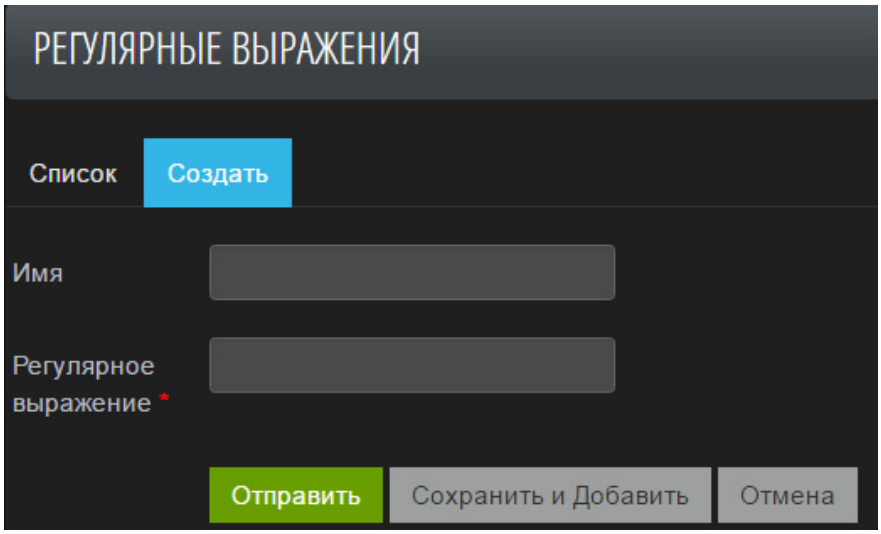


Рис. 97 – Создание регулярного выражения

6.3.2. Сеть

К разделу *Сеть* относятся следующие вкладки:

- Шлюзы;
- Группа серверов;
- Файрвол;
- Сниффер;
- Алиасы сетевых интерфейсов;
- Виртуальные IP;

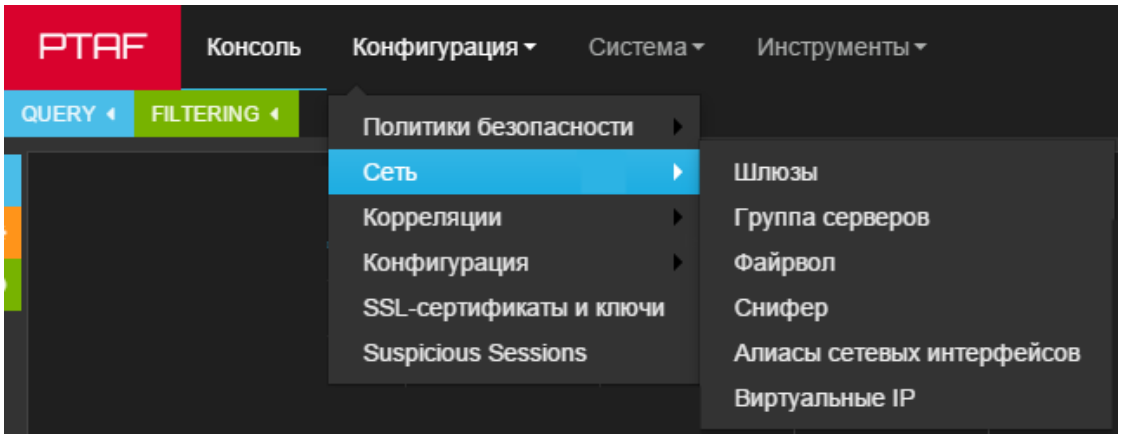


Рис. 98 – Сеть

6.3.2.1. Шлюзы

Данная вкладка предназначена для просмотра конфигурации сетевых настроек PT AF и привязки алиасов сетевых интерфейсов. Шлюз создается на этапе установки системы при помощи утилиты wsc. Нажмите кнопку редактировать в строке шлюза, чтобы изменить настройки.

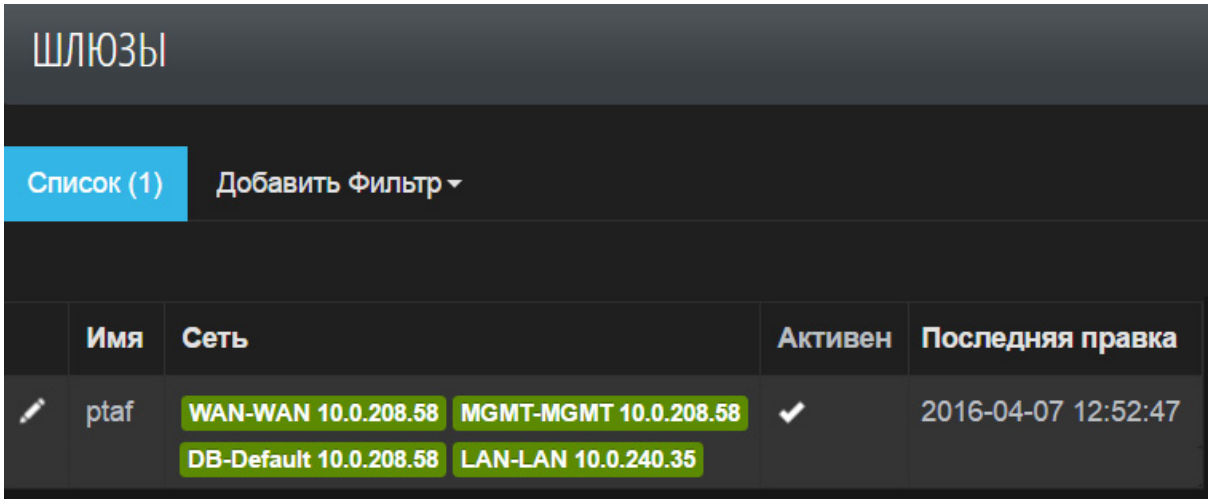


Рис. 99 – Шлюзы

6.3.2.1.1. Основные опции

На данной вкладке в поле *Имя* указан текущий шлюз. Опция *Активен* служит для того, чтобы сервис waf-sync на шлюзе начал следить за изменениями в базе данных и, по мере надобности, обновлять конфигурационные файлы и перезапускать сервисы. При отключенной опции waf-sync не выполняет указанные действия.

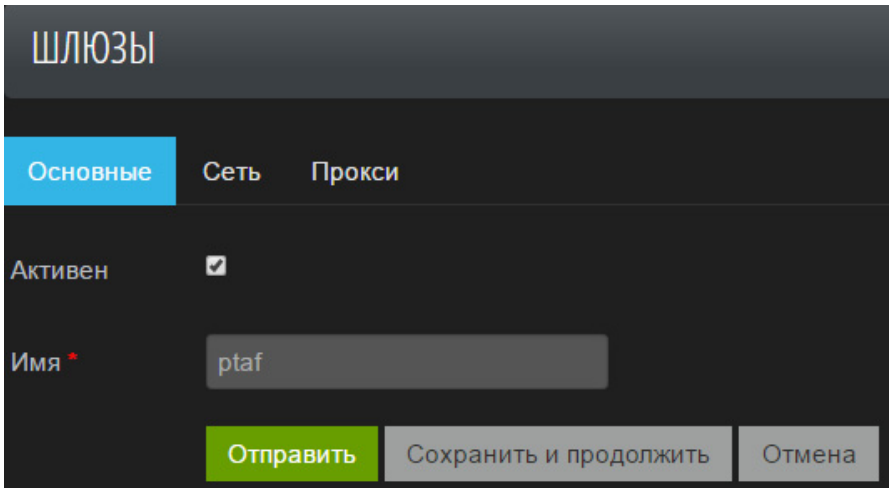


Рис. 100 – Основные опции шлюза

6.3.2.1.2. Сеть

Данная вкладка позволяет просмотреть конфигурацию сетевых интерфейсов PT AF и назначить интерфейсам алиасы (см. главу «Алиасы сетевых интерфейсов»). Вкладка содержит четыре группы настроек: *Сетевой интерфейс*, *Модули*, *DNS*, *Дата и время*. Изменения можно вносить только в первую группу, и только в части настроек списка алиасов.

6.3.2.1.2.1. Сетевой интерфейс

После добавления алиасов (см. главу «Алиасы сетевых интерфейсов») пользователь должен в поле *Сеть* назначить их на нужные сетевые интерфейсы.

Здесь отображается список интерфейсов, собранный при помощи CLI (см. главу «Настройка сети»).

6.3.2.1.2.2. Модули

Вкладка предназначена для просмотра маршрутов.

6.3.2.1.2.3. DNS

На вкладке отображаются IP-адреса DNS-серверов.

6.3.2.1.2.4. Дата и время (Date & Time)

На данной вкладке отображается часовой пояс и используемые на PT AF NTP-серверы, которые показывают актуальное, точное время, и крайне важны для журналирования и расследования инцидентов

6.3.2.1.3. Прокси

В системе представлены следующие настройки Proxy:

- *Количество процессов;*
- *Количество одновременных соединений.*

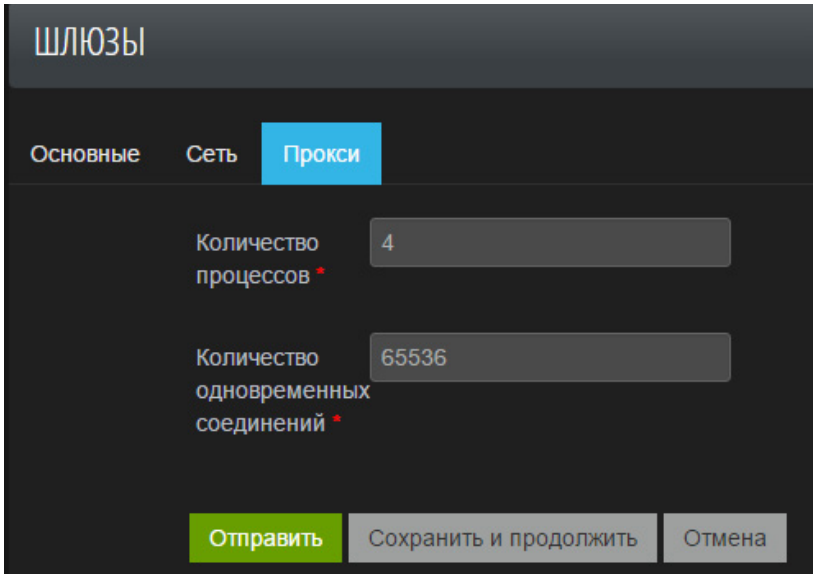


Рис. 101 – Настройки Proxy

6.3.2.2. Группа серверов

При работе в режиме обратного прокси-сервера, PT AF защищает либо отдельный сервер, либо кластер, состоящий из нескольких серверов. Для настройки адресов защищаемых PT AF серверов и предусмотрена вкладка *Группа серверов*.

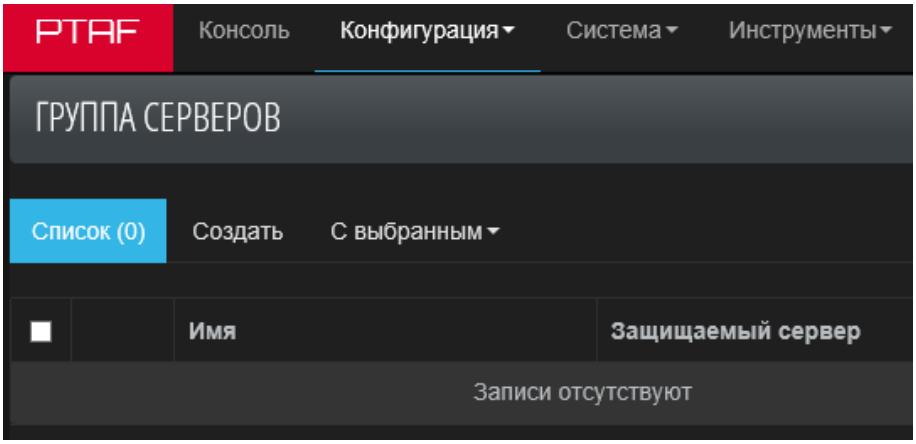


Рис. 102 – Группа серверов

Каждая группа серверов состоит из одного или нескольких *Защищаемых серверов*. Нажмите кнопку *Создать*, чтобы создать новую группу серверов. Затем нажмите кнопку *Добавить* в строке *Защищаемый сервер*, чтобы добавить сервер в группу и указать его настройки.

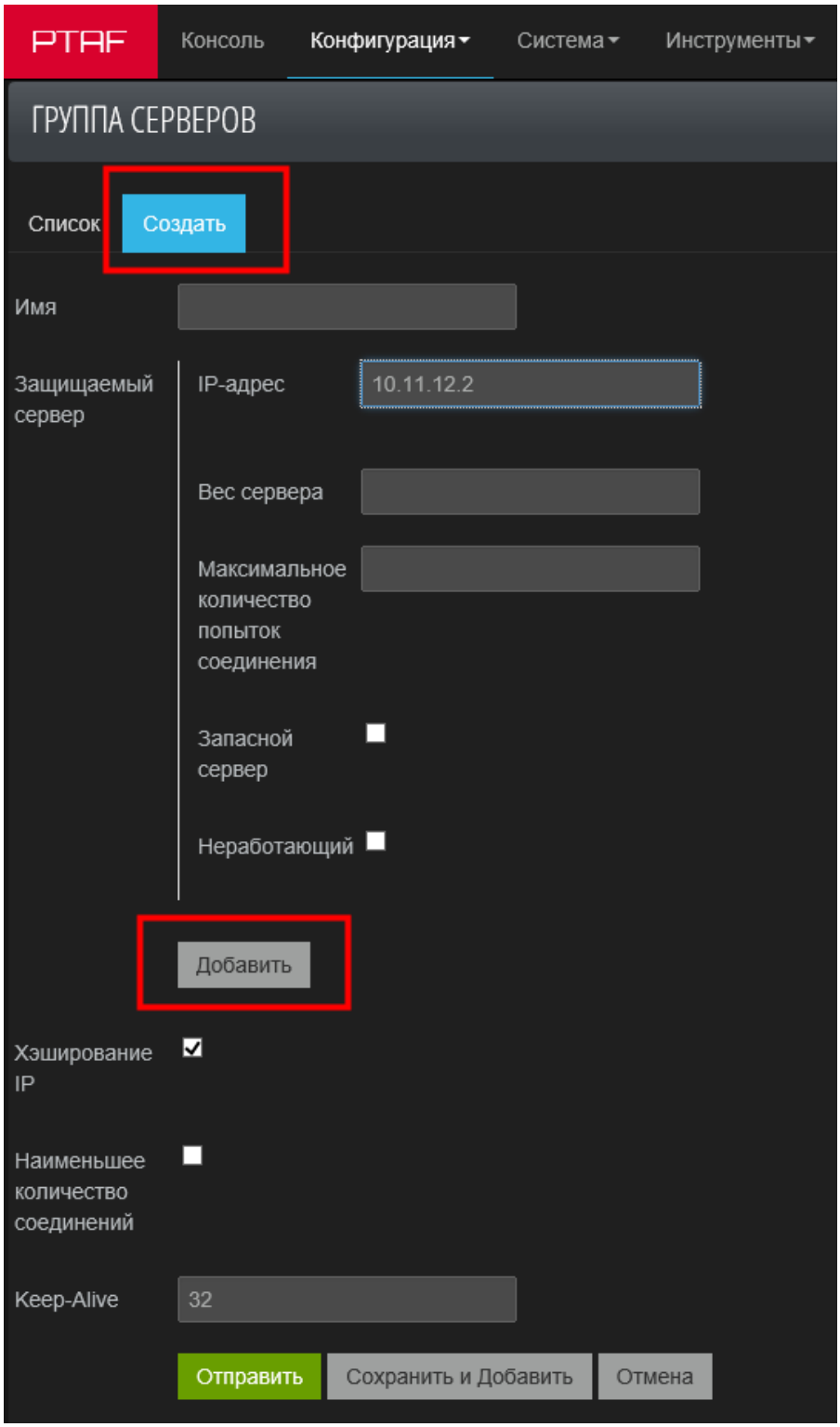


Рис. 103 – Создание группы серверов и добавление защищаемого сервера

В настройках защищаемого сервера указывается его IP-адрес или имя узла (hostname). В случае, если защищаемый сервер слушает нестандартный порт, отличный от 80, этот порт можно указать через двоеточие. Например: 192.0.2.123:8080.

Параметр *Вес сервера* означает частоту использования сервера. Вес может принимать целочисленное значение (по умолчанию равное 1). Если группа состоит из нескольких серверов, укажите вес сервера в группе. PT AF выбирает сервер, на который следует передать клиентский запрос, по алгоритму Round-robin, т.е. перебирает сервера по кругу согласно указанному весу (см. так же опцию *Наименьшее количество соединений*). Например, если в группу входит шесть серверов и вес сервера myserv-1.local равен 3, вес myserv-2.local равен 1 и вес myserv-3.local равен 2, то Nginx передает запрос первых трех клиентов к серверу myserv-1.local, четвертого — к myserv-2.local, а пятого и шестого — к myserv-3.local, после чего передача запросов продолжится заново по кругу.

Параметр *Максимальное количество попыток соединения* указывает максимальное количество неудачных попыток соединения за время описанное в параметре fail_timeout (настраивается в конфигурационном файле Nginx на PT AF). Если данное пороговое значение оказывается достигнутым, то сервер считается неработающим и в течение этого же периода времени не получает входящих соединений от PT AF. Значение по умолчанию помечает сервер как неработающий после первой же неудачной попытки. Нулевое значение отключает учет попыток и обращения к серверу направляются вне зависимости от наличия ответов.

Опция *Запасной сервер* помечает сервер как резервный. Запросы на него будут отправлены в том случае, если остальные сервера выйдут из строя.

Опция *Неработающий* предназначена для принудительного отключения сервера в группе.

В случае если PT AF обслуживает группу из нескольких серверов и синхронизация между веб-серверами отсутствует, становится важно направлять запросы приходящие в рамках одного сеанса работы клиента на один и тот же сервер. Для этой цели служит опция *Хеширование IP*.

Включение опции *Наименьшее количество соединений* меняет механизм выбора защищаемого сервера в группе серверов. Если без этой опции сервер выбирается по механизму round-robin, то при включенной опции *Наименьшее количество соединений*, при новом входящем соединении открывается соединение на тот сервер, к которому открыто наименьшее количество соединений (вес при этом продолжает учитываться). И только если взвешенное количество открытых соединений к серверам одинаково, применяется механизм round-robin.

Опция *Keep-Alive* позволяет включить пул соединений к защищаемой группе серверов. Если соединение закрывается клиентом, PT AF поддерживает соединение активным для ускорения последующих подключений к тому же серверу. Опция задает максимальное количество ожидающих соединений, которые могут быть одновременно открыты к группе серверов.

6.3.2.3. Файрвол

Вкладка предназначена для ведения черных списков IP-адресов. Для каждого IP-адреса в списке указаны следующие параметры: тип блокировки (tag), способ блокировки (права), дата прекращения блокировки и последняя правка.

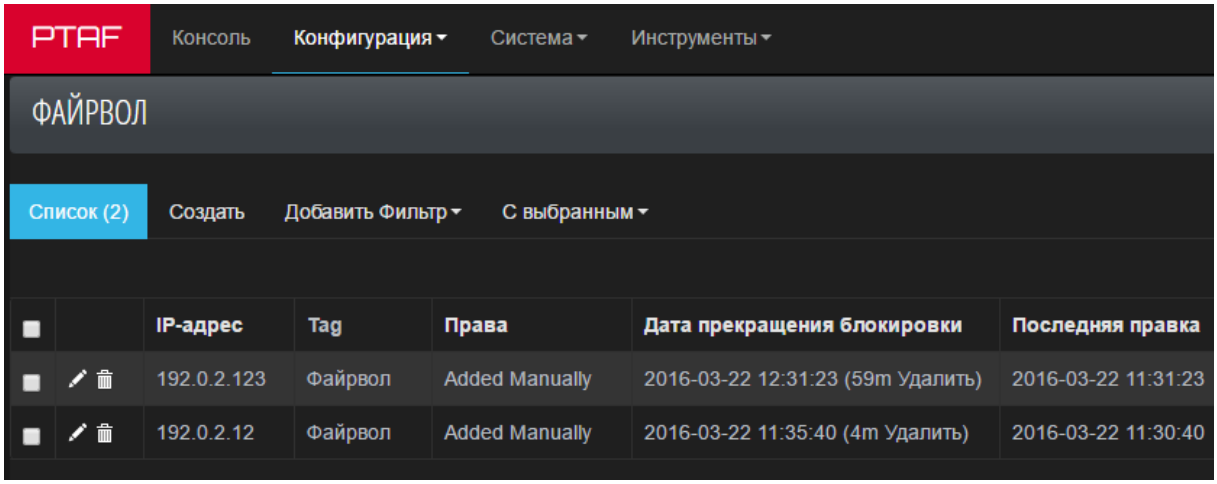


Рис. 104 – Черный список IP-адресов

IP-адрес можно заблокировать несколькими способами:

1. При помощи действия *Block IP*;
2. При помощи действия *Send to Arbor*.
3. Вручную через консоль в панели атак, нажав кнопку *Блокировать* (см. главу «Панель Attacks»);
4. Вручную на вкладке *Конфигурация -> Сеть -> Файрвол*.

Нажмите кнопку *Создать* на вкладке *Конфигурация -> Сеть -> Файрвол*, чтобы вручную добавить IP-адрес в список заблокированных. Укажите следующие параметры:

- IP-адрес;
- Tag – тип блокировки, указывающий на причину блокирования IP-адреса. Возможные значения: Файрвол и Arbor.
- Длительность – длительность нахождения IP-адреса в черном списке. По истечении времени блокировки, если не указано «навсегда», адрес будет выведен из списка.

Альтернативный способ проверки списка заблокированных IP-адресов рассмотрен в главе 10.12.

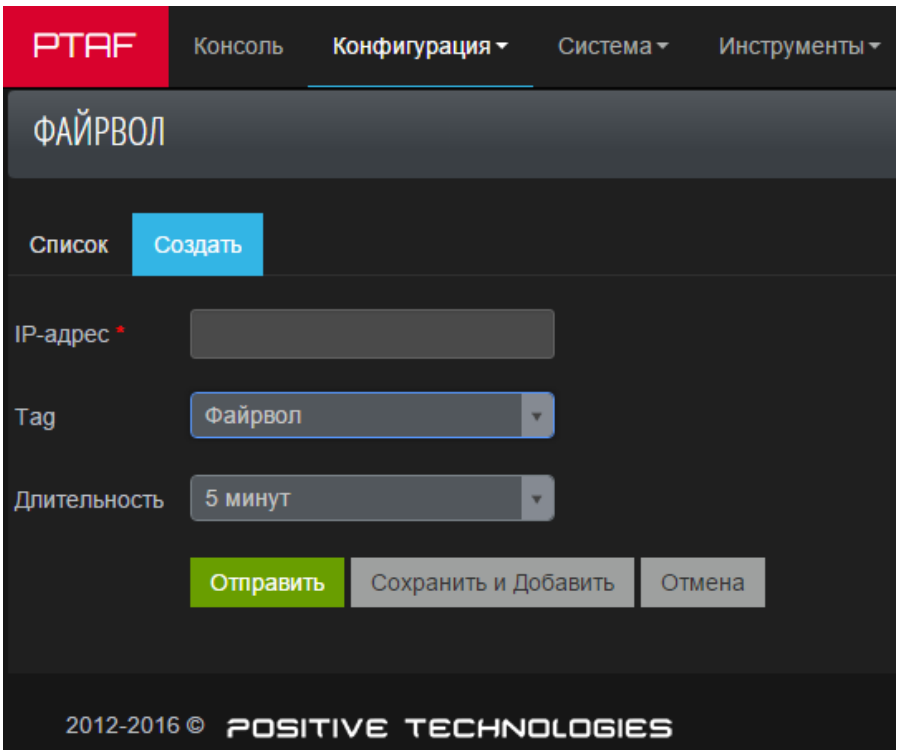


Рис. 105 – Файрвол

6.3.2.4. Сниффер

Группа опций на вкладке *Сниффер* предназначена для конфигурации источника данных, попадающих в PT AF в режиме мониторинга.

В опции *Сетевой интерфейс* следует указать сетевой интерфейс, подключенный к span-порту коммутатора. Этот интерфейс будет переведен в *promiscuous mode* и сетевой трафик при помощи модуля *tapered* через доменный сокет будет направлен в PT AF для анализа.

Опция *Режим* позволяет выбрать, в каком из режимов будет работать сниффер:

- User space – режим, реализуемый сервисом waf-sniffer;
- Kernel space – режим, реализуемый сервисом taperedng.

Примечание: выбор производится в зависимости от предпочтений пользователя.

Опция *Сервер* предназначена для того, чтобы указать PT AF трафик к какому защищаемому серверу необходимо анализировать. Здесь усматривается некоторая аналогия с настройками на вкладке *Группа серверов*, где указывается сервер, защищаемый в режиме обратного прокси. Тут же есть возможность передать в PT AF приватный SSL-ключ и, при необходимости, пароль, которым он был зашифрован. Загрузка ключей в PT AF делает возможным мониторинг за шифрованным трафиком. В настройках сервера следует указать один или несколько портов, используемых защищаемым узлом для приема веб-трафика.

Опция *Сетевой интерфейс для отправки RST* предназначена для обрыва трафика злоумышленника путем отправки TCP-Reset пакета методом TCP-HiJack.

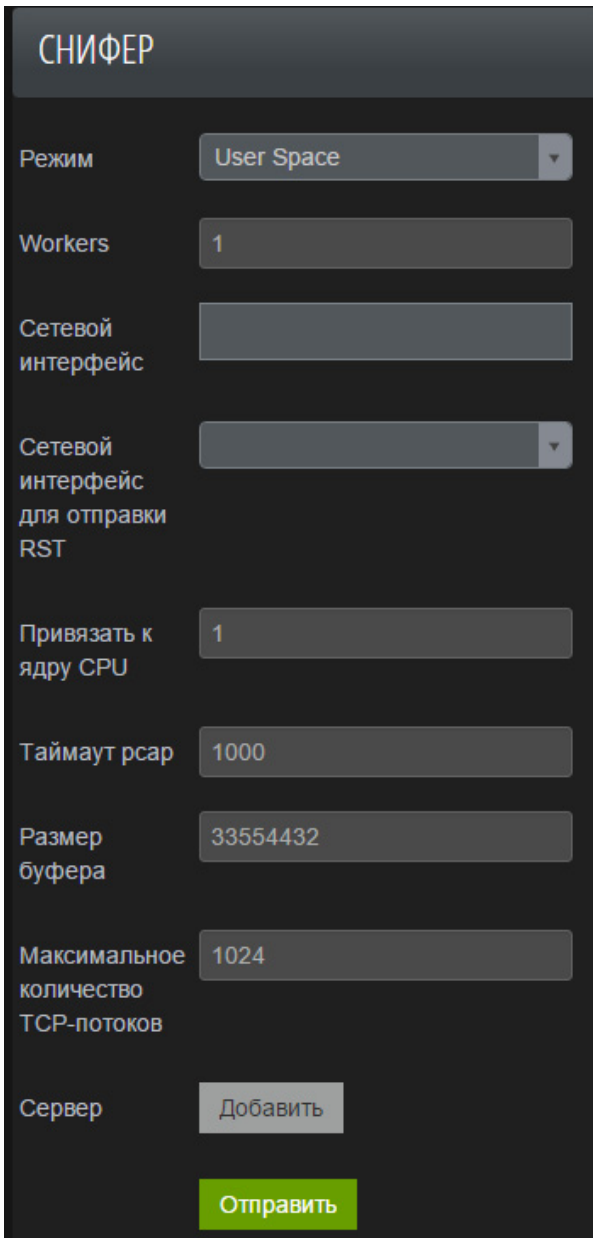


Рис. 106 – Настройки сниффера

6.3.2.5. Алиасы сетевых интерфейсов

На вкладке *Алиасы сетевых интерфейсов* настраиваются логические роли для интерфейсов (SPAN, Management, WAN, LAN), т.е. задаются алиасы (псевдонимы) для последующего их применения к физическим сетевым интерфейсам, к которым подключен PT AF.

При создании алиаса ему указывается тип (WAN, LAN, MGMT, SPAN), после чего можно дополнительно указать ему список нестандартных портов (TCP и UDP), которые должны быть открыты. Для интерфейса MGMT порт 22 является зарезервированным и открыт по умолчанию.

В системе также всегда присутствует алиас *Default* типа DB с зарезервированными портами для взаимодействия со всеми необходимыми для функционирования PT AF СУБД.

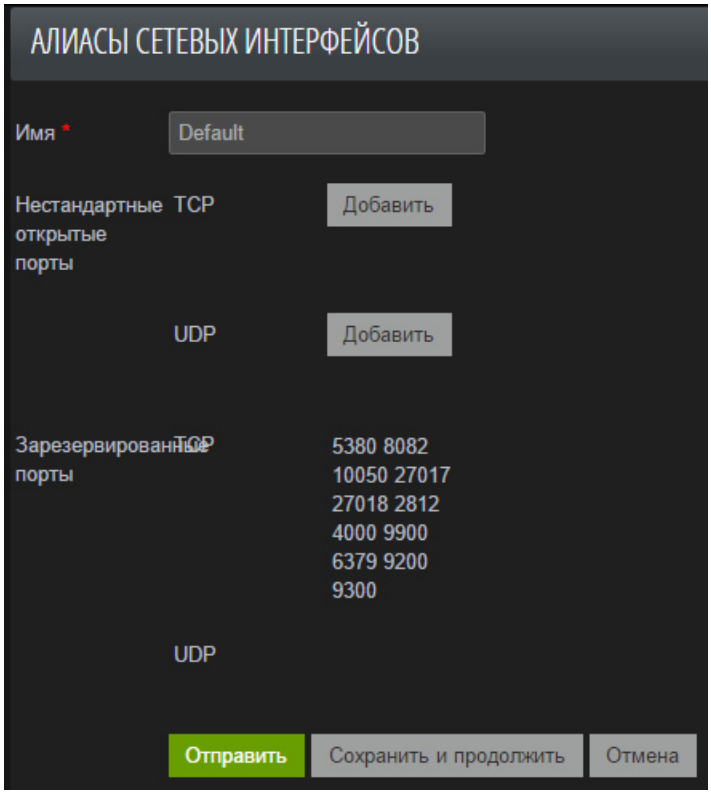


Рис. 107 – Network Interface Aliases

6.3.2.6. Виртуальные IP

На данной вкладке частично представлены настройки кластера, когда PT AF работает в виде кластера из нескольких устройств: общие IP-адреса, которые делятся между несколькими элементами PT AF, в зависимости от их статуса (работает/сломался) и режима работы (active/passive или active/active).

6.3.2.6.1. Поддержка режима работы active/active

Для режима работы active/active на данной вкладке настраиваются следующие опции:

- Опция *Load balancing* включает режим балансировки нагрузки между нодами кластера. При включении опции появляются дополнительные настройки. Опция *Виртуальные IP -> Интерфейс синхронизации* позволяет указать интерфейс синхронизации между нодами кластера. Общая настройка *Delay Loop* предназначена для указания интервала синхронизации.

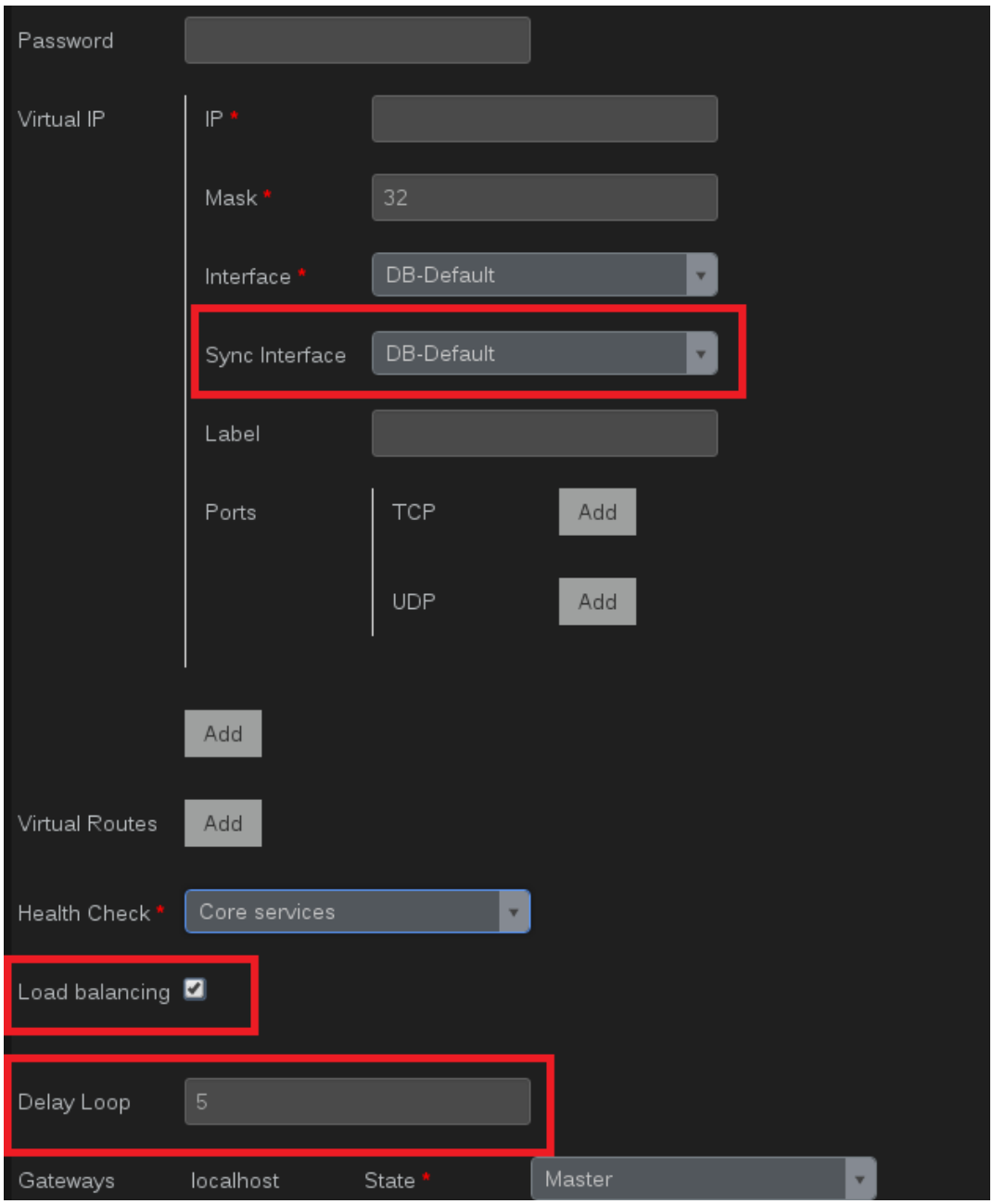


Рис. 108 – Виртуальные IP

После выбора виртуального IP в окне редактирования профиля в поле IP-адрес или после изменения настроек виртуального IP, который уже связаны с профилями, waf-sync добавляет в конфигурацию keepalived настройки, отвечающие за балансировку нагрузки.

За балансировку нагрузки отвечает секция virtual_server. Каждому набору из профиля, одной из его секций во вкладке *Прокси* раздела *Серверы* и выбранного в нем виртуального адреса, соответствует одна конфигурация virtual_server:

```
virtual_server fwmark {% if vs.ssl %}200{% else %}100{% endif %} {
    delay_loop {{ vs.delay_loop|default(5, True) }}
    lb_algo lc
```

```
        lb_kind NAT
        net_mask {{ vs.netmask }}
        protocol TCP
        virtualhost {{ virtualhost }}
        {% for real_ip in vs.real_servers_ips %}
        real_server {{ real_ip }} {{ vs.port }} {
            weight 1
            inhibit_on_failure
            HTTP_GET {
                url {
                    path /healthcheck/index.html
                    status_code 200
                }
                connect_port {{ vs.port }}
                connect_timeout 10
                nb_get_retry 3
                delay_before_retry 8
            }
        }
        {% endfor %}
    }
```

где

- vs.ssl - признак включенности SSL у соответствующего сервера;
- vs.netmask - маска Virtual IP;
- virtualhost - первый hostname из заданных в профиле или ptaf.ru;
- vs.real_servers_ips - адреса нод, соответствующие интерфейсу Sync Interface;
- vs.port - порт соответствующего сервера;
- real_ip - адрес ноды, соответствующий интерфейсу Sync Interface.

С такой конфигурацией нагрузка, поступающая по виртуальному адресу, выбранному для сервера, будет распределяться между нодами кластера, пока они находятся в рабочем состоянии. Работоспособность нод определяется проверкой статуса ответа на HTTP GET запрос keepalived по специальному пути /healthcheck/index.html. WafNginx на каждой ноде настроен на ответ со статусом 200 по этому пути.

6.3.3. Корреляция

Перейдите к одной из вкладок данного раздела, чтобы посмотреть или настроить *События* и *Корреляции*.

6.3.3.1. События

Событие (составное событие, Event) – это последовательность из одного или нескольких Составных или Базовых событий в определенном диапазоне времени.

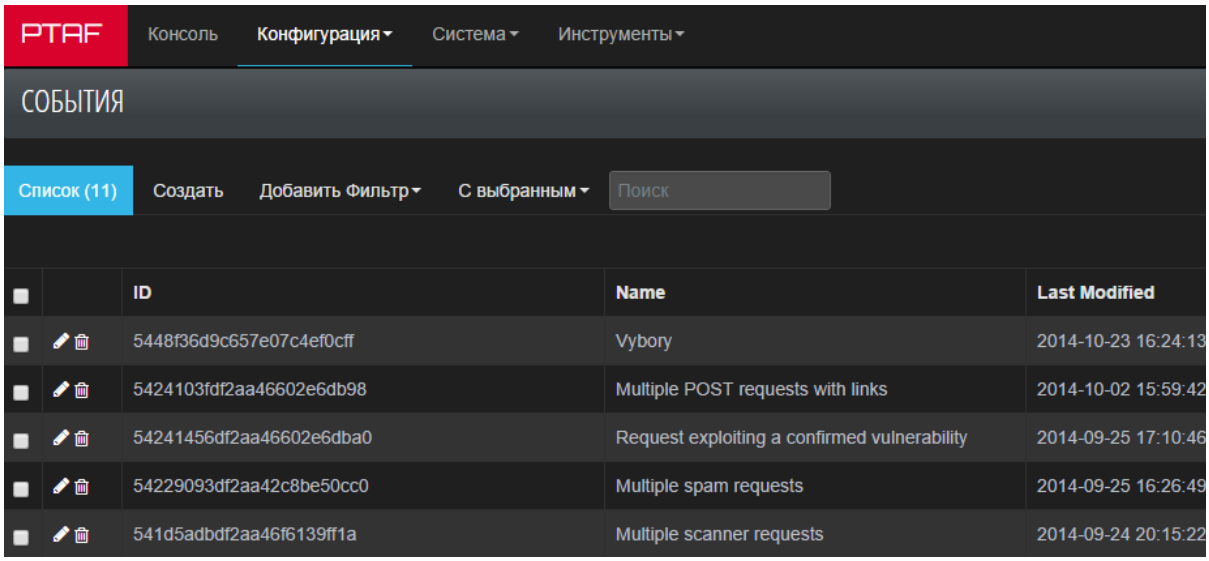


Рис. 109 – События

На вкладке *События* отображается список шаблонов составных событий.

Шаблон события описывает правила возникновения составного события: на основе одного или нескольких экземпляров одного и того же родительского события (Parent Event), объединенных по определенным условиям (например, по значениям тегов или других атрибутов). Для шаблона события можно задать минимальное количество срабатываний события, лежащего в его основе, а также временные диапазоны для ожидания срабатывания и таймаута.

В качестве родительского события для шаблона можно использовать либо Базовое событие, либо Составное.

При добавлении потенциального события указываются следующие параметры:

- Родительское событие – событие, срабатывания которого могут породить данное;
- Условие – набор объединяемых через И или ИЛИ правил, накладываемых на значения атрибутов родительских событий. Поддерживаемые операторы:
  - =
  - !=
  - >
  - <
  - >=
  - <=
  - contains
  - does not contain
  - regex
  - in
  - not in

• Минимальное количество срабатываний условия – сколько раз должно сработать условие, для того, чтобы был сгенерирован экземпляр данного события;

• В промежутке – указывается интервал времени, в течение которого должно произойти «Минимальное количество срабатываний условия» для генерации экземпляра события;

• Продолжать накапливать события в течение – после того как экземпляр составного события сгенерирован, следующие срабатывания условий будут продолжать добавляться к данному событию, а не образовывать новое.

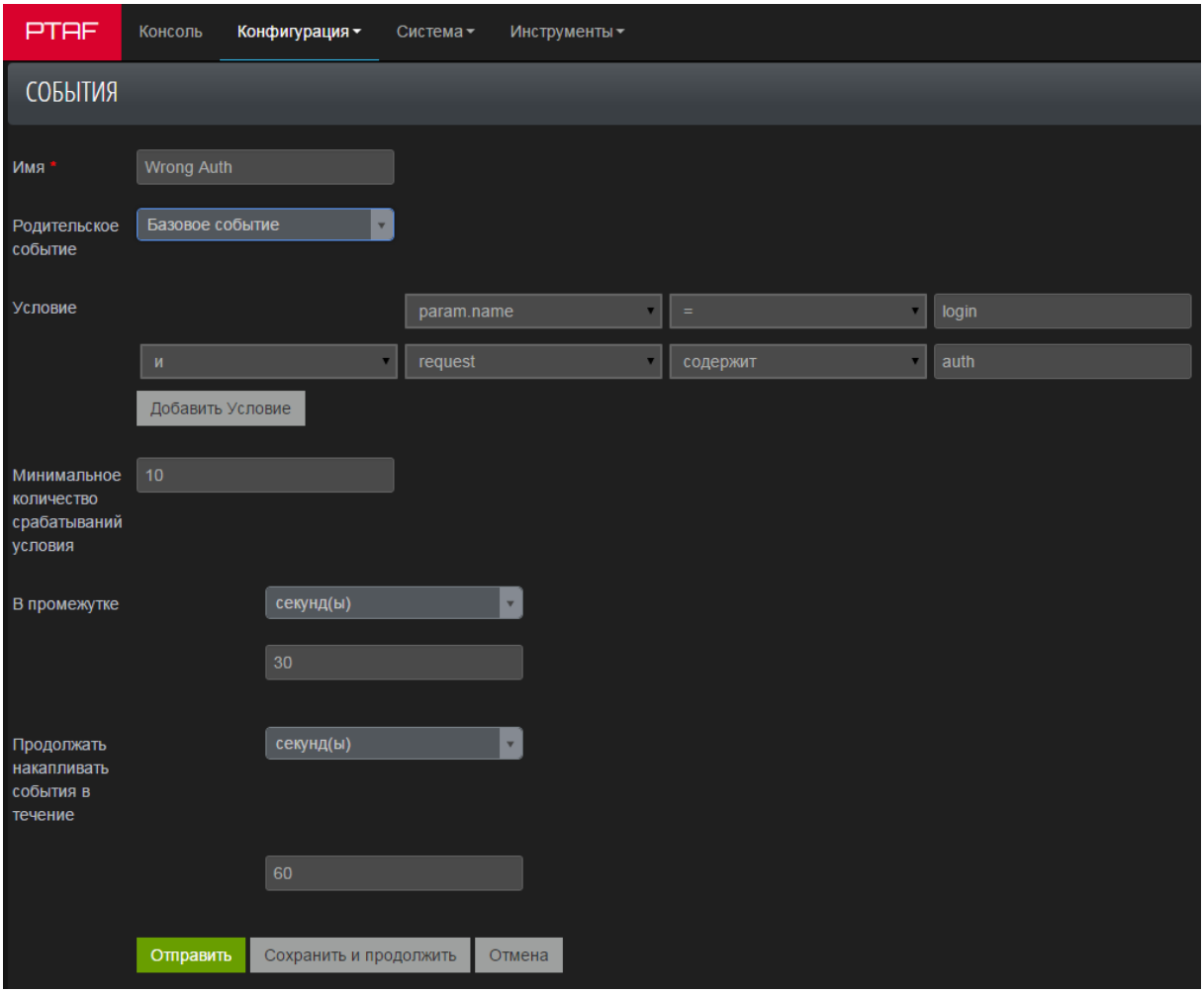


Рис. 110 – Редактирование события

6.3.3.2. Корреляции

Корреляции (Alerts) – это правила, позволяющие объединять последовательность составных событий в логическую сущность и генерировать алерт в системе.

Корреляции – это правила корреляции, которые состоят из одного и более потенциальных событий и предназначены для объединения нескольких событий в одно скоррелированное событие путем выставления в событии атрибута correlation_id. Правила корреляции обладают следующими атрибутами:

• Имя – имя алерта;

• Родительская корреляция – правило корреляции, которое должно сработать перед данным. Если в поле указано «Базовая корреляция», то это правило корреляции работает на событиях, а не корреляциях;

• Описание – описание алерта;

• Группа – группировка событий по соответствующему полю. В качестве поля группировки можно выбрать такие поля составного события как browser, city, country, event.severity, ip, os, param.name и другие. Например, если создать правило корреляции, описывающее атаку BruteForce, то правило корреляции может сработать оттого, что попытки неудачного входа в систему идут одновременно от разных пользователей. Если же мы сделаем привязку по объекту IP, то правило корреляции будет срабатывать, только если неудачные попытки входа идут с одного IP-адреса.

• Events Sequence – последовательность составных событий, которая должна быть соблюдена для работы данного правила корреляции. Порядок перечисления соответствует порядку событий, которые должны будут произойти. Если для срабатывания правила корреляции порядок срабатывания событий не важен, то достаточно сформировать еще одно составное событие, в котором будет несколько условий объединенных через ИЛИ и сформировать правило корреляции на его основе.

• Действие – действие, которое можно выполнить по результатам срабатывания правила корреляции и появления скоррелированного события. Возможные варианты:

- Log to DB – поместить информацию о скоррелированном событии в базу PT AF;

- Send to ArcSight – отправить информацию о скоррелированном событии в систему ArcSight;

- Send to syslog – отправить информацию о скоррелированном событии внешний сервер syslog;

- Bock IP – блокировать источник скоррелированного события в межсетевом экране (см. описание вкладки *Файрвол*).

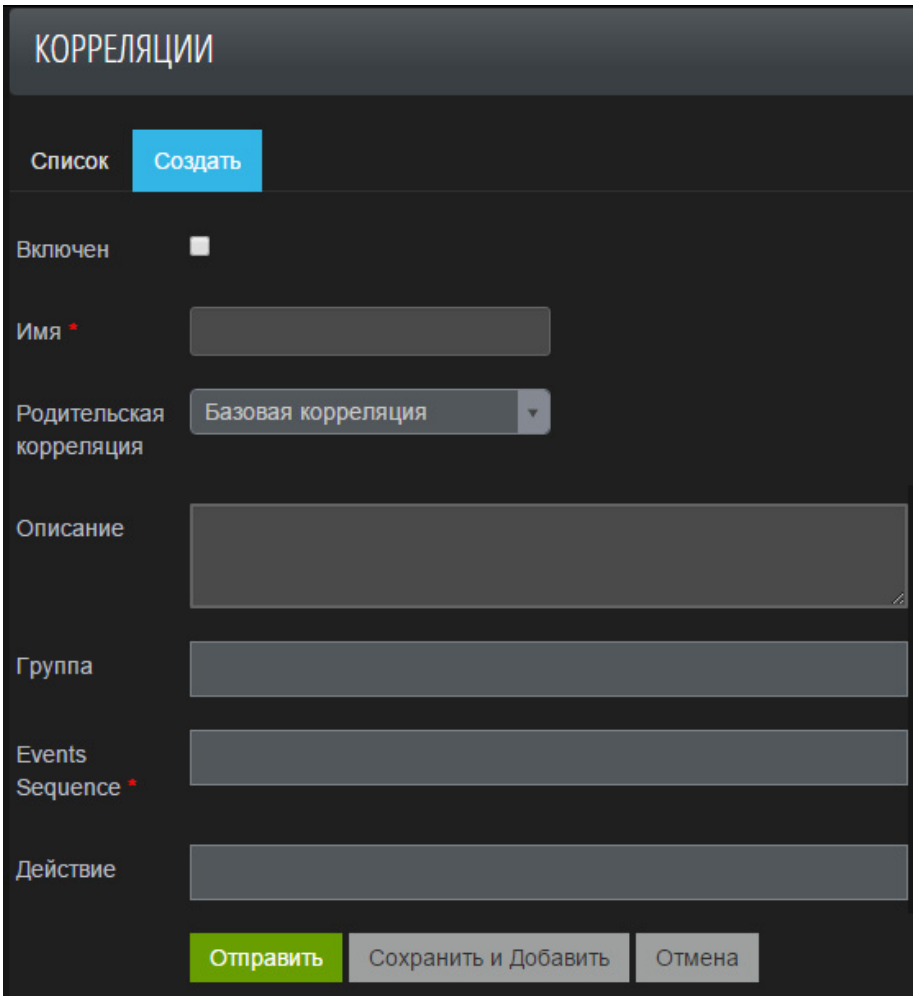


Рис. 111 – Корреляции

6.3.4. Конфигурация

К данному разделу относятся вкладки *ICAP-сервисы* и *LDAP-сервисы*.

6.3.4.1. ICAP-сервисы

PT Application Firewall позволяет осуществлять дополнительную проверку загружаемых файлов с помощью внешних систем, интегрированных с протоколом icap. По умолчанию используется потоковый антивирус ClamAV, но можно осуществлять интеграцию и с другими потоковыми антивирусами и DLP-системами. Подключение к профилю и отключение от него сервисов ICAP выполняется на вкладке *Конфигурация -> Политики безопасности -> Профили -> Модули -> ICAP-интеграция.*

На вкладке *ICAP-сервисы* можно настроить список систем, которые будут анализировать файлы, загружаемые с/на приложение. Для этого указываются параметры:

- *Имя*: название сервиса;
- *Режим ICAP*: режим работы ICAP-сервера (request/response modification);
- *ICAP URI*: ссылка, на которую будут отправляться обнаруженные файлы;
- *Событие*: тип события – загрузка файла на/с приложения;

• *Заголовок ICAP-ответа*: заголовок, в котором будет передаваться решение сервера – пропустить или заблокировать запрос.

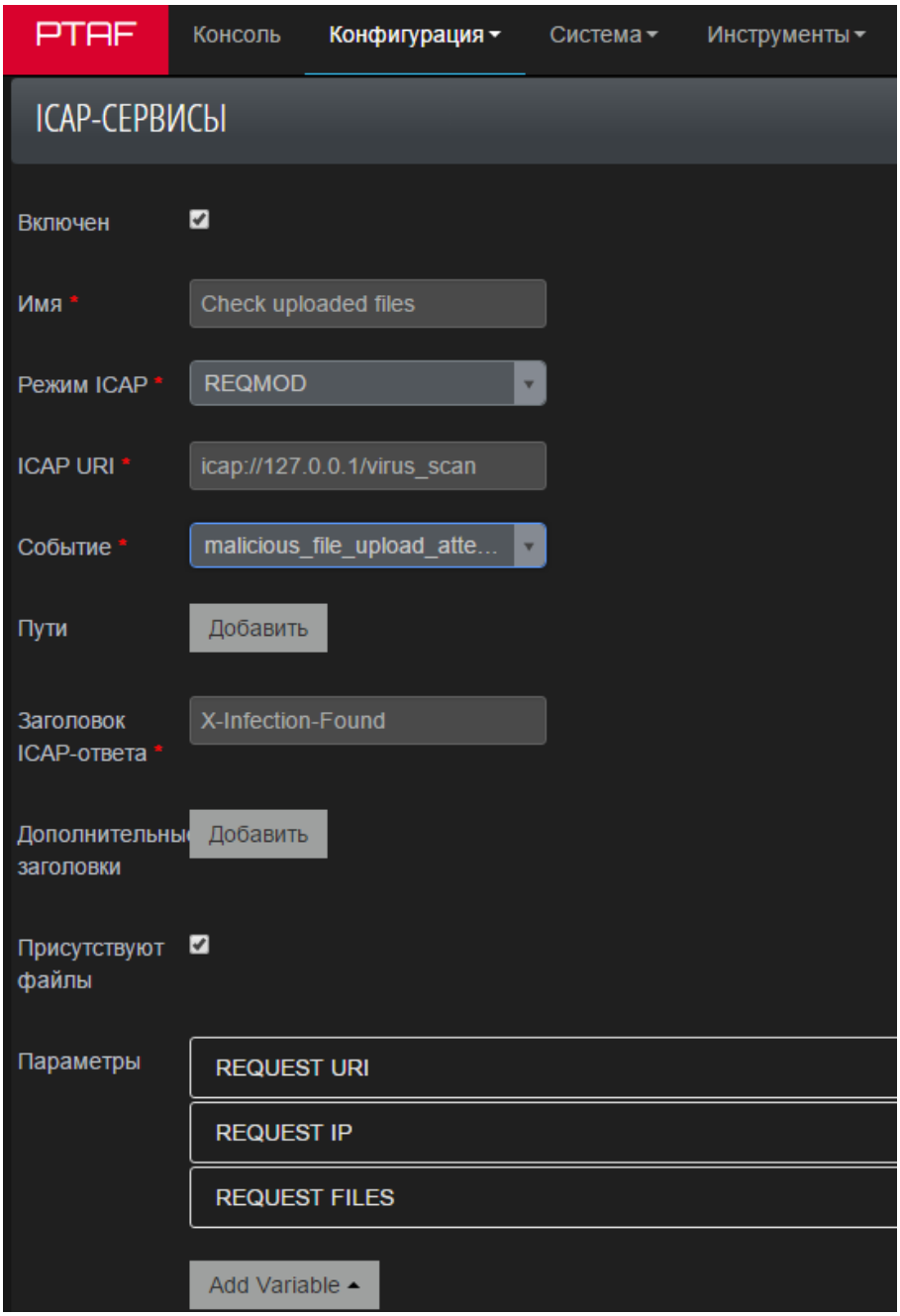


Рис. 112 – ICAP-сервисы

6.3.4.2. LDAP-сервисы

Настройка предназначена для подключения к ActiveDirectory (для работы защитного механизма LDAP).

На вкладке *LDAP-сервисы* выполняется настройка механизма OpenLDAP для доменной аутентификации пользователей при их обращениях к XML API. Здесь можно задать URI LDAP-сервиса и его настройки безопасности (SASL, SSL).

Проверка прав пользователя на XML-ресурсы и дальнейшие настройки действий задаются в настройках модуля защиты *LDAP-авторизация*.

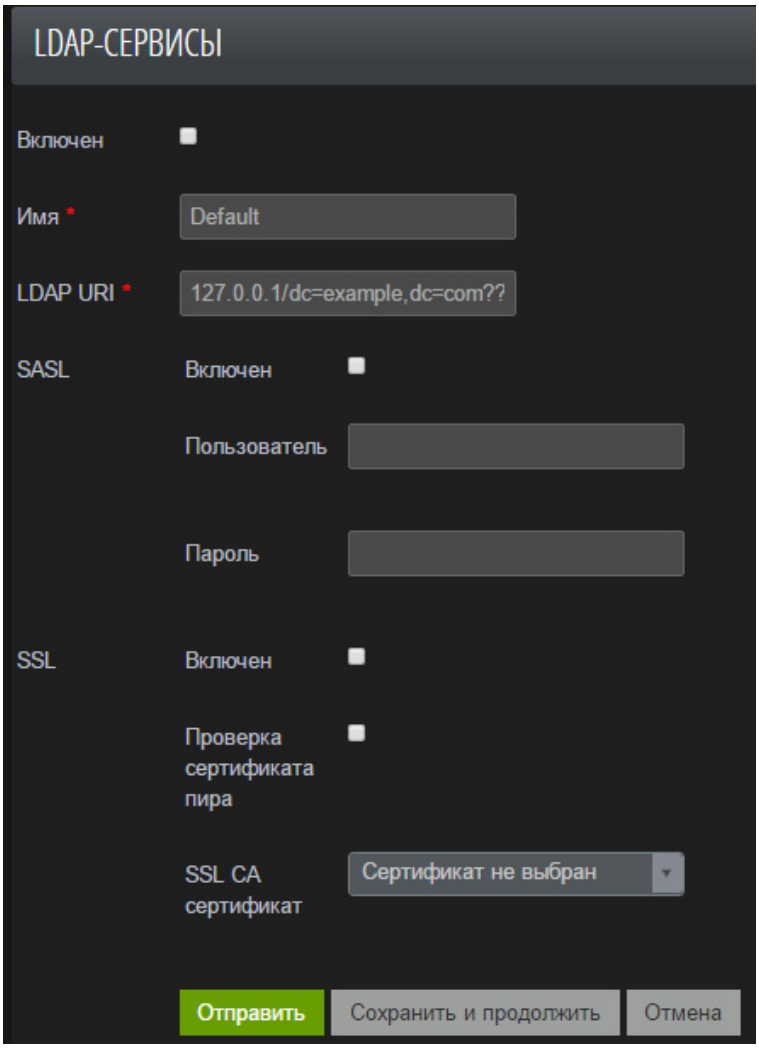


Рис. 113 – Настройки LDAP-сервиса

6.3.5. SSL-сертификаты и ключи

На данной вкладке можно загрузить ключи и сертификаты используемые для SSL-защиты сайтов, расположенных за PT AF.

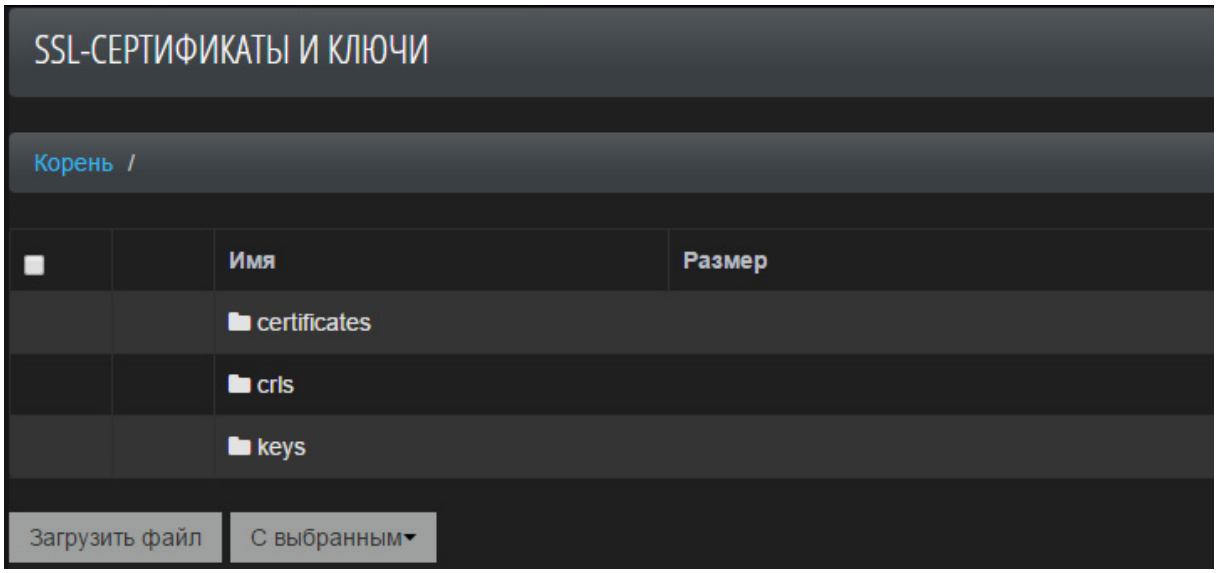


Рис. 114 – SSL-сертификаты и ключи

Нажмите кнопку *Выберите файл*, чтобы выбрать файл ключа/сертификата и загрузить его в общий список.

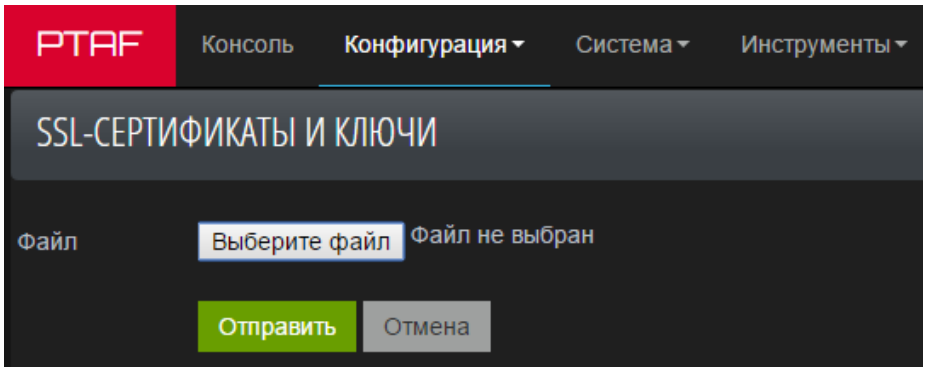


Рис. 115 – Загрузка файла ключа/сертификата

Чтобы удалить ключ/сертификат, выберите необходимую запись или несколько записей и нажмите кнопку *Удалить*. Для замены файла с ключом или сертификатом, нажмите на кнопку 🔄 возле имени файла.

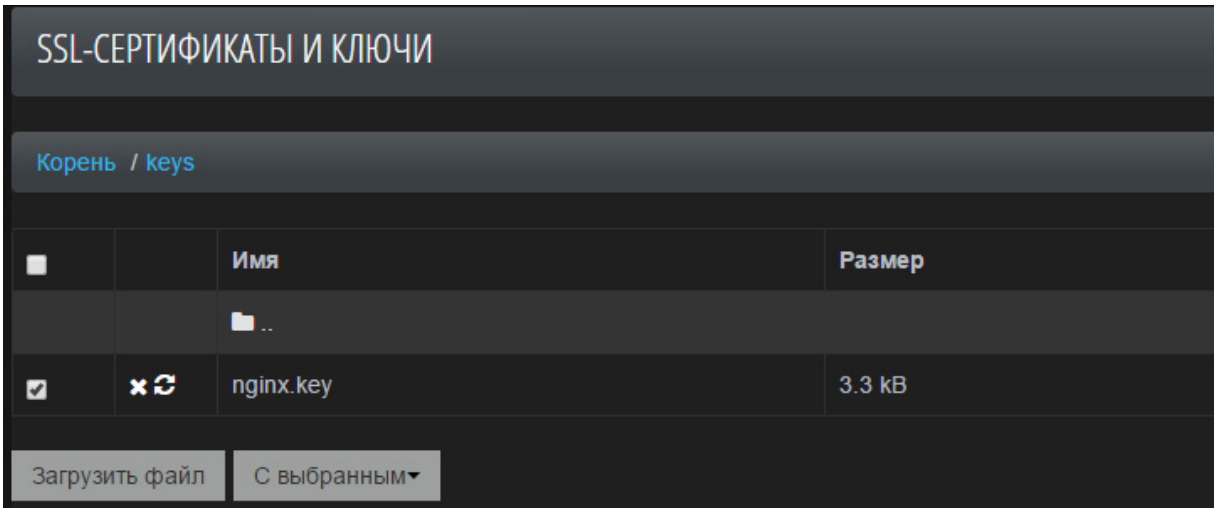


Рис. 116 – Удаление или замена ключа/сертификата

6.3.6. Подозрительные сессии

На вкладке *Конфигурация -> Suspicious Sessions* находится список подозрительных сессий. В списке указаны следующие параметры сессии:

- *Тип* – тип сессии. Возможные значения: *Session*, *User*. Тип *Session* присваивается, когда в клиентском запросе содержатся Cookie, выданные защитным модулем *Отслеживание сессий*. Тип *User* присваивается в случае отсутствия в клиентском запросе такой Cookie;
- *Session ID* – идентификатор сессии. Принимает значение в зависимости от типа сессии. Если тип сессии *Session*, то идентификатор является 96-и символьной строкой, если тип сессии *User*, то – 16 символьной строкой.
- *IP-адрес*;
- *User Agent* – приложение, используемое клиентом;
- *Expire At* – время истечения блокировки. Значение, заданное по умолчанию: 5 минут.

Для каждой сессии на данной вкладке можно изменить свойства. Например, изменить время блокировки сессии. Чтобы удалить сессию из списка подозрительных, выберите опцию *Delete*.

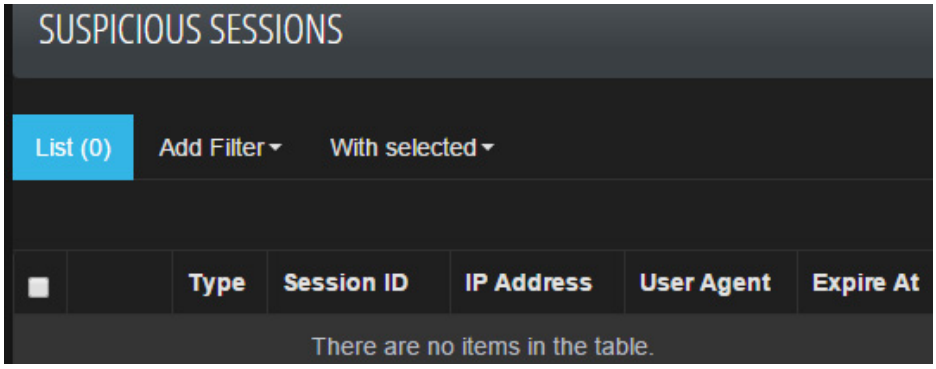


Рис. 117 – Список подозрительных сессий

6.4. Система

В разделе *Система* доступны следующие вкладки:

- Статус;
  - Services status;
  - Мониторинг;
  - Services Event Log;
- Пользователи;
  - Пользователи;
  - Группы пользователей;
  - Журнал действий пользователей;
  - Настройки Active Directory;
- Настройки обучающего модуля;
- Web UI Settings;
- Web UI Security Settings;
- About.

6.4.1. Статус

6.4.1.1. Services status

Во вкладке *Система -> Статус -> Services status* можно проследить за состоянием сервисов всех нод кластера.

Для каждой ноды выделена своя страница. Цветовой индикатор ноды указывает на работоспособность сервисов. Если цвет индикатора зеленый, то все сервисы находятся в рабочем состоянии. Если цвет индикатора красный, то следует проверить работоспособность системы, так как в работе сервисов произошли неполадки.

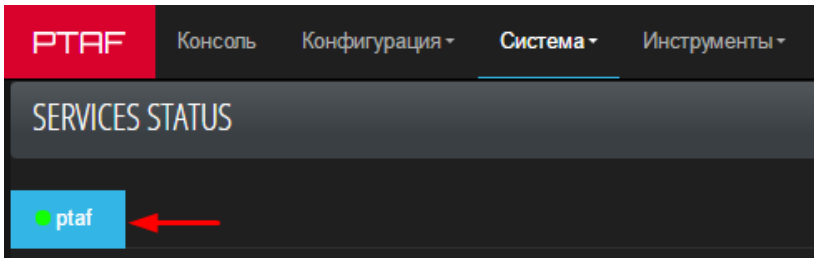


Рис. 118 – Индикация состояния ноды

Все сервисы разделены на два списка.

• Верхний список отображает статус дополнительных программ-проверок. К верхнему списку относятся следующие сервисы:

- check_status_UI, check_status_DB, check_status_GW – отображают состояние сети;
- health_check_guardant – отображает проверку статуса ключа.

- В нижнем списке отображаются метрики работоспособности и прочая информация о сервисах. Также здесь доступны ссылки на события по конкретному сервису в журнале событий. Полный список сервисов нижнего списка представлен в Табл. 8.

Каждому сервису в нижнем списке присвоен один из следующих статусов:

- Running – сервис находится в рабочем состоянии;
- Error – в работе сервиса произошли неполадки, требуется проверить состояние системы. Нажмите кнопку *View* в строке сервиса, чтобы выяснить причину неполадки;
- Not Monitored – отключен мониторинг состояния сервиса;
- Pending – сервис запущен, проверяется состояние сервиса.

Подробная информация по изменению состояния сервиса представлена в главе «Управление сервисами».

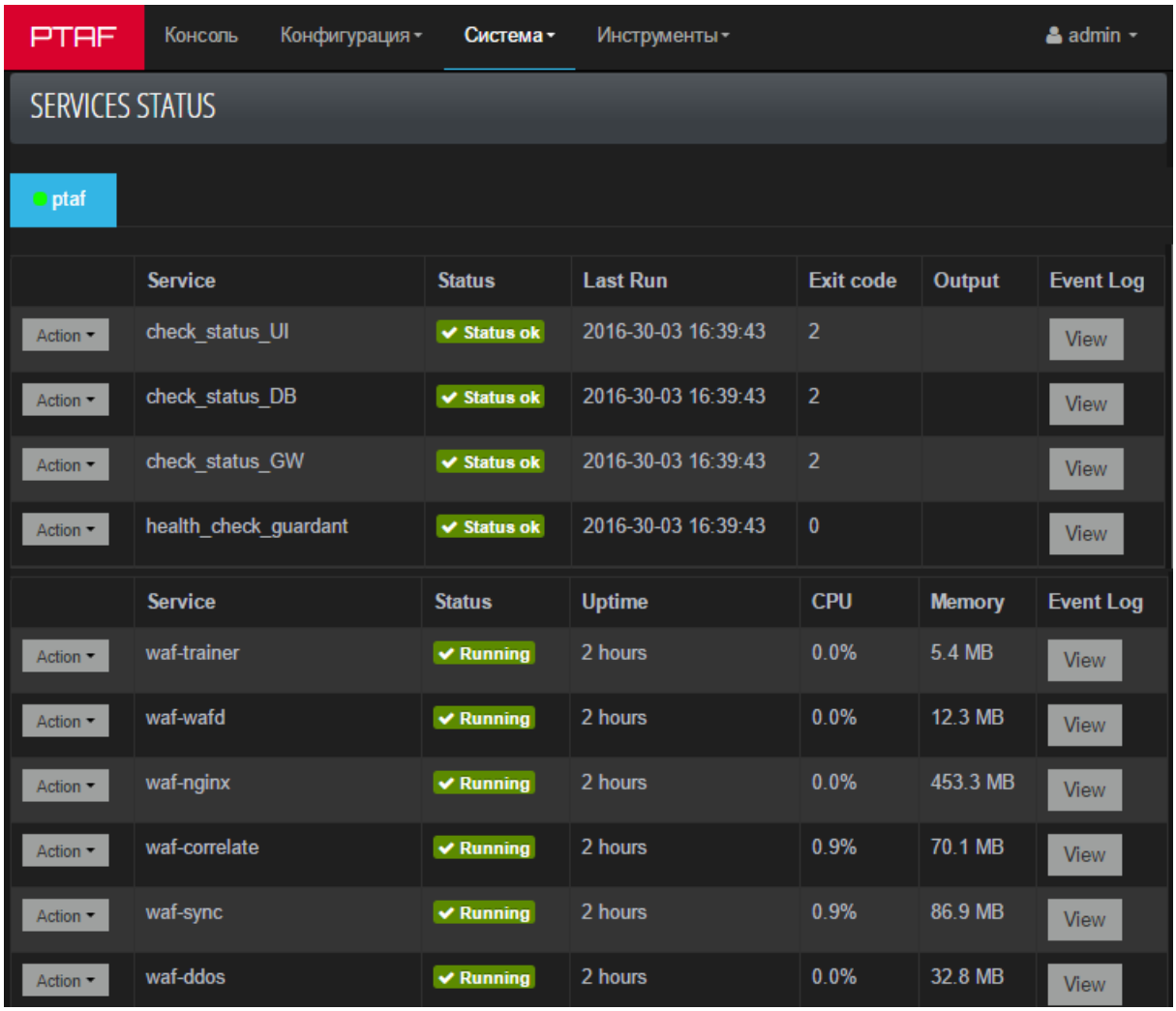


Рис. 119 – Статус

Таблица 8. Сервисы

Сервис	Описание
waf-trainer	Выполняет задачи по обучению модуля защиты HMM для создания HMM-моделей. Если запустить данный сервис, то дополнительно запустится зависимый от него сервис mongod
waf-wafd	Выполняет задачи по отправке в SysLog, записи в базу данных Elasticsearch и работе с ipset. Если запустить данный сервис, то дополнительно запустится зависимый от него сервис mongod
waf-nginx	Веб-сервер nginx с модулем Application Firewall. Если запустить данный сервис, то дополнительно запустится зависимый от него сервис mongod
waf-correlate	Сервис построения корреляций между атаками. Если запустить данный сервис, то дополнительно запустится зависимый от него сервис redis
waf-sync	Читает новые записи (как команда tail -f) в коллекции tailable, в которую были записаны операции (insert, update, delete) службой Mongo, и обновляет конфигурационные файлы, отправляет SIGHUP
waf-sniffer/ taperedng	Реализует работу PT AF в режиме сниффера
waf-ddos	Реализует работу модуля защиты от DDoS
ui	Отвечает за интерфейс системы. Если запустить данный сервис, то дополнительно запустится зависимый от него сервис mongod
graphite	Хранит цифровые данные временных рядов и выдает по требованию их графическое представление
waf_api	Часть rest api, которая может аналогично выполнять некоторые функции сервиса ui
wsc_api	Часть rest api, которая управляет настройками сети и базы данных
nginx	Фронтенд-сервер интерфейса
celery	Сервис распределенного выполнения задач. Если запустить данный сервис, то дополнительно запустится зависимый от него сервис redis
celerybeat	Сервис распределенного выполнения периодических задач (аналог cron)
redis	Ключевой сервис для хранения кэш-данных корреляций и очереди задач celery. Если запустить/отключить данный сервис, то дополнительно запустится/ отключится зависимый от него сервис celery и waf-correlate
diamond	Сервис сбора метрик и отправки на carbon
carbon	Сервис приема и хранения метрик. Составная часть graphite
libreoffice	Используется для составления отчетов
mongod	База данных для постоянного хранения конфигураций и прочей нужной информации. Если запустить/отключить данный сервис, то дополнительно запустятся/отключатся зависимые от него сервисы uwsgi, waf-nginx, waf-wafd и waf-trainer
waf-keepalived	Используется для балансировки в кластере
elasticsearch	База данных для хранения атак

Таблица 8. Сервисы

Сервис	Описание
c-icap	ICAP-сервер для взаимодействия с антивирусами и DLP
clamd	ClamAV антивирус
clamav-fresh	Выполняет задачи по обновлению ClamAV антивируса
monit	Сервис мониторинга за прочими сервисами, позволяющий запускать и останавливать их. Не отображается в списке сервисов во вкладке Services Status.

6.4.1.1.1. Управление сервисами

Нажмите кнопку *Action*, а затем выберите необходимую команду. Список команд следующий:

- Stop – остановить сервис;
- Start – запустить сервис;
- Restart – перезапустить сервис;

Примечание: действия *Stop, Start* и *Restart* недоступны для сервисов, от которых зависит интерфейс.

- Disable Monitoring – отключить мониторинг сервиса. Действие *Disable Monitoring* применимо ко всем сервисам.

Внимание! Отключение сервисов не рекомендуется. Остановка некоторых сервисов может привести к остановке других, зависимых от них, сервисов. Для правильной работы системы все сервисы должны находиться в рабочем состоянии и иметь статус *Running*.

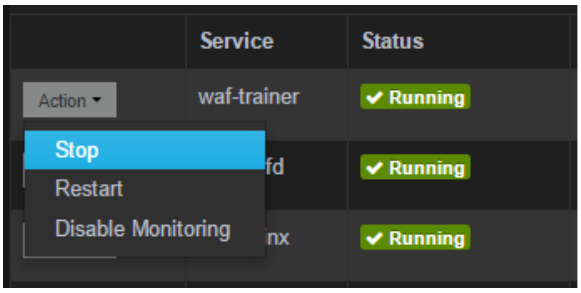


Рис. 120 – Команды управления сервисами

При отключении сервиса важно учесть, в каком режиме работает система.

- Для выполнения всех задач, в частности для режима обратного прокси-сервера (Reverse Proxy) критичными сервисами, то есть определяющими базовую функциональность, являются:
  - waf-wafd
  - waf-sync
  - waf-nginx
  - waf-correlate
  - mongod
  - elasticsearch

Если несколько приложений объединены в кластер, то добавляется:

- waf-keepalived

• Для режима Сниффер требуется сервис:

- waf-sniffer (для режима User Space)/taperedng (для режима Kernel Space)

Примечание: сервисы waf-sniffer и taperedng являются взаимоисключающими и включаются в зависимости от режима, выбранного в разделе *Конфигурация -> Сеть -> Сниффер*.

• Для режима работы Forensics, работы виртуального патчинга (Virtual Patching) требуются:

- redis
- celery
- celerybeat

• Для создания отчетов необходимы:

- libreoffice
- redis
- celery
- celerybeat
- elasticsearch

• Для создания резервных копий требуются:

- celery
- celerybeat
- elasticsearch
- redis
- mongod

• Для работы интерфейса пользователя необходимы:

- ui
- nginx

• Для поддержки мониторинга состояния и загруженности компонентов необходимы:

- diamond
- carbon
- graphite

Внимание! Для управления всеми сервисами PT AF необходим сервис monit.

6.4.1.2. Мониторинг

На вкладке *Мониторинг* выводится системная информация о состоянии сервисов и программ PT AF в виде графиков, необходимых для контроля производительности системы. На вкладке есть 16 графиков в трех категориях:

- Application Firewall;
- System;
- Databases.

Для изменения параметров отображения графика нажмите на его имя, а затем установите необходимые значения настроек в панели редактирования.

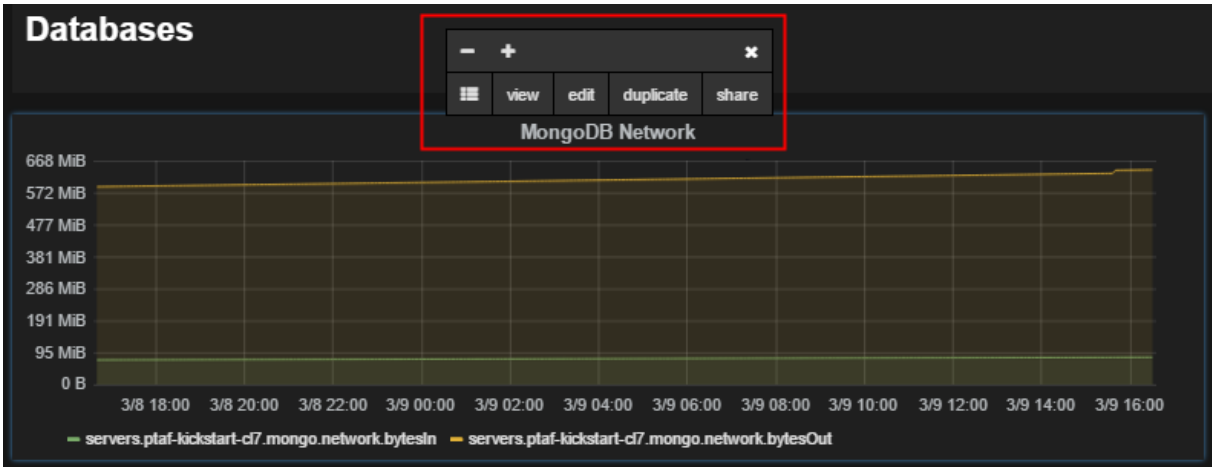


Рис. 121 – Панель редактирования настроек графика

6.4.1.2.1. Application Firewall

К данной категории относятся следующие графики:

- Proxy Requests per second – показывает количество обработанных HTTP-запросов на PT AF в секунду (Requests handled);
- Proxy connections – отображает информацию о соединениях AF:
  - accepted – количество разрешенных соединений;
  - active connections – количество активных соединений;
  - conn_handled – количество обработанных соединений.
- Uptime – показывает время с момента начала работы системы и текущий ее статус.
- CPU – показывает загрузку CPU PT AF'ом с точки зрения monit;
- Memory – показывает расход памяти PT AF'ом с точки зрения monit.

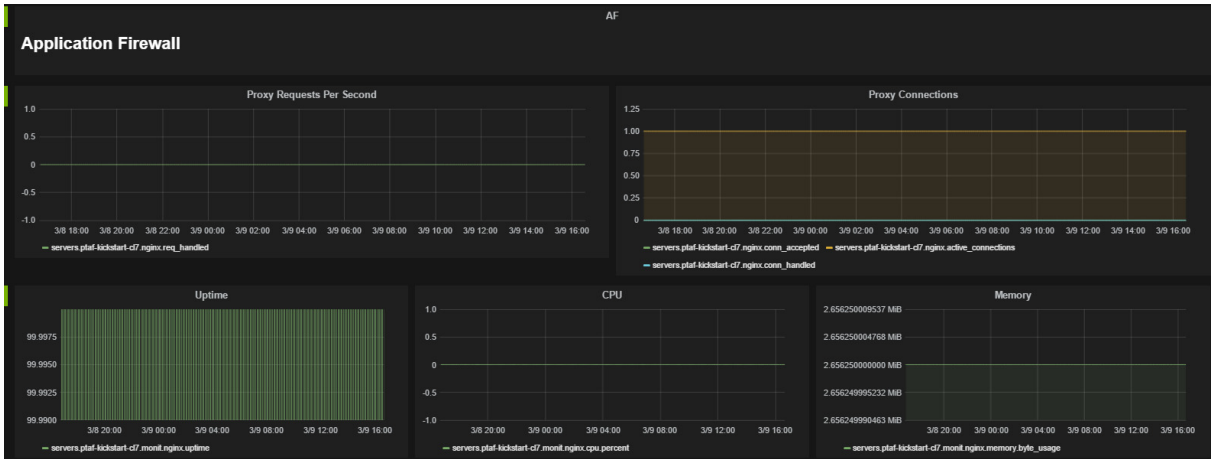


Рис. 122 – Графики группы Application Firewall

6.4.1.2.2. System

К категории *System* относятся следующие графики:

- CPU – показывает загрузку CPU:
  - user – загрузка CPU процессами пользователей системы;
  - system – загрузка CPU процессами ядра;
  - idle – простаивание CPU;
- Memory – отображает загруженность оперативной памяти системой:
  - MemFree – отображает свободную память системы;
  - MemTotal – отображает используемую системой память;
- Disk Space – отображает загруженность памяти жесткого диска:
  - byte_used – отображает размер используемого пространства на диске;
  - byte_available – отображает размер свободного пространства на диске;
- Disk IO – отображает нагрузку на дисковую подсистему:
  - writes_per_second – количество успешных записей на диск в секунду;
  - reads_per_second – количество успешных чтений с диска в секунду;
- Network – отображает количество переданных и полученных данных в секунду для всех активных интерфейсов;
- Processes – отображает активность процессов:
  - proccesses_running – количество текущих активных процессов;
  - proccesses_total – количество запущенных процессов с потоками выполнения.

Рис. 123 – Графики группы System

6.4.1.2.3. Databases

К категории *Databases* относятся следующие графики:

- MongoDB Network – отображает количество данных, пропущенных через сетевой сокет:
  - bytesin – количество полученных данных;
  - bytesout – количество переданных данных;
- MongoDB Operations – количество данных, задействованных при операциях с MongoDB (insert, update, delete etc.);
- Elasticsearch Disk IO – отображает объем записанных и прочитанных из Elasticsearch данных;
- Elasticsearch Memory – использованная Elasticsearch память:
  - resident – память, используемая процессом Elasticsearch;
  - share – объем разделяемой памяти;
- Elasticsearch Access Timings – отображает время, потраченное на доступ к данным в Elasticsearch.

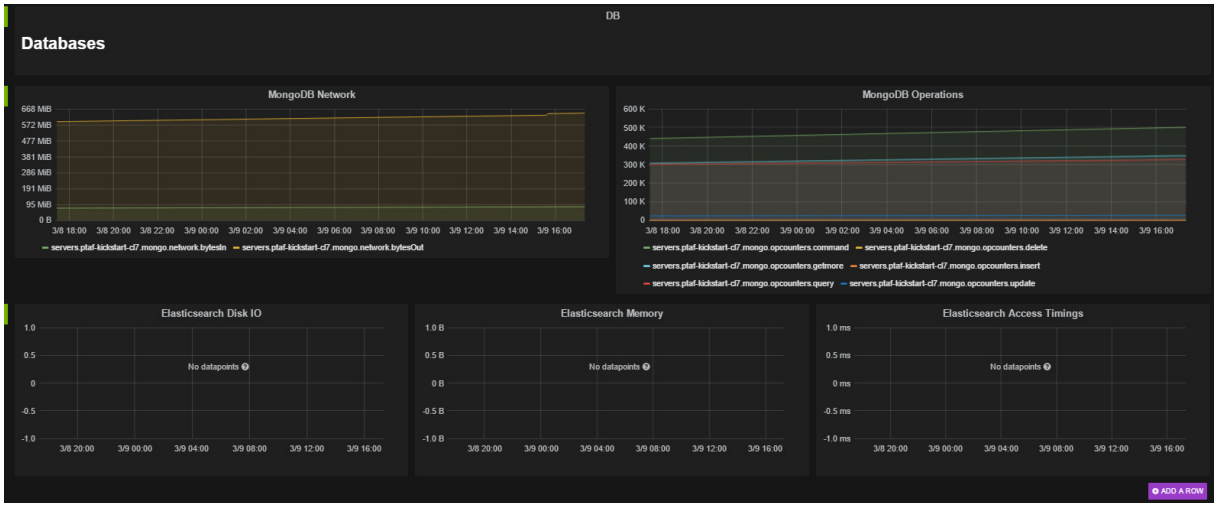


Рис. 124 – Графики группы Databases

6.4.1.3. Services Event Log

При возникновении каких-либо проблем, которые не нарушают работу административного интерфейса, либо для ретроспективного анализа произошедших с системой событий удобно использовать вкладку *Система -> Статус -> Services Event Log*. Вкладка представляет собой сводку со всех наиболее важных файлов журналирования и предоставляет возможности поиска и фильтрации.

Используйте вкладку *Services Event Log*, чтобы получать уведомления об изменении конфигурации, включении/выключении/перезапуске сервисов.

6.4.2. Пользователи

Система PT AF поддерживает многопользовательскую среду. Список команд по управлению и контролю пользователей представлен в Табл. 9.

Таблица 9. Возможности по управлению и контролю пользователей

Команда	Вкладка
Создание нового пользователя	*Система-> Пользователи -> Пользователи*
Изменение пароля пользователя	*Система-> Пользователи -> Пользователи*
Создание новой группы пользователей	*Система-> Пользователи -> Группы пользователей*
Редактирование прав группы пользователей	*Система-> Пользователи -> Группы пользователей*
Просмотр действий пользователя	*Система-> Пользователи -> Журнал действий пользователей*
Синхронизация пользователей с AD	*Система-> Пользователи -> Настройки Active Directory*

Внимание! При редактировании одного объекта несколькими пользователями системой принимаются данные первого, сохранившего объект. Второму при попытке сохранения выдается сообщение о том, что данные были изменены другим пользователем и сохранение невозможно (Рис. 125). Следует нажать кнопку *Открыть новую версию* и заново внести изменения при необходимости.

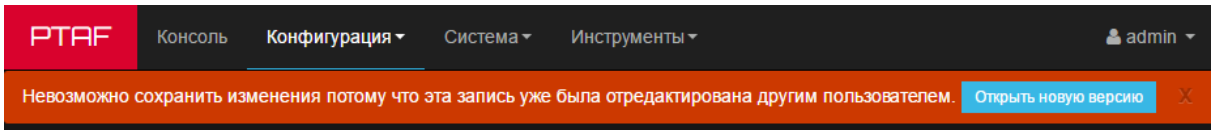


Рис. 125 – Сообщение о редактировании страницы другим пользователем

6.4.2.1. Пользователи

Во вкладке *Пользователи* представлен список пользователей системы, где для каждого пользователя указан логин, Email, группа, статус активности и время последнего изменения.

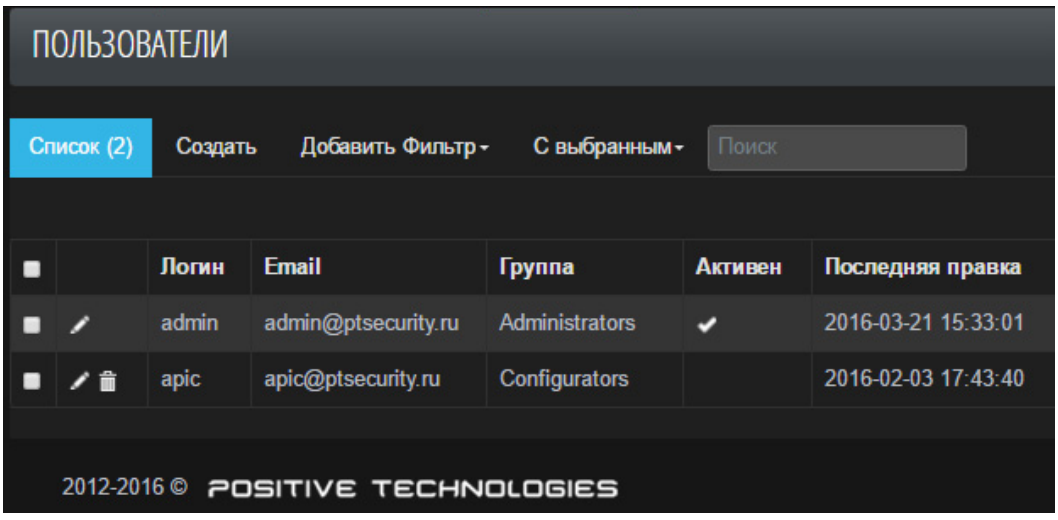


Рис. 126 – Список пользователей

Если пользователь заблокирован, то в строке пользователя появляется значок 🔓. Подробную информацию о блокировке смотри в главе «Web UI Security Settings».

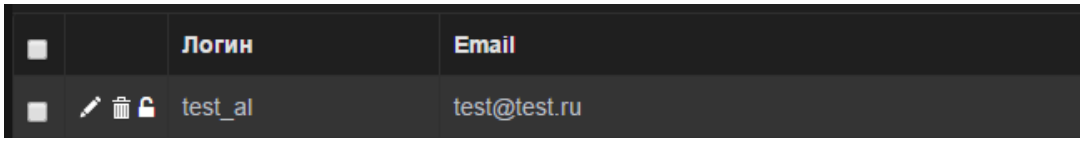


Рис. 127 – Строка заблокированного пользователя

Автоматически в системе создаются два пользователя:

- admin – пользователь с правами администратора системы, который входит в группу Administrators. Данный пользователь необходим для обеспечения доступности системы при любых обстоятельствах;
- apic – пользователь создается и добавляется в группу *Configurators* автоматически, но нуждается в смене пароля и активации для отправки запросов к REST API. Сделать это можно через wsc, используя следующие команды (подробная информация по командам wsc представлена в главе «Настройка сети»):

```
wsc> user password apic
wsc> user activate apic
```

Нажмите кнопку *Создать*, чтобы добавить пользователя в систему, и заполните все поля формы. Затем нажмите кнопку *Сохранить и Добавить*.

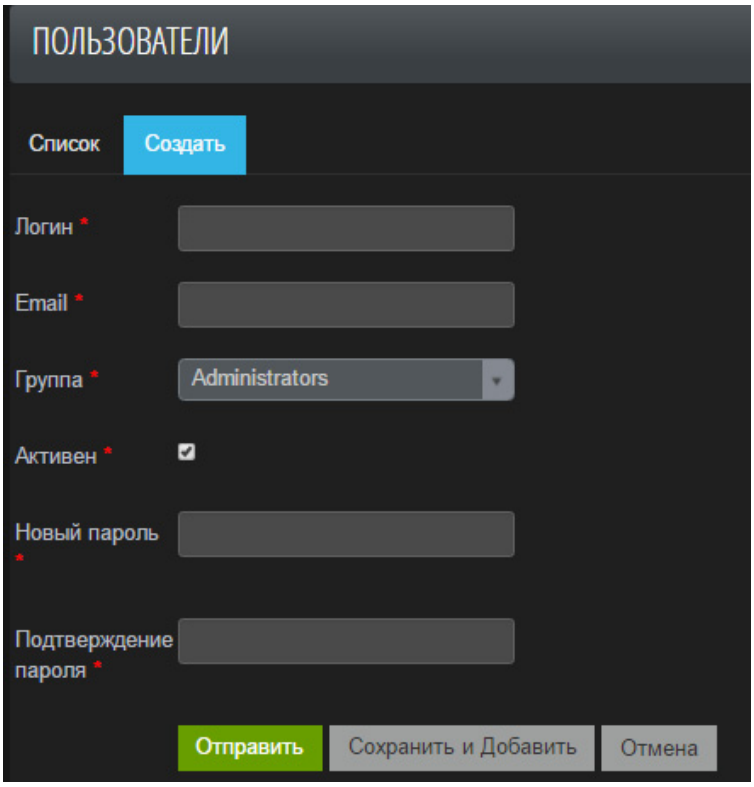


Рис. 128 – Окно создания пользователя в системе

Чтобы изменить данные выбранного пользователя, нажмите кнопку (*Редактировать)* в строке пользователя во вкладке *Пользователи.* В появившемся окне укажите новые данные и нажмите кнопку *Сохранить и продолжить.*

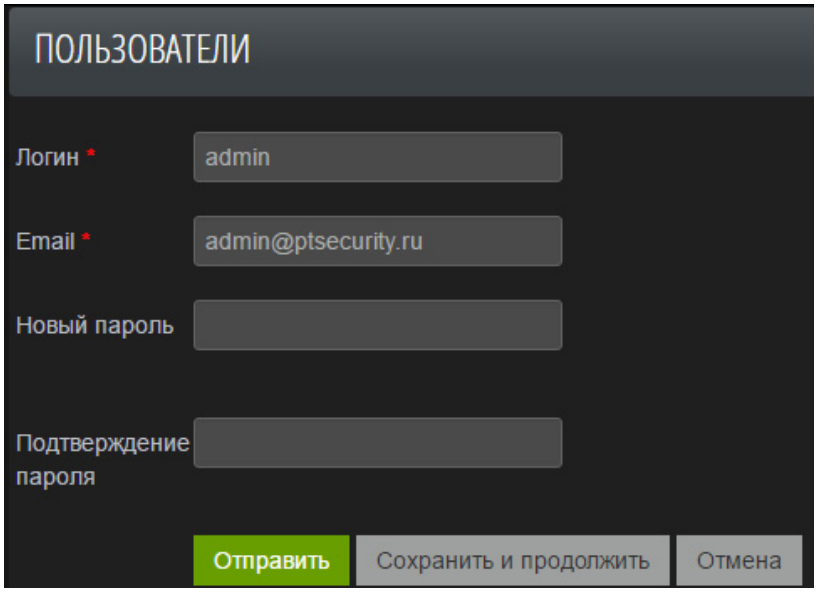


Рис. 129 – Окно редактирования пользователя

6.4.2.1.1. Смена пароля

Текущий пользователь системы может изменить свой пароль. Для этого необходимо нажать на свою учетную запись в правом верхнем углу страницы и выбрать команду *Мой аккаунт*.

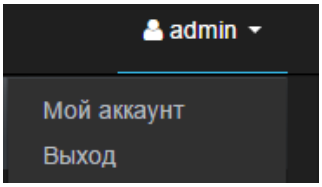


Рис. 130 – Управление аккаунтом

В открывшемся окне заполните все поля формы и нажмите кнопку *Сохранить и продолжить* (Рис. 131).

> **Примечание:** настройки политики безопасности доступны во вкладке *Система -> Web UI Security Settings*.

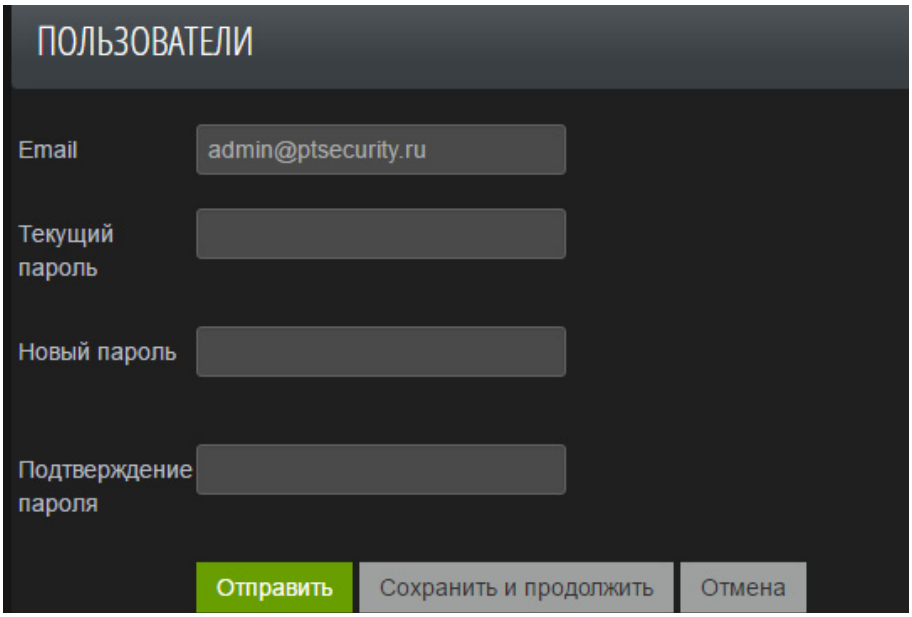


Рис. 131 – Окно изменения пароля

6.4.2.2. Группы пользователей

Во вкладке *Группы пользователей* представлен список групп пользователей системы, где для каждой группы указано имя и время последнего изменения.

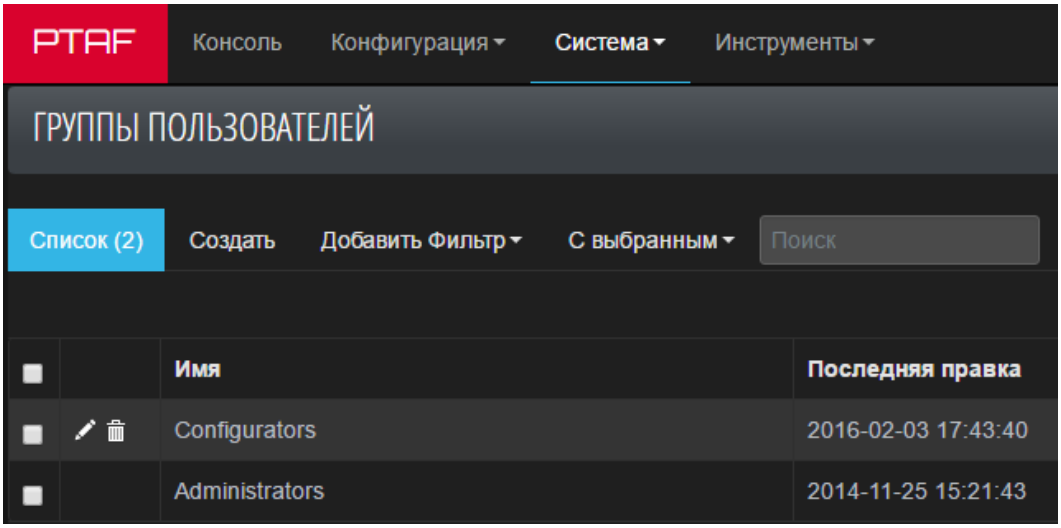


Рис. 132 – Список групп пользователей

В системе предустановлены две группы пользователей:

- Administrators – группа администраторов системы. Данная группа пользователей не подлежит редактированию;
- Configurators – группа пользователей системы, которые могут отправлять запросы к REST API. Данный функционал необходим для интеграции с Cisco ACI или с любой аналогичной системой. Права группы ограничены сетевыми настройками и базовыми настройками профилей и групп серверов.

Нажмите кнопку *Создать*, чтобы добавить группу пользователей в систему. В появившемся окне укажите имя группы и обозначьте список привилегий на функционал системы. Затем нажмите кнопку *Сохранить и Добавить*.

Полный список прав представлен в Табл. 10.

Таблица 10. Права для групп пользователей

Право	Описание	Применимость к функционалу
can_view	Право на просмотр	Консоль, Профили, Правила, Действия, Тэги, HMM-модели, Content Security Policy, Черный список IP-адресов, Черный список хостов, XML-схемы, Политики, Регулярные выражения, Шлюзы, Группа серверов, Файрвол, Сниффер, Алиасы сетевых интерфейсов, Виртуальные IP, События, Корреляции, ICAP-сервисы, LDAP-сервисы, SSL-сертификаты и ключи, Suspicious Sessions, Services Status, Мониторинг, Services event log, Пользователи, Группы пользователей, Журнал действий пользователей, Настройки Active Directory, Настройки обучающего модуля, Web UI Settings, Web UI Security Settings, About, Анализ файлов журналирования, Отчеты, Резервные копии, Расписание резервных копий, IP Whois, Проверка регулярных выражений, Управление обучающим модулем, Виртуальный патчинг
can_scan	Право на сканирование	Консоль
can_exclude	Право на функцию исключения	Консоль
can_block	Право на функцию блокировки	Консоль
can_list	Право на экспорт списка IP-адресов для Arbor через REST API	Файрвол
can_create	Право на создание	Профили, Правила, Действия, Тэги, Content Security Policy, Черный список IP-адресов, Черный список хостов, Политики, Регулярные выражения, Группа серверов, Файрвол, Алиасы сетевых интерфейсов, Виртуальные IP, События, Корреляции, ICAP-сервисы, LDAP-сервисы, Пользователи, Группы пользователей, Настройки Active Directory, Резервные копии, Расписание резервных копий
can_edit	Право на редактирование	Профили, Правила, Действия, Тэги, Content Security Policy, Черный список IP-адресов, Черный список хостов, Политики, Регулярные выражения, Шлюзы, Группа серверов, Файрвол, Алиасы сетевых интерфейсов, Виртуальные IP, События, Корреляции, ICAP-сервисы, LDAP-сервисы, Suspicious Sessions, Пользователи, Группы пользователей, Настройки Active Directory, Расписание резервных копий
can_upload	Право на загрузку в систему	XML-схемы, SSL-сертификаты и ключи, Анализ файлов журналирования, Резервные копии
can_download	Право на выгрузку из системы	XML-схемы, Отчеты, Резервные копии

Таблица 10. Права для групп пользователей

Право	Описание	Применимость к функционалу
can_delete	Право на удаление	Профили, Правила, Действия, Тэги, Content Security Policy, Черный список IP-адресов, Черный список хостов, XML-схемы, Политики, Регулярные выражения, Группа серверов, Файрвол, Алиасы сетевых интерфейсов, Виртуальные IP, События, Корреляции, ICAP-сервисы, LDAP-сервисы, SSL-сертификаты и ключи, Suspicious Sessions, Пользователи, Группы пользователей, Настройки Active Directory, Анализ файлов журналирования, Отчеты, Резервные копии, Расписание резервных копий, Виртуальный патчинг
can_restore	Право на восстановление	Резервные копии.

Чтобы изменить права выбранной группы пользователей, нажмите кнопку ✎ (*Редактировать)* в строке группы во вкладке *Группы пользователей.* В окне редактирования укажите права и нажмите кнопку *Сохранить и продолжить.*

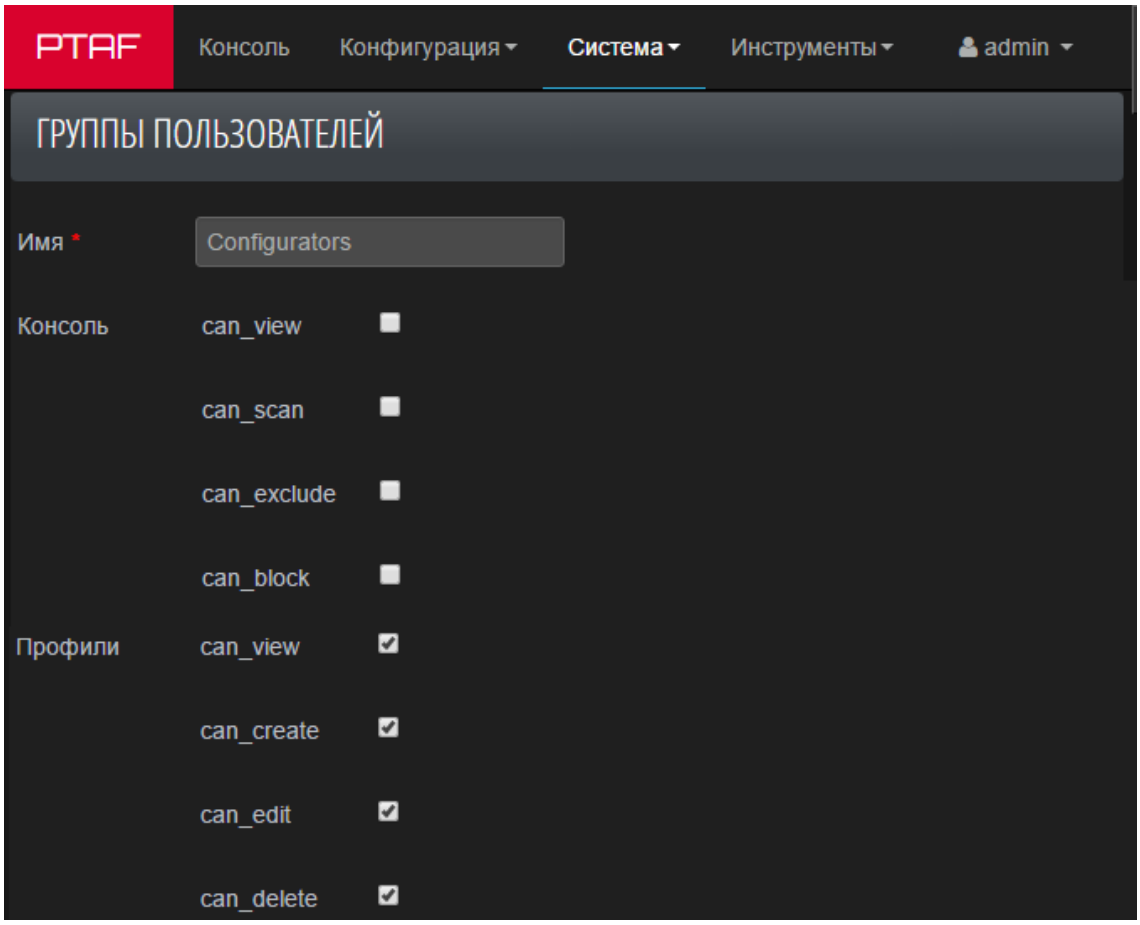
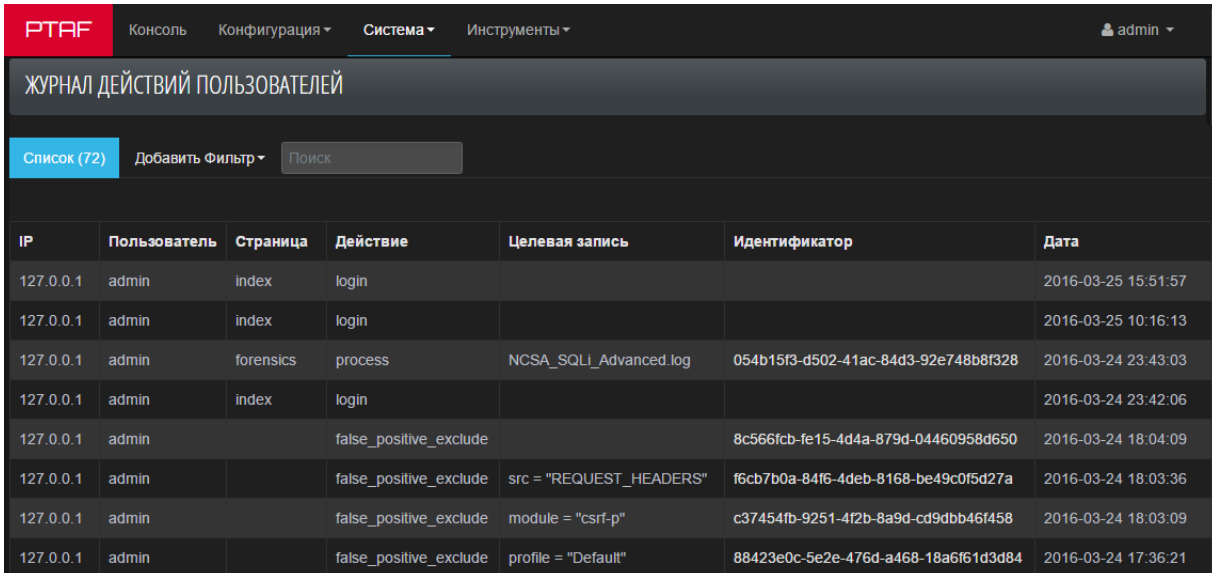


Рис. 133 – Окно редактирования группы пользователей

6.4.2.3. Журнал действий пользователей

Данная вкладка предназначена для мониторинга действий пользователей системы. В журнале указано имя пользователя, IP-адрес, страница, на которой были произведены

изменения, действие, целевая запись, идентификатор, а также дата и время совершения изменений. В левом верхнем углу страницы в поле *Список* указано общее количество записей журнала.



IP	Пользователь	Страница	Действие	Целевая запись	Идентификатор	Дата
127.0.0.1	admin	index	login			2016-03-25 15:51:57
127.0.0.1	admin	index	login			2016-03-25 10:16:13
127.0.0.1	admin	forensics	process	NCSA_SQLi_Advanced.log	054b15f3-d502-41ac-84d3-92e748b8f328	2016-03-24 23:43:03
127.0.0.1	admin	index	login			2016-03-24 23:42:06
127.0.0.1	admin		false_positive_exclude		8c566fcb-fe15-4d4a-879d-04460958d650	2016-03-24 18:04:09
127.0.0.1	admin		false_positive_exclude	src = "REQUEST_HEADERS"	f6cb7b0a-84f6-4deb-8168-be49c0f5d27a	2016-03-24 18:03:36
127.0.0.1	admin		false_positive_exclude	module = "csrf-p"	c37454fb-9251-4f2b-8a9d-cd9dbb46f458	2016-03-24 18:03:09
127.0.0.1	admin		false_positive_exclude	profile = "Default"	88423e0c-5e2e-476d-a468-18a6f61d3d84	2016-03-24 17:36:21

Рис. 134 – Журнал действий пользователей

При необходимости используйте фильтр. Нажмите кнопку *Добавить Фильтр*, в выпадающем списке выберите критерий фильтрации.

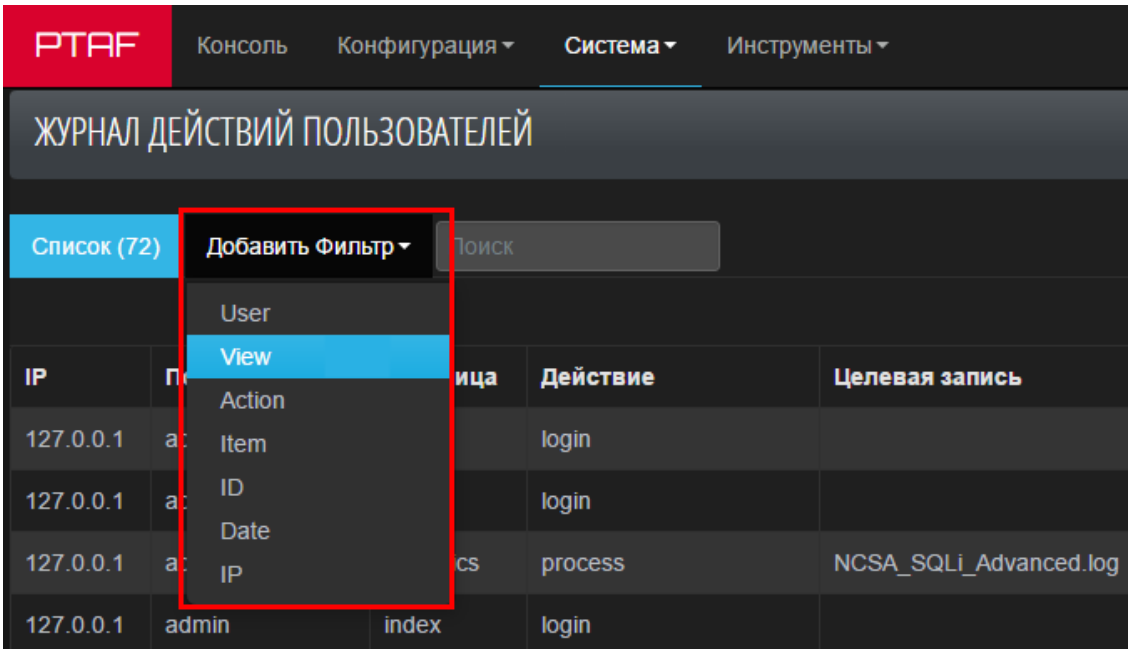


Рис. 135 – Меню добавления фильтра

Укажите значение, по которому следует провести фильтрацию. Нажмите кнопку *Применить*.

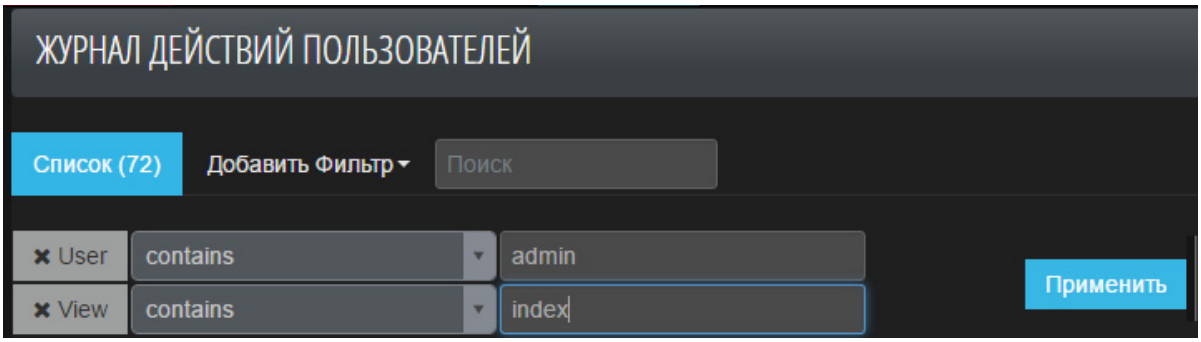


Рис. 136 – Добавление значения в поле фильтра

Чтобы отменить фильтрацию, нажмите кнопку *Сброс фильтров*.

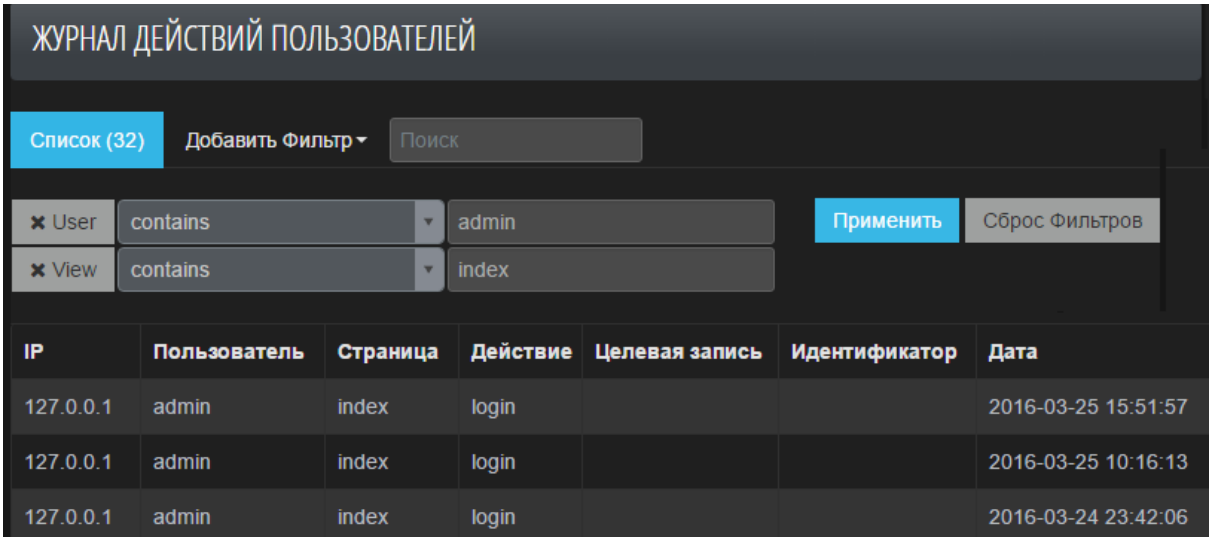


Рис. 137 – Журнал действий пользователей с примененным фильтром

Чтобы найти необходимое значение, в системе предусмотрен текстовый поиск. Введите в поле поиска искомое значение или часть искомого значения. При нажатии клавиши ENTER будут выбраны все значения, соответствующие поисковому запросу. В поле *Список* отразится количество найденных записей, соответствующих поисковому запросу.

Примечание: максимальная длина поискового запроса – 100 символов.

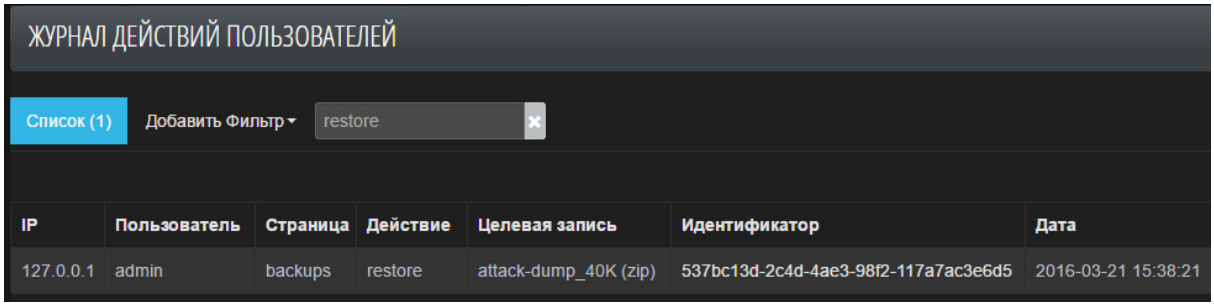


Рис. 138 – Журнал действий пользователей с примененным поисковым запросом

6.4.2.4. Настройки Active Directory

Настройте поля во вкладке *Настройки Active Directory*, чтобы авторизоваться в PT AF с использованием доменных учетных записей.

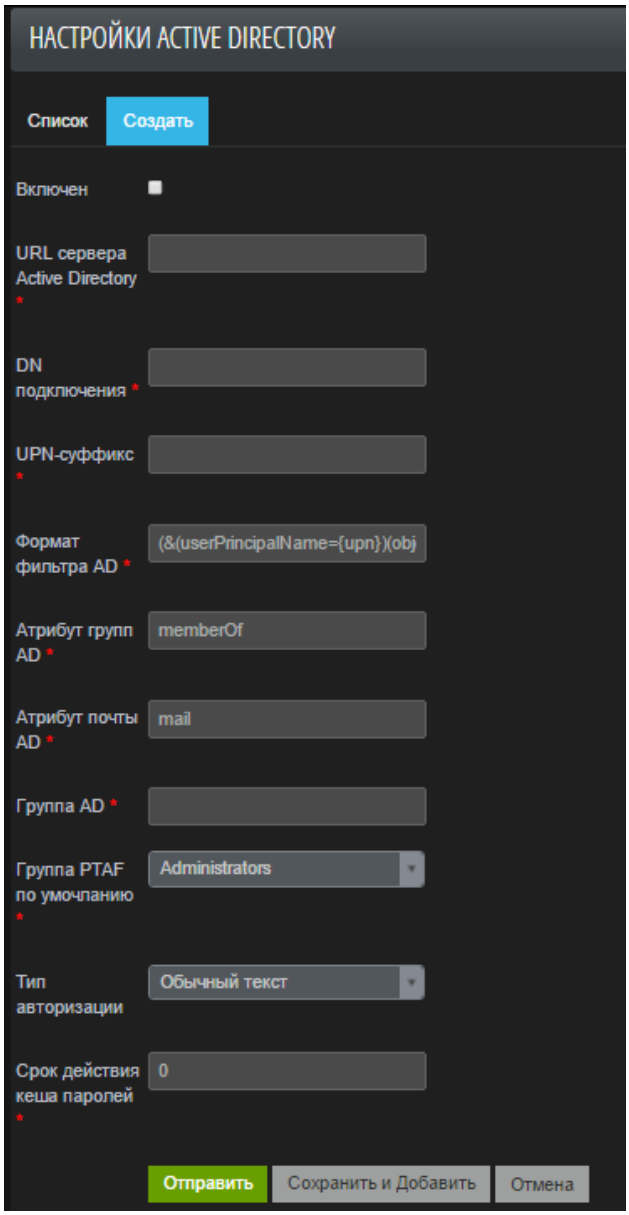


Рис. 139 – Настройки Active Directory

6.4.3. Web UI Settings

Вкладка Web UI Settings представляет собой группу настроек сетевого интерфейса и пользовательского интерфейса самого PT AF: номер порта, IP-адрес, из каких сетей можно подключаться и пр.

> **Внимание!** Неверные значения настроек, установленных на этой странице, могут заблокировать доступ к пользовательскому интерфейсу через браузер.

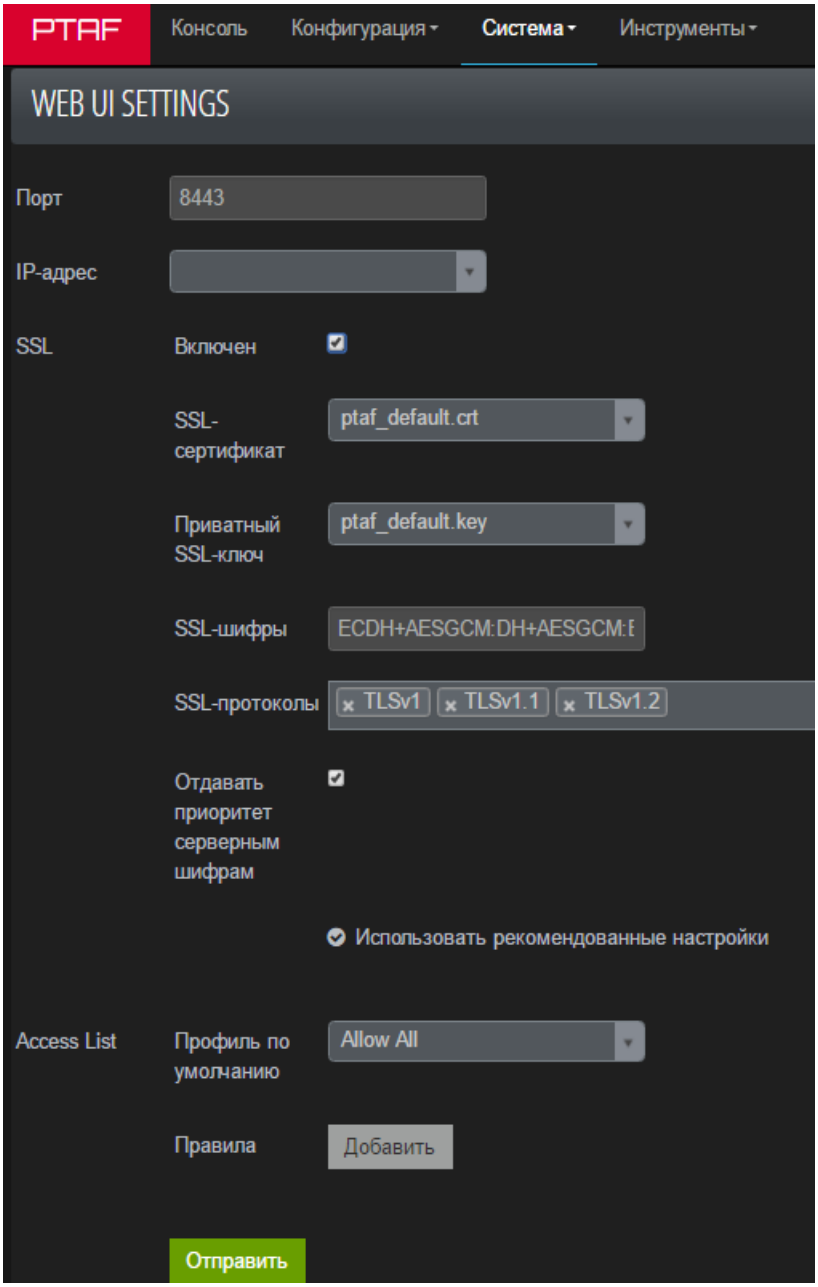


Рис. 140 – Web UI Settings

Настройка *Access List - Default mode* предназначена для управления списком адресов, которым разрешен доступ к интерфейсу:

- Allow All – разрешено всем, кроме адресов, указанных в поле *Правила*;
- Deny all – запрещено всем, кроме адресов, указанных в поле *Правила*.

Примечание: поддерживаются адреса IPv4 и диапазон адресов в формате CIDR.

6.4.4. Web UI Security Settings

В разделе Web UI Security Settings задаются параметры политики безопасности для пользователей (см. Рис. 142).

К группе параметров *Пароль* относятся следующие настройки сложности пароля:

- Minimum length – поле ввода требуемой минимальной длины пароля;
- Lowercase Letters – опция, позволяющая проверять наличие в пароле латинских букв в нижнем регистре;
- Uppercase Letters – опция, позволяющая проверять наличие в пароле латинских букв в верхнем регистре;
- Special chracters – опция, позволяющая проверять наличие в пароле специальных символов (из -+*\=\._!@#%^\$\&());
- Digits – опция, позволяющая проверять наличие в пароле цифр.

Настройки сложности служат правилами для проверки пароля при редактирования учетной записи текущего пользователя (см. главу «Смена пароля») и при редактировании учетных записей пользователей администратором (см. главу «Пользователи», Рис. 141).

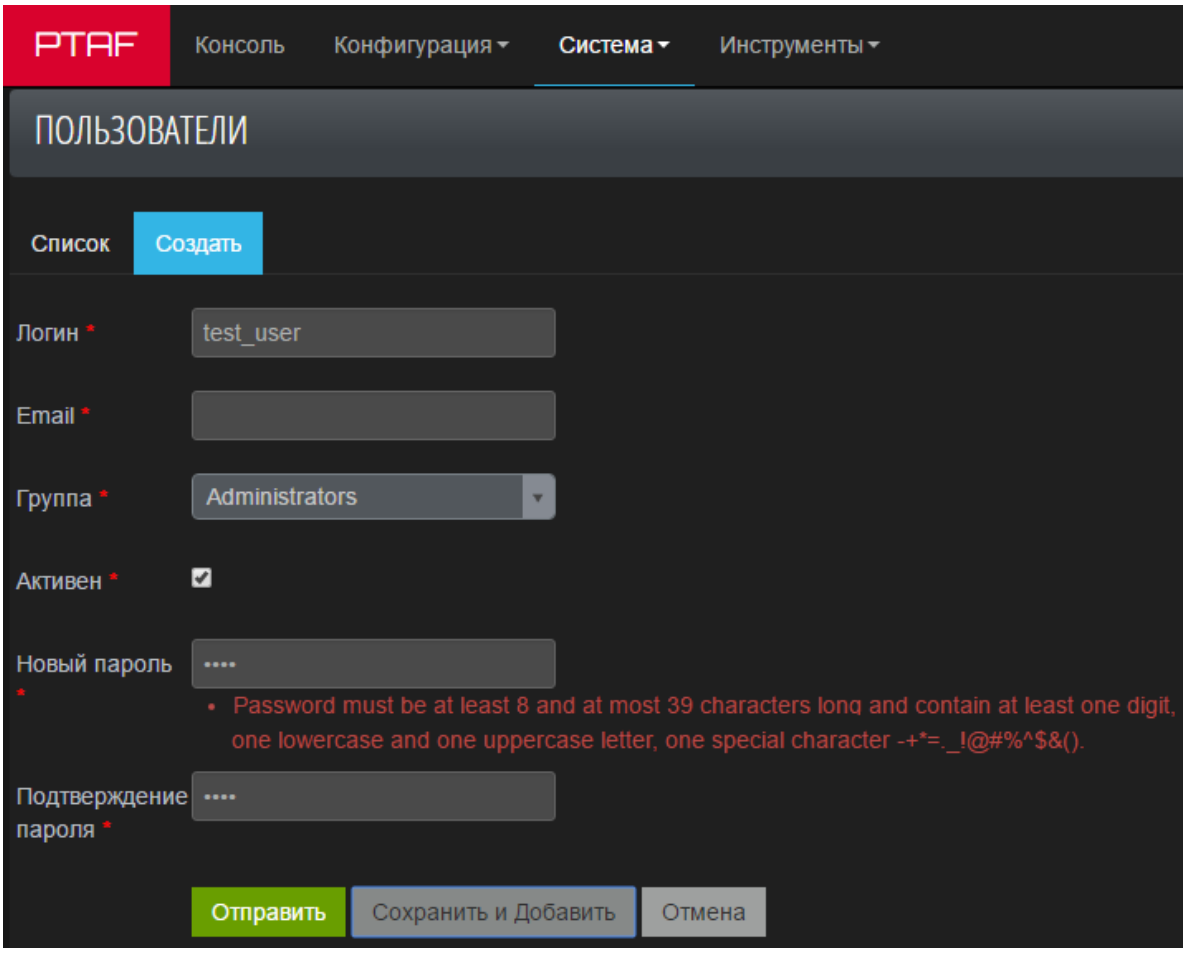


Рис. 141 – Сообщение о неправильном вводе пароля

Настройка *Длительность* позволяет установить срок действия пароля: продолжительность и тип (месяцы, недели, дни) периода действия пароля.

> **Примечание.** При истечении срока действия пароля все запросы пользователя перенаправляются на страницу учетной записи пользователя с предложением изменить пароль. Перенаправление будет осуществляется до тех пор, пока не будет введен новый, отличающийся от старого, пароль.

К группе настроек *Account lock* относятся настройки блокировки учетной записи:

- Включен – включение блокировки учетной записи;
- Allowed login attempts – число допустимых попыток ввода имени пользователя перед блокированием учетной записи. Счетчик попыток привязан к имени пользователя и его IP-адресу;
- Lock period – продолжительность и тип (секунды, минуты, часы) периода блокировки. Продолжительность с нулевым значением означает постоянную блокировку.

Примечание. Если последняя из допустимого числа попытка входа в систему завершается неудачей, то на экране отображается сообщение о блокировке учетной записи и указывается время, оставшееся до разблокирования. Все последующие запросы на вход в систему в период блокировки автоматически завершаются неудачей с сообщением времени, оставшимся до разблокирования. Если блокирование постоянное, то оставшееся до разблокирования время не указывается, и вновь сделать попытку входа в систему можно будет только при изменении политики блокирования администратором.

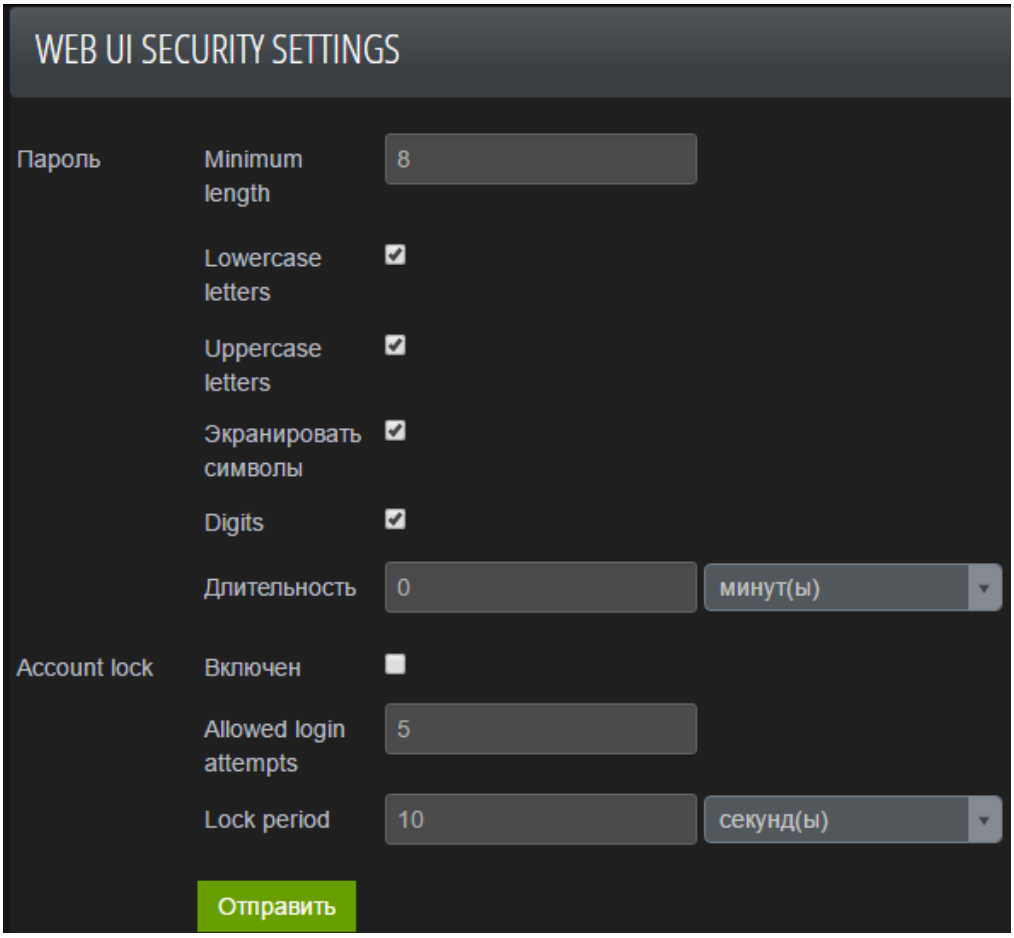


Рис. 142 – Параметры политики безопасности для пользователей

Если пользователь заблокирован, то во вкладке *Пользователи* в строке заблокированного пользователя появляется значок 🔓. Для оперативной разблокировки пользователя нажмите значок 🔓.

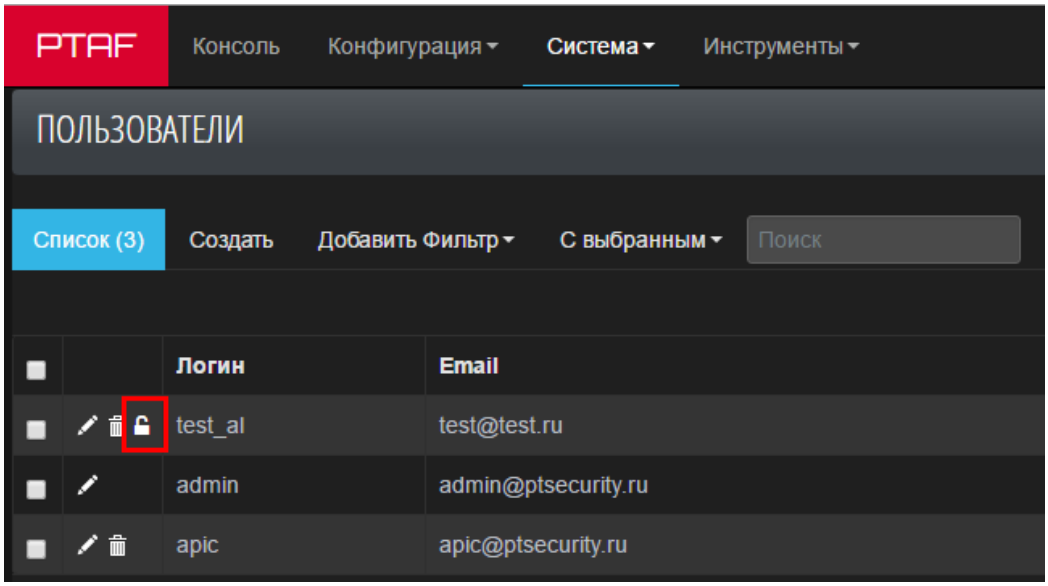


Рис. 143 – Список с заблокированным пользователем

6.4.5. Настройки обучающего модуля

Данная вкладка позволяет задать настройки обучающего модуля для работы HMM. На вкладке представлены следующие настройки:

- Включать HMM-модели по окончании обучения – настройка, позволяющая включить модель после обучения;
- Welford HMM Factor – коэффициент порога лояльности модели по отношению к отклонениям от обучающей выборки. Чем выше коэффициент, тем модель более лояльна к трафику;
- Минимальное количество образцов для обучения – минимальное количество запросов для добавления модели. По умолчанию выставлено значение 500. Это ограничение комбинируется с настройкой *Коэффициент атак*. Два условия объединены логическим оператором И. Во вкладке *Конфигурация -> Политики безопасности -> HMM-модели* отображаются только стабильные обученные модели;
- Коэффициент атак – процент ошибочных запросов (в долях от единицы);
- Workers – количество потоков обучения, определяет период обучения. Чем больше количество потоков, тем больше ресурсов системы тратится на математические обсчеты данных в трафике. Чем меньше значение, тем дольше будет проходить обучение модели.

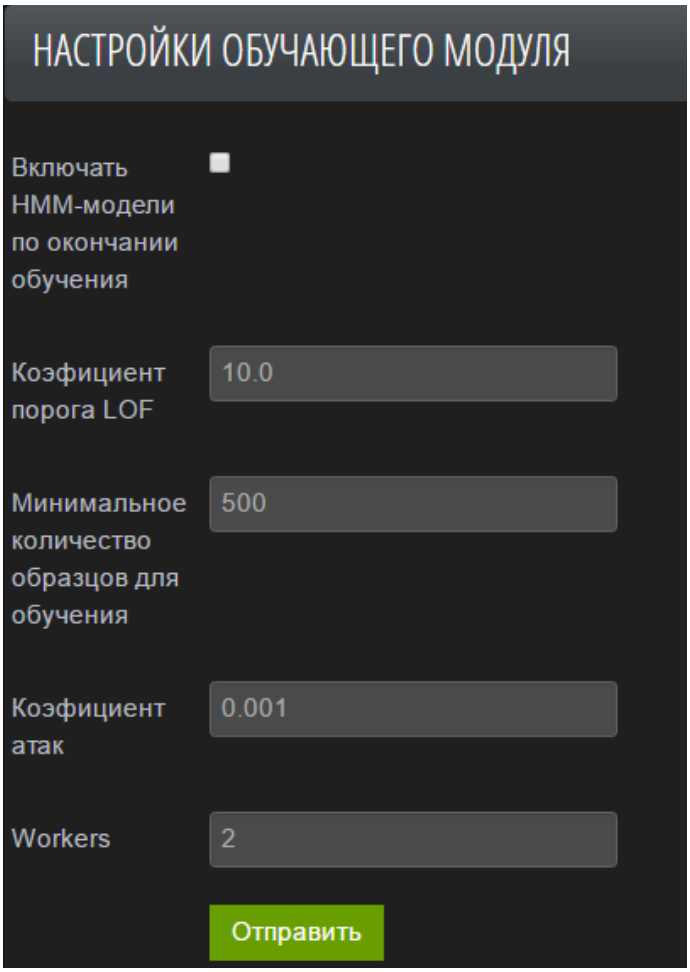


Рис. 144 – Окно настроек обучающего модуля

6.4.6. About

Откройте вкладку *About*, чтобы получить информацию о лицензии на систему PT AF.

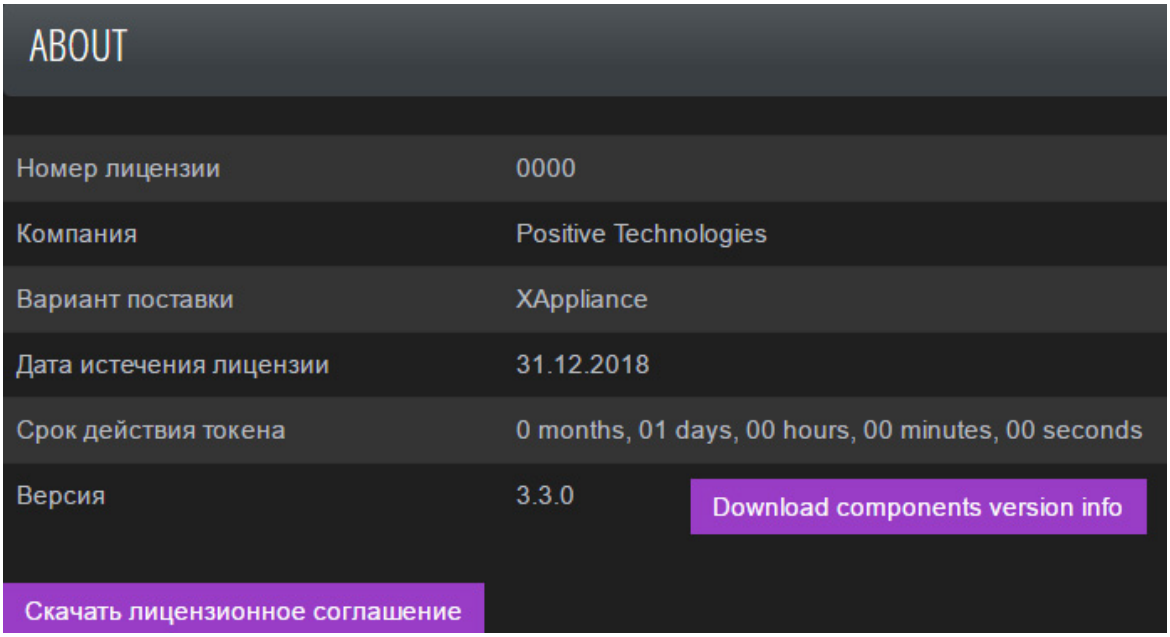


Рис. 145 – Лицензия

Если лицензия не загружена, то необходимо принять условия использования и загрузить лицензионное соглашение, страница будет выглядеть следующим образом:

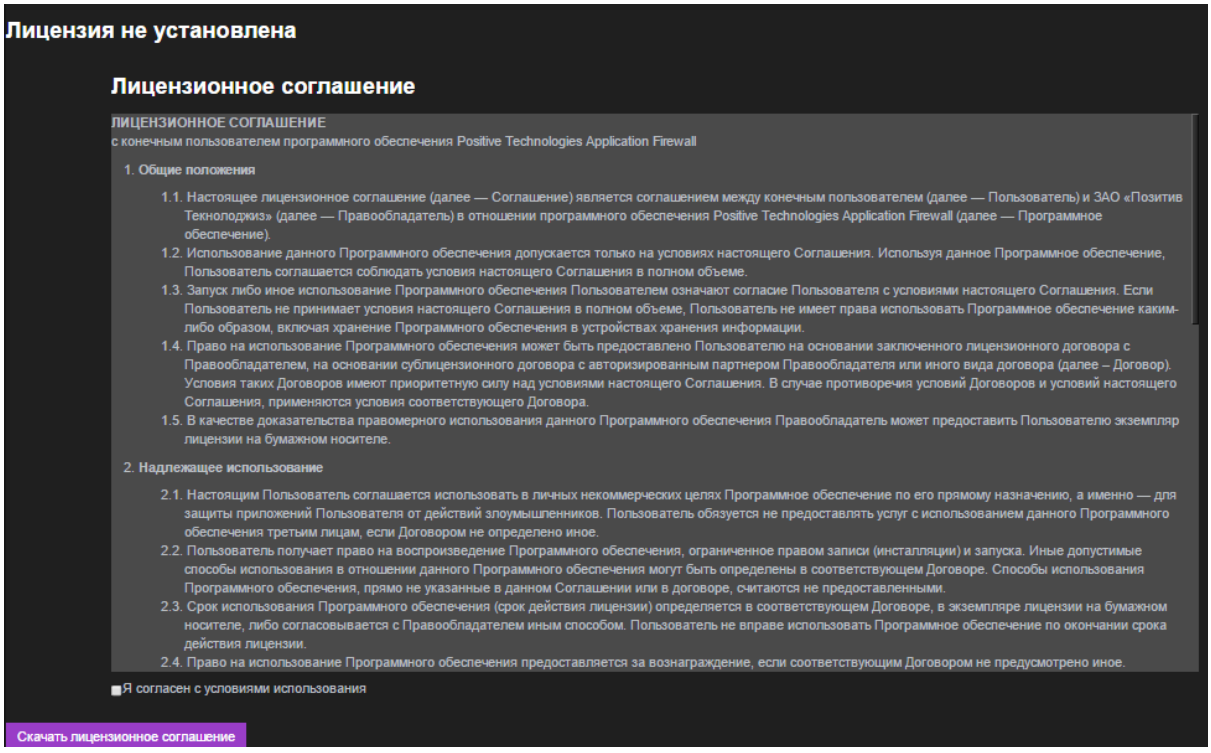


Рис. 146 – Лицензионное соглашение

Примите условия использования, включив опцию *Я согласен с условиями использования*. При необходимости нажмите кнопку *Скачать лицензионное соглашение*. Затем нажмите кнопку *Загрузить файл лицензии* и выберите требуемый системой файл.

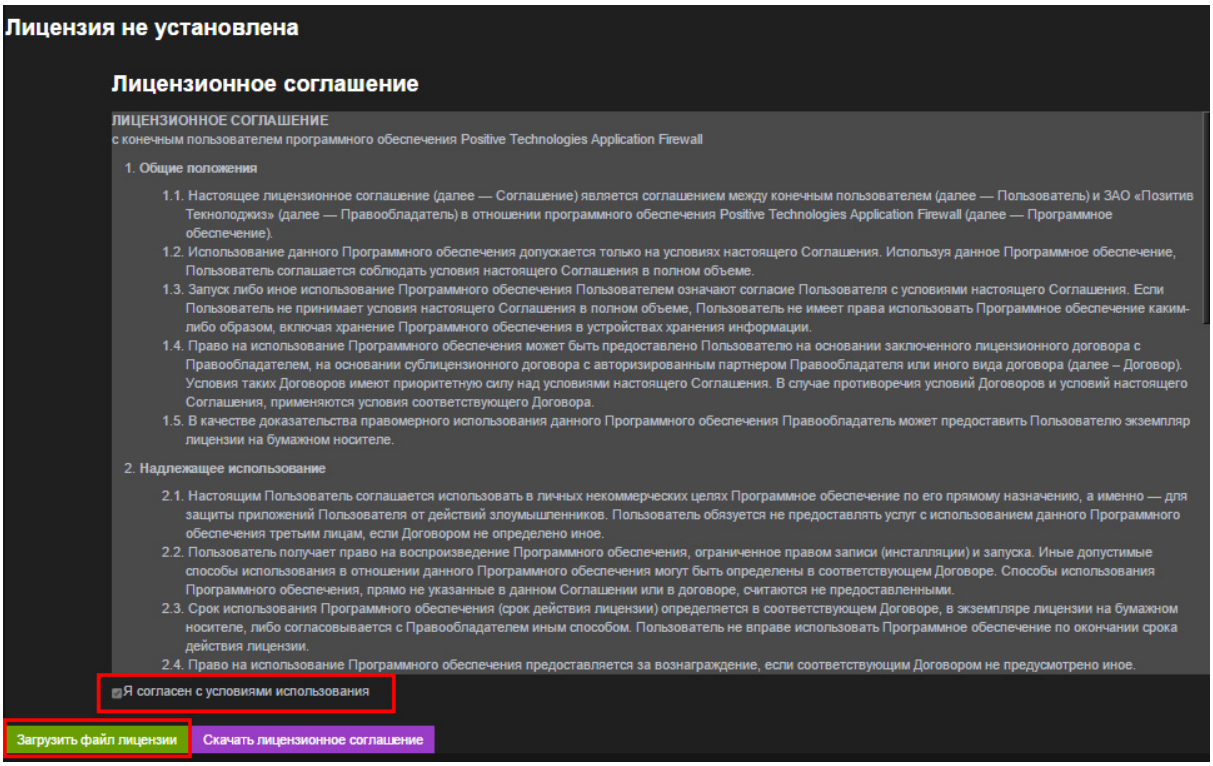


Рис. 147 – Согласие с условиями использования лицензионного соглашения

6.5. Инструменты

В состав инструментов PT AF входит набор вспомогательных средств:

- Анализ файлов журналирования;
- Отчеты;
- Резервные копии;
- Расписание резервных копий;
- IP Whois;
- Проверка регулярных выражений;
- Управление обучающим модулем;
- Виртуальный патчинг.

6.5.1. Анализ файлов журналирования

Вкладка *Анализ файлов журналирования (Forensics)* позволяет загружать журналы веб-серверов для дальнейшего изучения зарегистрированных в журнале инцидентов.

Во вкладке размещен список загруженных файлов, где для каждого файла указаны следующие параметры (Рис. 148):

• Имя – имя файла;

• Формат – формат файла. Может принимать значение *Unknown*, если формат не определен, и значение *Определить*, если не было совершено действие *Определить формат*;

• Размер – размер файла;

• Статус – статус обработки файла. Может принимать следующие значения:

- STOPPED, если обработка была принудительно завершена;
- UNPROCESSABLE, если обработка невозможна;
- UPLOADING, если происходит обработка файла;
- PENDING, если была запущена команда *Обработать*;
- PROCESSED, если обработка файла была успешно завершена;
- FAILED, если обработка файла прошла безуспешно.

• Идентификатор задания – номер задания, выданный после успешной обработки файла, со ссылкой на вкладку *Консоль* для анализа атак, зафиксированных в файле журнала.

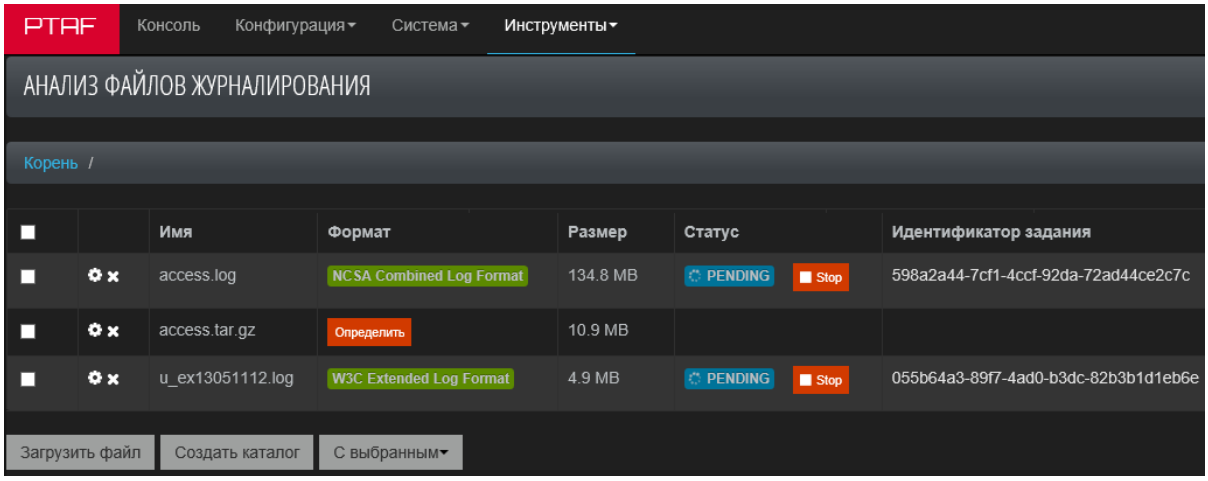


Рис. 148 – Анализ файлов журналирования

Для анализа нового файла журналирования нажмите кнопку *Загрузить файл*, в новом окне выберите файл для загрузки и нажмите кнопку *Отправить*.

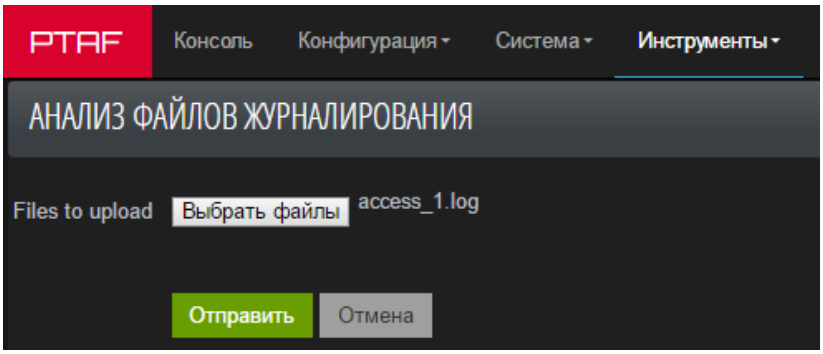


Рис. 149 – Окно загрузки файла

После загрузки файла происходит этап определения формата файла, нажмите кнопку *Определить* в поле *Формат*.

Если файл не определен, то можно указать формат вручную, нажав значок ⚙ в строке файла. В новом окне следует указать формат файла в соответствии с его типом. Возможные типы файлов журналирования:

- NCSA Common: https://httpd.apache.org/docs/trunk/logs.html#common
- W3C Extended: https://www.w3.org/TR/WD-logfile

Затем требуется нажать кнопку *Проверить формат*.

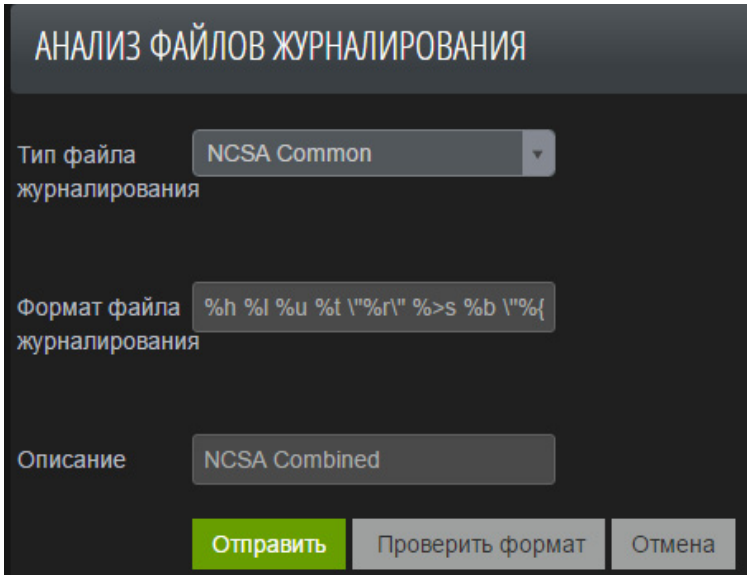


Рис. 150 – Окно для ручного ввода формата файла

Если файл определен, то выполните действие *Обработать* (Рис. 151), произойдет обработка файла журнала с загрузкой атак в Elasticsearch.

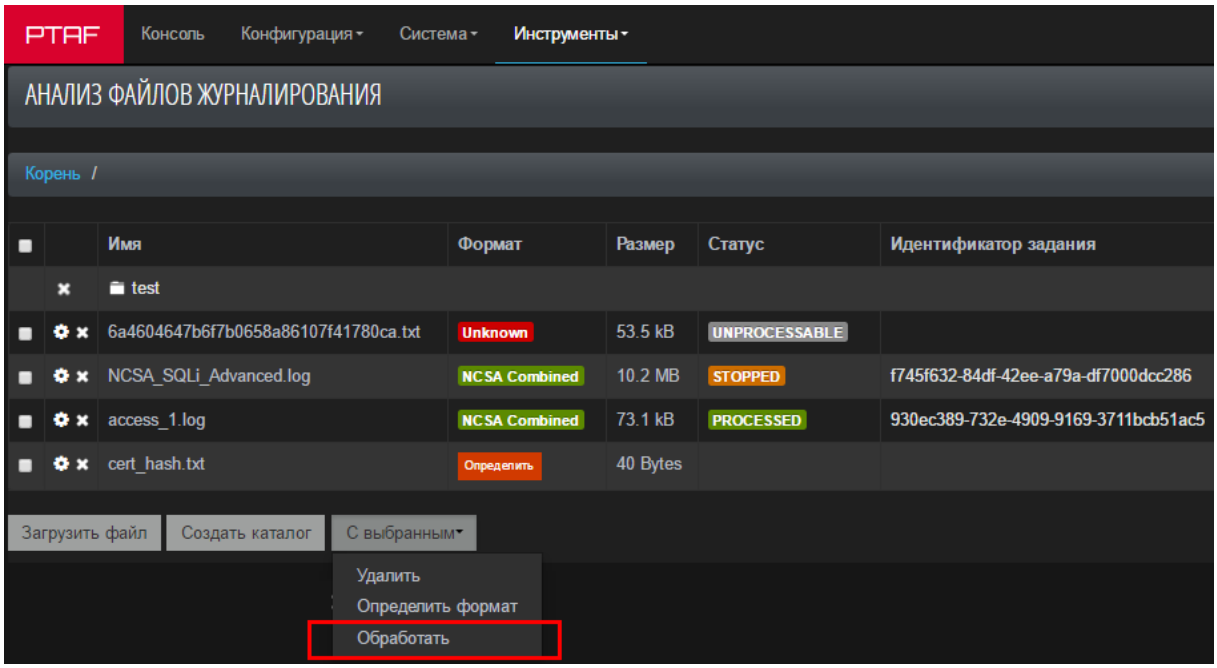


Рис. 151 – Список файлов журналирования

После успешной загрузки внешнего журнала нажмите ссылку в столбце *Идентификатор задания*, откроется новая вкладка с консолью, где можно будет изучать события данного журнала.

6.5.2. Отчеты

Вкладка *Отчеты* позволяет просматривать список выгруженных отчетов. Информация по генерации отчетов представлена в главе «Генератор отчетов». Нажмите на имя отчета, чтобы сохранить его на диске.

Отчет может быть создан в формате CSV (в zip-архиве), PDF, DOC или ODT.

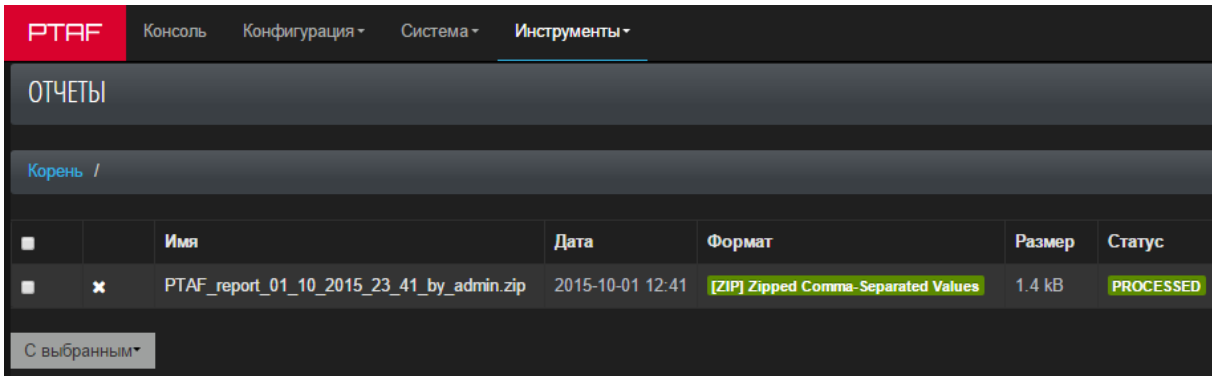


Рис. 152 – Отчеты

6.5.3. IP Whois

Инструмент позволяет получить регистрационную информацию о владельце IP-адреса, для этого введите IP-адрес в строку ввода и нажмите кнопку *Whois*.

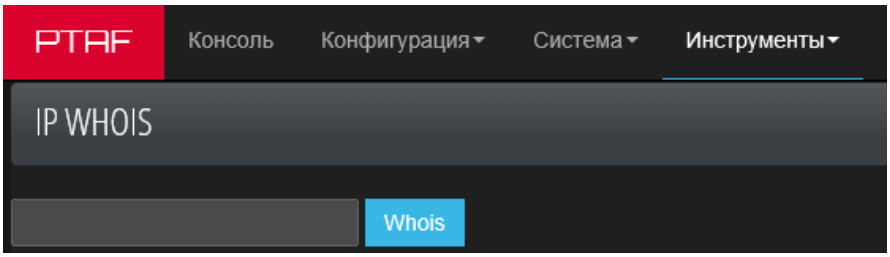


Рис. 153 – Поиск информации по IP-адресу

6.5.4. Проверка регулярных выражений

Инструмент отладки регулярных выражений. Может использоваться для написания новых правил в модуле *Правила*.

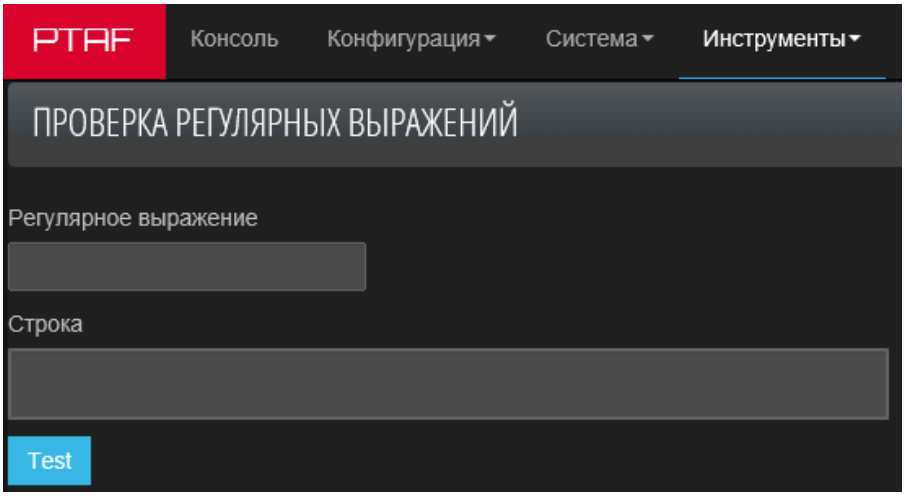


Рис. 154 – Проверка регулярных выражений

6.5.5. Управление обучающим модулем

Инструмент проверки качества работы модели HMM. Также позволяет обучить модель на примерах. Более подробно об обучающем модуле рассказано в главе «Модуль HMM».

Для управления тренером используется протокол JSON-RPC 2.0, в качестве транспорта - HTTP POST. Отправка запроса производится на порт 4000, путь "/", "Content-Type" должен иметь значение "application/json".

Чтобы отправить команду HMM-тренеру, в поле *Команда* укажите необходимую команду, а затем нажмите кнопку *Send* (Рис. 155).

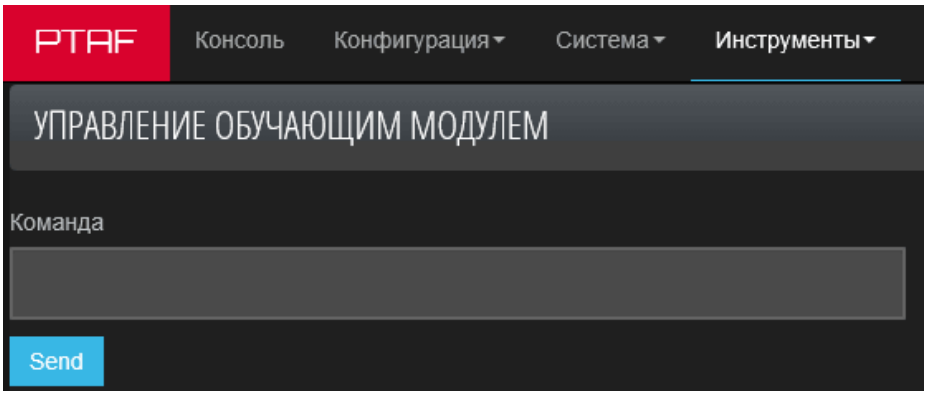


Рис. 155 – Управление обучающим модулем

6.5.5.1. Примеры запросов

Пример правильного запроса:

```
POST / HTTP/1.1
Host: localhost:4000
Content-Type: application/json
Content-Length: 99
Connection: close

{
    "jsonrpc": "2.0",
    "method": "enabled_stab_models",
    "params": true,
    "id": 1
}
```

Пример положительного результата выполнения команды:

```
HTTP/1.1 200 OK
Content-Type: application/json
Content-Length: 63
Connection: close

{
   "id" : 1,
   "jsonrpc" : "2.0",
   "result" : "success"
}
```

Пример запроса, содержащего ошибку:

```
POST / HTTP/1.1
Host: localhost:4000
Content-Type: application/json
Content-Length: 81
Connection: close

{
    "jsonrpc": "2.0",
    "method": "blah",
```

```
    "params": 1,
    "id": 1
}
```

Пример сообщения об ошибке:

```
HTTP/1.1 200 OK
Content-Type: application/json
Content-Length: 119
Connection: close

{
   "error" : {
      "code" : -32601,
      "message" : "Method not found"
   },
   "id" : 1,
   "jsonrpc" : "2.0"
}
```

6.5.5.2. Команды

Команда train

Создает модель и записывает результат в базу. Команда выполняется синхронно, т.е. после получения json-ответа результат будет в базе.

Параметры:

- model - имя модели в соответствии с принятым форматом;
- iter - количество итераций, используемых для тренировки;
- seq - массив образцов, пример ["aa", "aaa", "aaaa", "aaaaa"];
- enable - необязательный параметр, который указывает, включать ли модель после тренировки. Если параметр не указан, берется значение enabled_stab_models.

Пример:

```
{
    "jsonrpc": "2.0",
    "method": "train",
    "params": {"model": "{\"param_name\":\"param\",\"path\":\"/
    \",\"profile_id\":\"528e2758cd80bc1b8633f863\",\"reqvar_type\":\"
    REQUEST_GET_ARGS\"}", "iter": 10000, "seq": ["aa", "aaa", "aaaa",
    "aaaaa"]},
    "id": 1
}
```

В качестве параметра допускается передача массива объектов с описанием задач на тренировку. В этом случае будет возвращен массив с результатами создания каждой модели.

Пример:

```
--> {
    "jsonrpc": "2.0",
    "method": "train",
    "params": [
```

```
            {"model": "{\"param_name\":\"param\",\"path\":\"/a/
            \",\"profile_id\":\"528e2758cd80bc1b8633f863\",\"reqvar_type\
            ":\"REQUEST_GET_ARGS\"}", "iter": 10000, "seq": ["aa", "aaa",
            "aaaa", "aaaaa"]},
            {"model": "{\"param_name\":\"param\",\"path\":\"/b/
            \",\"profile_id\":\"528e2758cd80bc1b8633f863\",\"reqvar_type\
            ":\"REQUEST_GET_ARGS\"}", "iter": 10000, "seq": ["aa", "aaa",
            "aaaa", "aaaaa"]},
            {"model": "{\"param_name\":\"param\",\"path\":\"/c/
            \",\"profile_id\":\"528e2758cd80bc1b8633f863\",\"reqvar_type\
            ":\"REQUEST_GET_ARGS\"}", "iter": 10000, "seq": ["aa", "aaa",
            "aaaa", "aaaaa"]}
        ],
        "id": 1
    }

    <-- {
       "id" : 1,
       "jsonrpc" : "2.0",
       "result" : [ "success", "success", "success" ]
    }
```

Команда start_train

Данная команда аналогична команде train, но запускается в отдельном потоке и сразу же возвращает результат «success». Одновременно может быть выполнена одна асинхронная задача. Если уже выполняется другая задача, вызов блокируется до ее завершения.

Команда enabled_stab_models

Устанавливает статус активности модели после стабилизации - включена/выключена. В качестве параметра передается «true/false». Не затрагивает уже существующие модели.

Пример:

```
    {
        "jsonrpc": "2.0",
        "method": "enabled_stab_models",
        "params": true,
        "id": 1
    }
```

Команда switch_model

Включает/выключает модель.

Параметры:

- model - имя модели;
- enable - статус «true/false»;

Пример:

```
    {
        "jsonrpc": "2.0",
        "method": "switch_model",
        "params": { "model": "{\"param_name\":\"param\",\"path\":\"/
        \",\"profile_id\":\"528e2758cd80bc1b8633f863\",\"reqvar_type\":\"
        REQUEST_GET_ARGS\"}", "enable": false },
```

```
    "id": 1
}
```

Команда get_loglik

Возвращает логарифм вероятности для указанной модели.

Параметры:

- model - имя модели;
- seq - необязательный параметр. Если параметр указан, команда возвращает значение для данного образца (т.е. показывает насколько образец соответствует модели). Если параметр не указан, выводится параметр самой модели, т.е показывает насколько хорошо модель объясняет исходные данные, на которых выполнялась тренировка.

Пример:

```
{
    "jsonrpc": "2.0",
    "method": "get_loglik",
    "params": { "model": "{\"param_name\":\"email\",\"path\":\"/path/
    resource\",\"profile_id\":\"528e2758cd80bc1b8633f863\",\"reqvar_t
    ype\":\"REQUEST_GET_ARGS\"}", "seq": "aaaaa"},
    "id": 1
}
```

Команда load

Импортирует модели и записывает их в базу.

Параметры:

- path - путь до каталога, в который буду выгружены файлы с моделями.

Команда unload

Выгружает все модели в указанный каталог. Имя каждого файла – это md5 от имени модели.

Параметры:

- path - путь до существующего каталога, из которого буду загружены файлы с моделями.

Пример:

```
{
    "jsonrpc": "2.0",
    "method": "unload",
    "params": {"path": "/home/pt/hmm/"},
    "id": 1
}
```

6.5.5.3. Отправка запроса с помощью curl

Для отправки запроса с помощью curl необходимо создать файл test.json с нужным содержимым, а затем из консоли выполнить следующую команду:

```
curl --verbose -H "Content-Type: application/json" --data @test.json
http://ptaf.example.ru:4000
```

6.5.5.4. Создание HMM-модели

Чтобы создать HMM-модель через графический интерфейс PT AF необходимо выполнить следующие шаги:

1. Открыть вкладку *Инструменты -> Управление обучающим модулем*;
2. В поле *Команда* вставить команду создания модели (см. главу «Примеры команд для создания HMM-модели»);
3. Нажать кнопку *Send*. На экране появится сообщение об успешном создании модели;

Рис. 156 – Сообщение об успешно созданной модели

4. Для проверки зайти на вкладку *Конфигурация -> Политики безопасности -> HMM-модели* и увидеть свою модель.

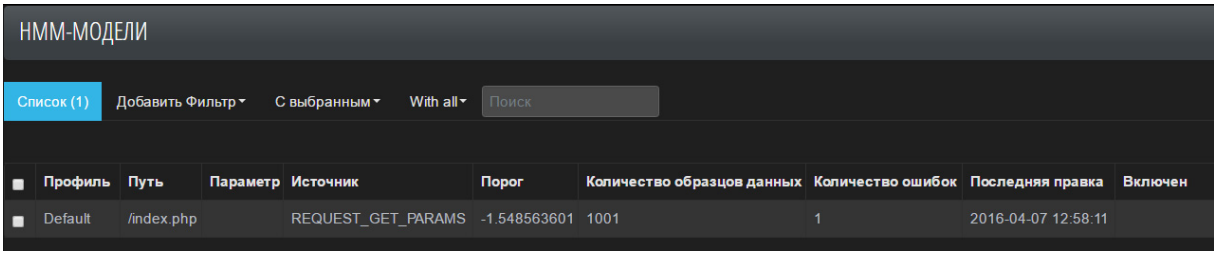


Рис. 157 – Список HMM-моделей

5. Чтобы проверить работу модели в модуле защиты HMM, следует отправлять соответствующие HTTP-запросы на PT AF.

6.5.5.4.1. Примеры команд для создания HMM-модели

В примерах ниже все модели создаются с целью пропускания в параметре с именем «param» символьных значений. Набор пропускаемых и блокируемых значений одинаков для всех моделей, меняется только канал (GET, POST, COKKIE и т.п.).

Пример 1

```
{
    "jsonrpc": "2.0",
    "method": "train",
    "params": {"model": "{\"param_name\":\"param\",\"path\":\"/safe/
\",\"profile_id\":\"528e2758cd80bc1b8633f863\",\"reqvar_type\":\"REQU
EST_GET_ARGS\"}", "iter": 10000, "seq": ["aa", "aaa", "aaaa",
"aaaaa"]},
    "id": 1
}
```

Пропускает запросы следующего вида:

```
GET /?param=abc HTTP/1.1
Host: wafcpp-debian-cl5.rd.ptsecurity.ru:8080
<No body>
GET /?param=abcd HTTP/1.1
Host: wafcpp-debian-cl5.rd.ptsecurity.ru:8080
<No body>
```

Блокирует запросы следующего вида:

```
GET /?param=1 HTTP/1.1
Host: wafcpp-debian-cl5.rd.ptsecurity.ru:8080
<No body>
```

Пример 2

```
{
    "jsonrpc": "2.0",
    "method": "train",
    "params": {"model": "{\"param_name\":\"param\",\"path\":\"/safe/
\",\"profile_id\":\"528e2758cd80bc1b8633f863\",\"reqvar_type\":\"REQU
EST_POST_ARGS\"}", "iter": 10000, "seq": ["aa", "aaa", "aaaa",
"aaaaa"]},
    "id": 1
}
```

Пропускает запросы следующего вида:

```
POST / HTTP/1.1
Host: wafcpp-debian-cl5.rd.ptsecurity.ru:8080
Content-Type: application/x-www-form-urlencoded
param=abc
POST / HTTP/1.1
Host: wafcpp-debian-cl5.rd.ptsecurity.ru:8080
Content-Type: application/x-www-form-urlencoded
param=abcd
```

Блокирует запросы следующего вида:

```
POST / HTTP/1.1
Host: wafcpp-debian-cl5.rd.ptsecurity.ru:8080
Content-Type: application/x-www-form-urlencoded
param=1
```

Пример 3

```
{
    "jsonrpc": "2.0",
    "method": "train",
    "params": {"model": "{\"param_name\":\"param\",\"path\":\"/safe/
\",\"profile_id\":\"528e2758cd80bc1b8633f863\",\"reqvar_type\":\"REQU
EST_COOKIES\"}", "iter": 10000, "seq": ["aa", "aaa", "aaaa", "aaaaa"]},
    "id": 1
}
```

Блокирует запросы следующего вида:

```
GET / HTTP/1.1
Host: wafcpp-debian-cl5.rd.ptsecurity.ru:8080
Cookie: param=1
<No body>
```

Пример 4

```
{
    "jsonrpc": "2.0",
    "method": "train",
    "params": {"model": "{\"param_name\":\"param\",\"path\":\"/safe/
\",\"profile_id\":\"528e2758cd80bc1b8633f863\",\"reqvar_type\":\"REQUEST_HEADE
RS\"}", "iter": 10000, "seq": ["aa", "aaa", "aaaa", "aaaaa"]},
    "id": 1
}
```

Блокирует запросы следующего вида:

```
GET / HTTP/1.1
Host: wafcpp-debian-cl5.rd.ptsecurity.ru:8080
param: 1
<No body>
```

Пример 5

```
{
    "jsonrpc": "2.0",
    "method": "train",
    "params": {"model": "{\"param_name\":\"param\",\"path\":\"/safe/
\",\"profile_id\":\"528e2758cd80bc1b8633f863\",\"reqvar_type\":\"REQUEST_XML\"}"
, "iter": 10000, "seq": ["aa", "aaa", "aaaa", "aaaaa"]},
    "id": 1
}
```

Блокирует запросы следующего вида:

```
POST / HTTP/1.1
Host: wafcpp-debian-cl5.rd.ptsecurity.ru:8080
Content-Type: application/xml
<?xml version="1.0"?>
<param>1</param>
```

Пример 6

```
{
    "jsonrpc": "2.0",
    "method": "train",
    "params": {"model": "{\"param_name\":\"param\",\"path\":\"/safe/
\",\"profile_id\":\"528e2758cd80bc1b8633f863\",\"reqvar_type\":\"REQUEST_JSON\"
}", "iter": 10000, "seq": ["aa", "aaa", "aaaa", "aaaaa"]},
    "id": 1
}
```

Блокирует запросы следующего вида:

```
POST / HTTP/1.1
Host: wafcpp-debian-cl5.rd.ptsecurity.ru:8080
Content-Type: application/json
{
"param":"1"
}
```

6.5.6. Виртуальный патчинг

Вкладка предназначена для создания виртуального патча, закрывающего уязвимое приложение, проанализированное в программе Application Inspector (далее PT AI) компании Positive Technologies. На данной вкладке система позволяет загружать отчеты, полученные в программе PT AI.

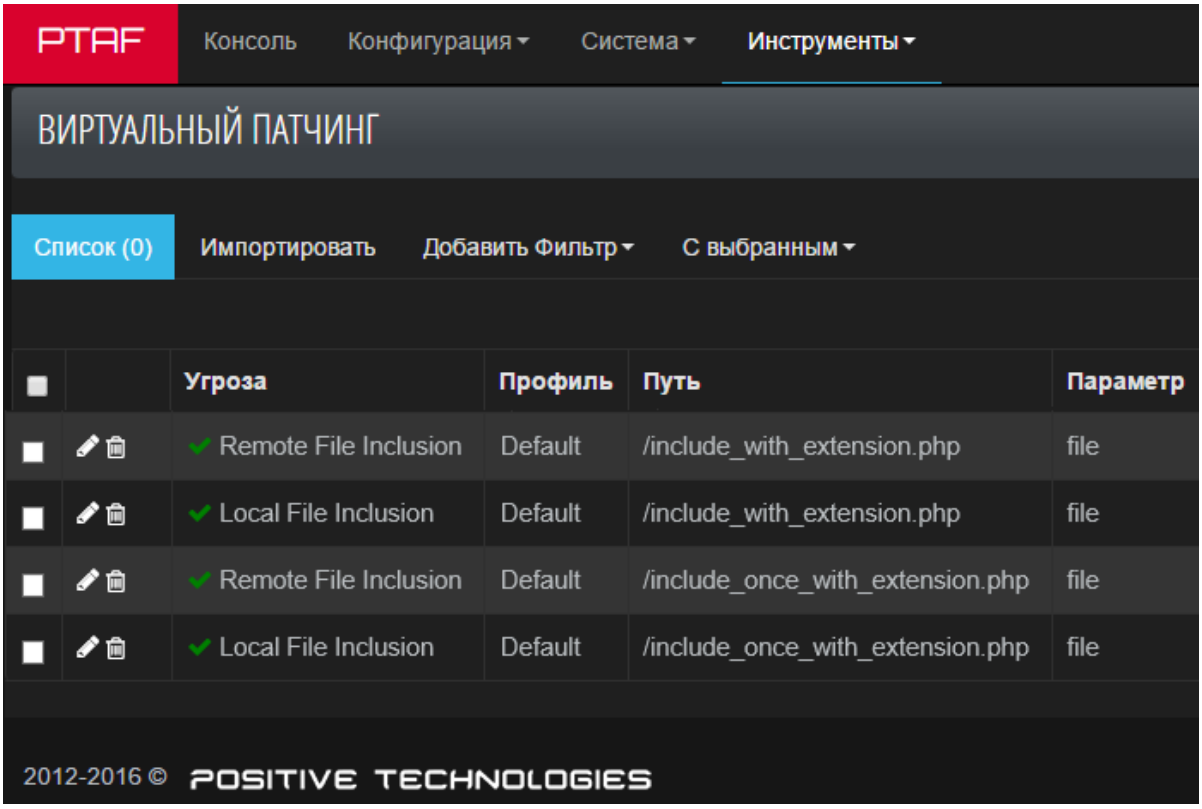


Рис. 158 – Виртуальный патчинг

6.5.7. Резервные копии

Вкладка предназначена для создания и восстановления резервных копий.

Для каждой резервной копии в списке указано: имя, дата, формат, размер и статус (UPLOADING – отображается в процессе загрузки файла резервной копии, PROCESSED – отображается в том случае, когда файл копии создан, FAILED – отображается при ошибке создания резервной копии).

6.5.7.1. Создание резервной копии

Нажмите кнопку *Создать резервную копию*, чтобы получить копию в заархивированном виде.

> **Внимание!** Рекомендуется создание резервной копии после завершения базовой конфигурации системы.

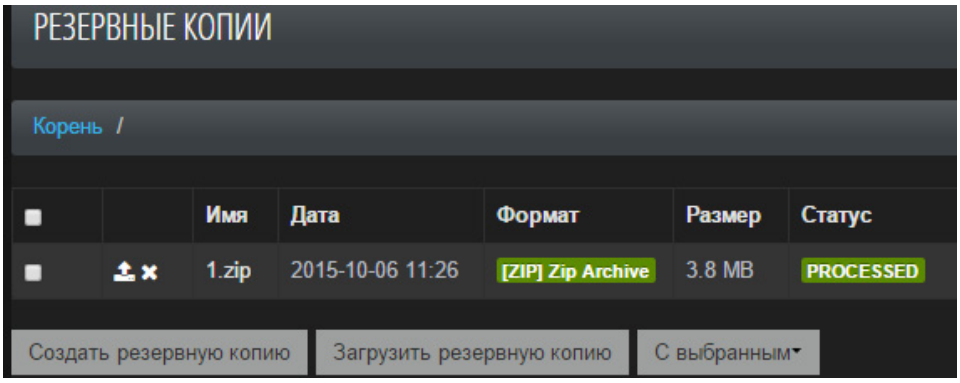


Рис. 159 – Резервные копии

Содержание архива зависит от установленных опций:

- Attacks – данные атак, хранящиеся в Elasticsearch;
- Users – пользователи системы. По умолчанию опция выключена, чтобы при восстановлении старых данных из резервной копии, новые пользователи не были удалены;
- Configuration – значения всех настроек системы, кроме тех, которые соответствуют объектам *SSL* и *Users* (конфигурация PT AF в MongoDB);
- SSL – файлы, загруженные в оснастку SSL Certificates Keys. По умолчанию опция выключена, чтобы исключить попадание в резервную копию данных, не подлежащих разглашению.

При необходимости полученный архив можно загрузить на диск и затем использовать для восстановления резервной копии.

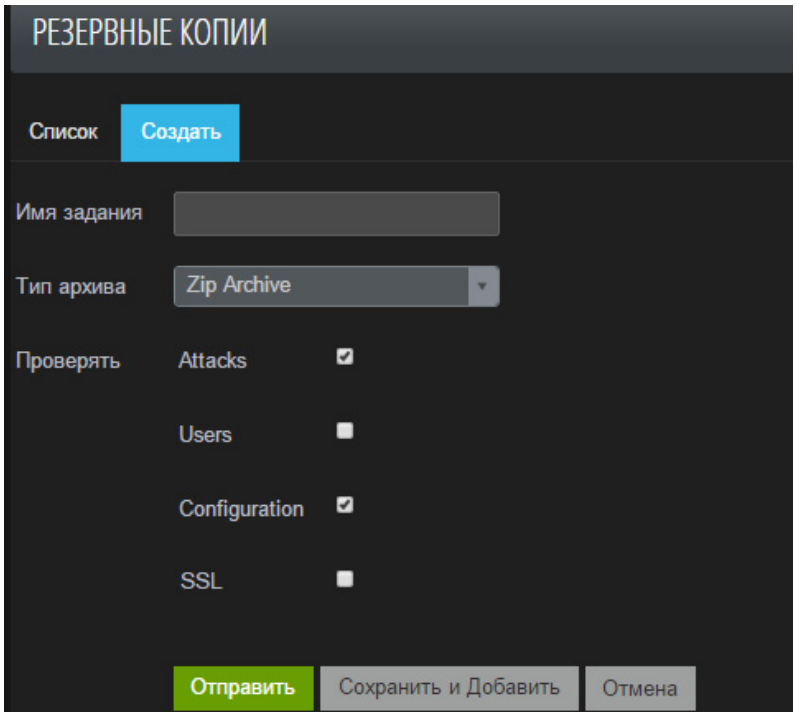


Рис. 160 – Создание резервной копии

6.5.7.2. Загрузка резервной копии

Нажмите кнопку *Загрузить резервную копию*, чтобы перейти на форму загрузки резервных копий.

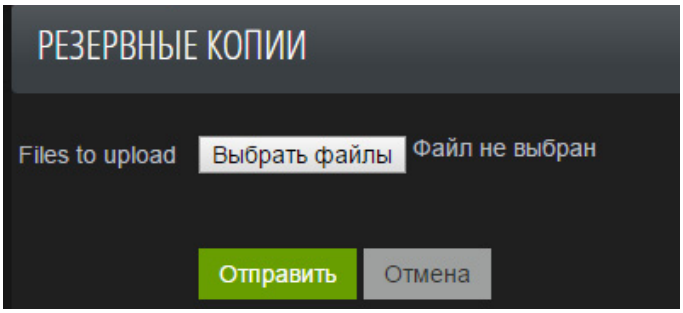


Рис. 161 – Форма загрузки резервных копий

Выберите файлы для загрузки (поддерживается множественная загрузка) и нажмите кнопку *Отправить*.

Загружаемые файлы проверяются на соответствие формату файлов резервных копий, поддерживаемому системой. Для поддерживаемых файлов начинается их загрузка, для неподдерживаемых пишется соответствующее сообщение.

6.5.7.3. Восстановление данных из резервной копии

Нажмите кнопку *Restore from backup* в строке файла резервной копии.

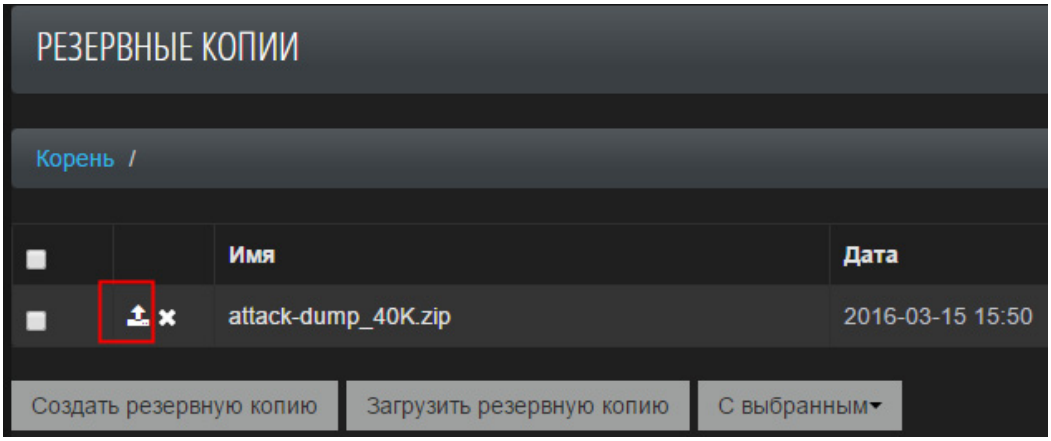


Рис. 162 – Кнопка восстановления из резервной копии

В новом окне необходимо выбрать объекты для восстановления и нажать кнопку *Отправить*.

Восстанавливаемые объекты переписывают существующие в системе объекты:

- Attacks - стирает все существующие атаки (удаляет индекс в elasticsearch) и записывает атаки из резервной копии.
- Configuration - стирает все существующие настройки (удаляет коллекции в mongodb) и записывает настройки из резервной копии.

• SSL - удаляет все файлы, загруженные в оснастку SSL Certificates Keys, и загружает файлы из резервной копии.

• Users - удаляет всех текущих пользователей системы и восстанавливает пользователей из резервной копии.

Примечание: в списке для восстановления отображаются только те объекты, которые есть в файле резервной копии.

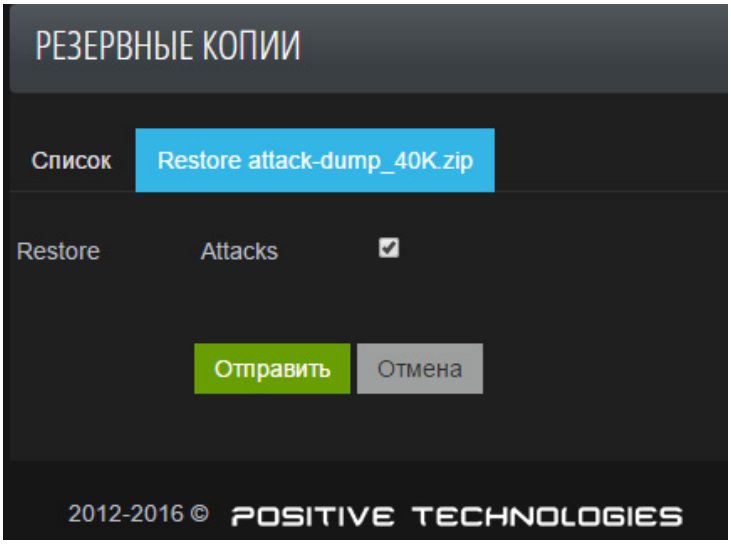


Рис. 163 – Окно восстановления из резервной копии

6.5.8. Расписание резервных копий

На этой странице можно автоматизировать создание резервных копий.

Нажмите кнопку *Создать* на данной вкладке, чтобы настроить расписание создания резервной копии.

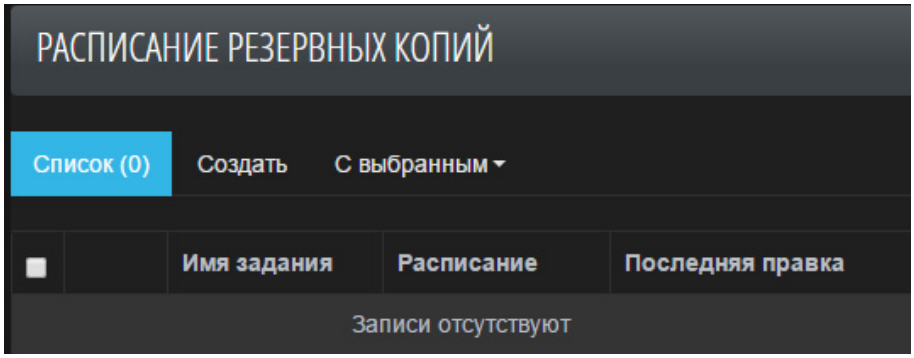


Рис. 164 – Расписание резервных копий

Установите необходимое время, день и месяц в формате crontab для запуска создания резервной копии. Настройте содержание архива резервной копии (см. описание параметров в главе «Резервные копии»).

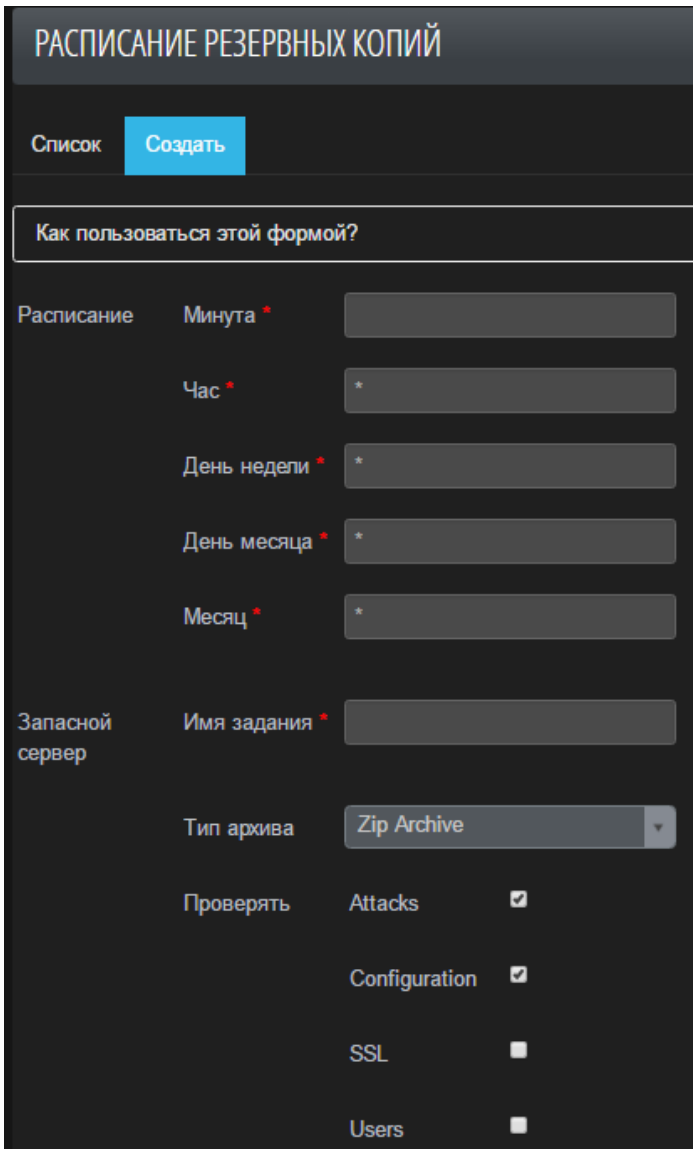


Рис. 165 – Создание расписания

Внимание! Значения в полях секции *Расписание* должны быть в формате crontab. Нажмите ссылку «Как пользоваться этой формой?», чтобы посмотреть примеры заполнения полей (Рис. 166).

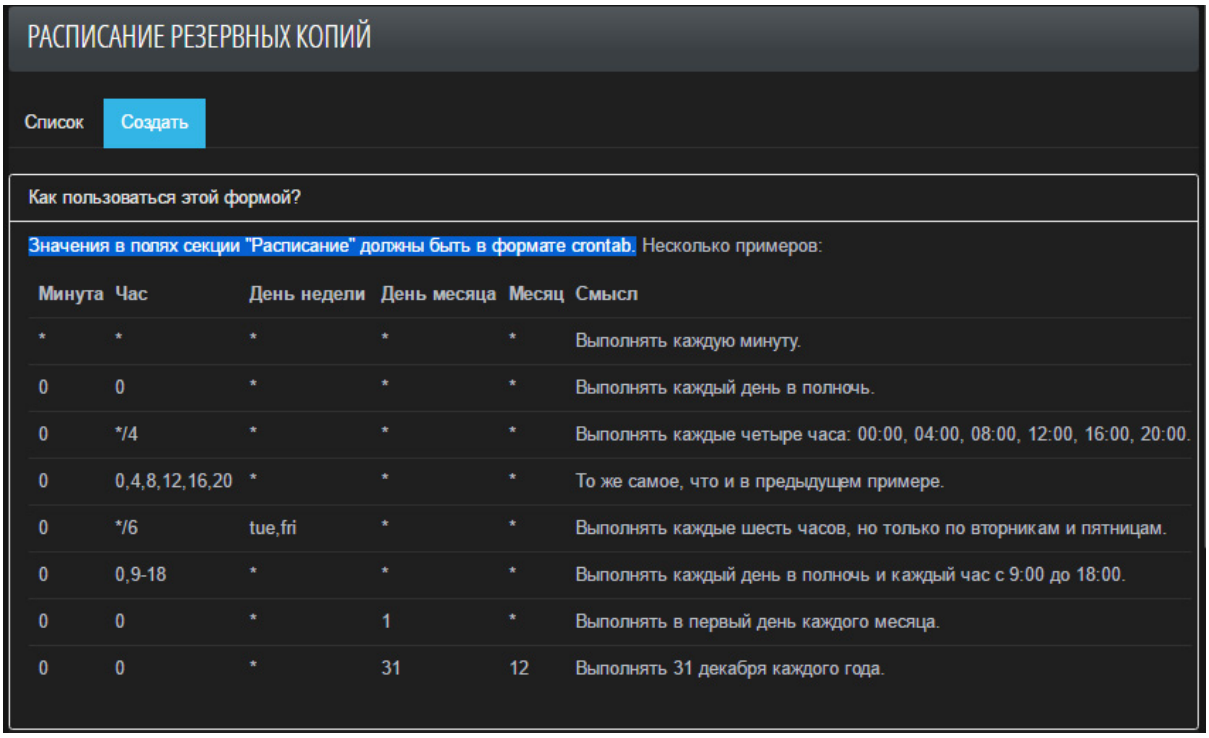


Рис. 166 – Примеры заполнения полей

7. Начальная настройка

7.1. Порядок настройки

При первом запуске PT AF подключение к нему происходит через локальную консоль сервера. Имя пользователя по умолчанию для консоли – «pt», а пароль – «p0s1t1v3». После входа в консоль необходимо настроить параметры сетевых интерфейсов. При этом интерфейс eth0 является интерфейсом управления PT AF.

После настройки сетевых интерфейсов согласно выбранной схеме подключения необходимо открыть веб-консоль администрирования и авторизоваться. Веб-консоль доступна по адресу: https://x.x.x.x:8443/, где x.x.x.x – IP-адрес интерфейса управления. Вместо IP-адреса можно использовать имя узла (hostname), т.е. https://hostname:8443/

Имя пользователя и пароль по умолчанию: «admin», «p0s1t1v3».

Внимание! Сразу после начала работы с PT AF смените пароль, который был выдан по умолчанию.

На домашней странице веб-консоли отображается сводная информация по работе PT AF.

Для дальнейшей настройки необходимо выполнить следующие шаги:

- Создать необходимые роли (WAN, LAN, MGMT, SPAN) на вкладке *Конфигурация -> Сеть -> Алиасы сетевых интерфейсов*. Если есть необходимость в нестандартных портах, то добавьте их в соответствующие поля на форме при редактировании алиаса;
- Назначить роли на используемые сетевые интерфейсы на вкладке *Конфигурация -> Сеть -> Шлюзы -> Сеть* (например, eth0: WAN+MGMT, eth1: SPAN);
- Включить опцию *Активен* на вкладке *Конфигурация -> Сеть -> Шлюзы - > Основные* и сохранить настройки;
- Добавить Группу Серверов (защищаемые сервера).

В веб-консоли предустановлен профиль *Default*, который обрабатывает трафик, приходящий на порт, указанный в этом профиле (по умолчанию используется 80 порт). Изменение группы серверов в этом профиле на защищаемый сервер уже будет достаточно, чтобы PT AF начал обрабатывать трафик, предназначенный для приложения. Так же для каждого веб-приложения может быть создан свой профиль, который определяет правила обработки трафика приложения.

После начала обработки трафика PT AF автоматически начинает постоянный процесс самообучения, то есть автоматически создает правила для снижения количества ложных срабатываний. Если окажется, что одно из правил работает неправильно, то можно отключить его.

Если администратору необходимо вручную создать разрешающее или запрещающее правило для обработки HTTP-пакетов, то для этого надо перейти ко вкладке *Конфигурация*.

Модуль для ведения черных и белых списков IP-адресов доступен на вкладке *Конфигурации -> Сеть -> Файрвол*.

Вкладка *Система -> Статус -> Мониторинг* предназначена для просмотра информации в режиме реального времени. Здесь отображается информация двух типов:

- обработка трафика;
- параметры производительности самого PT AF.

7.2. Создание нового профиля

Профиль – объект, который позволяет настроить PT AF для работы с конкретным сайтом или набором сайтов.

При создании нового профиля необходимо заполнить следующие поля:

- Имя – имя профиля;
- Шаблон – шаблон для применения предустановленных политик;
- Хост – узел источника (HTTP-заголовок Host), поддерживаются групповые символы для включения всех совпадений, например *.host.ru;
- ПО защищаемого сервера – платформа бэкэнд-сервера.

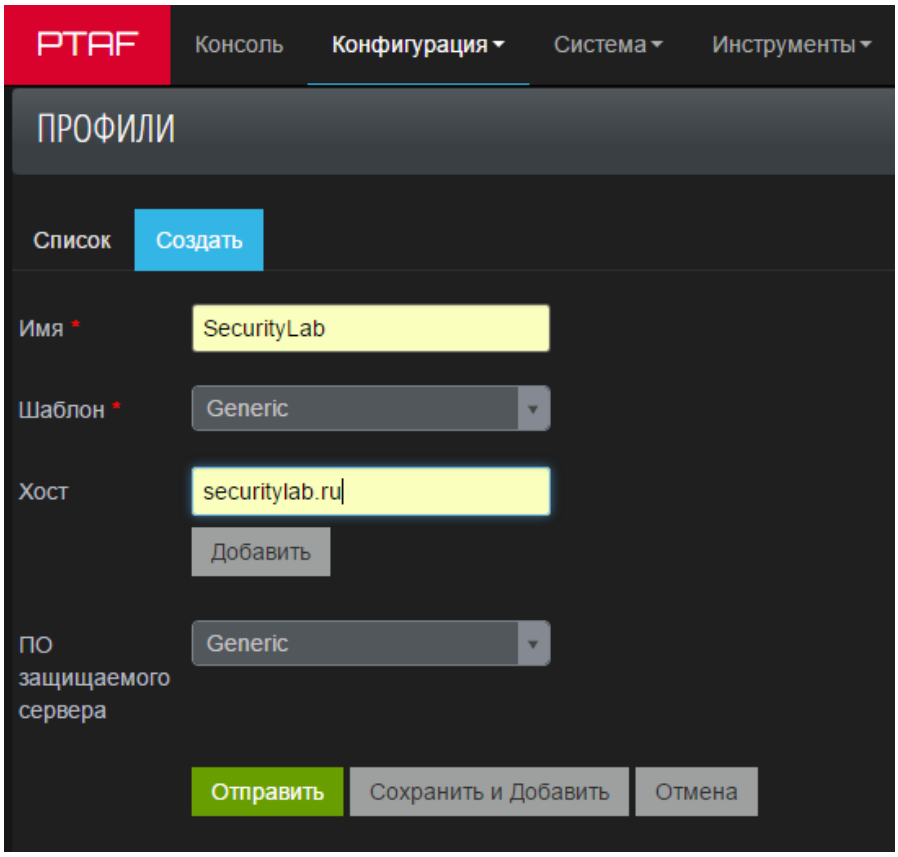


Рис. 167 – Создание нового профиля

7.3. Конфигурация модулей защиты

Для настройки параметров модулей защиты следует выбрать меню *Конфигурация - Политики безопасности -> Профили*, нажать кнопку *Редактировать* (см. Рис. 168) в строке конфигурируемого профиля, а затем перейти на вкладку *Модули*.

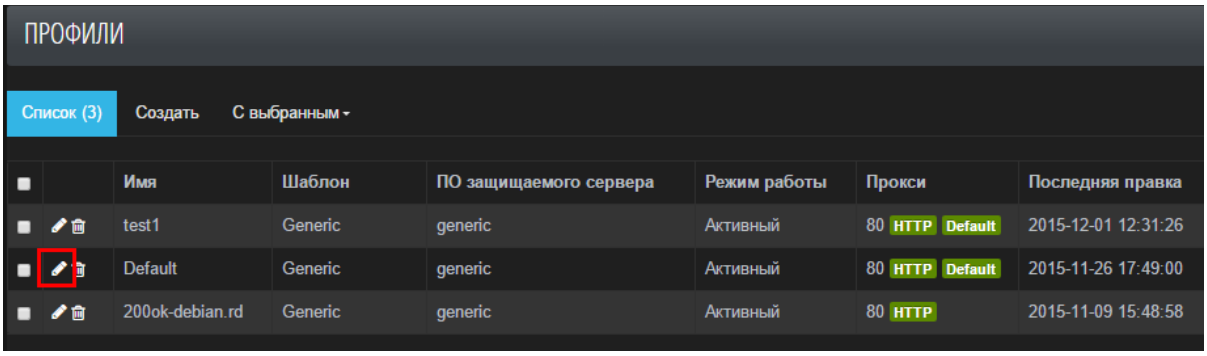


Рис. 168 – Редактирование профиля

7.3.1. Защита HTTP

Осуществляет проверки HTTP-запросов на низком уровне и на соответствие RFC.

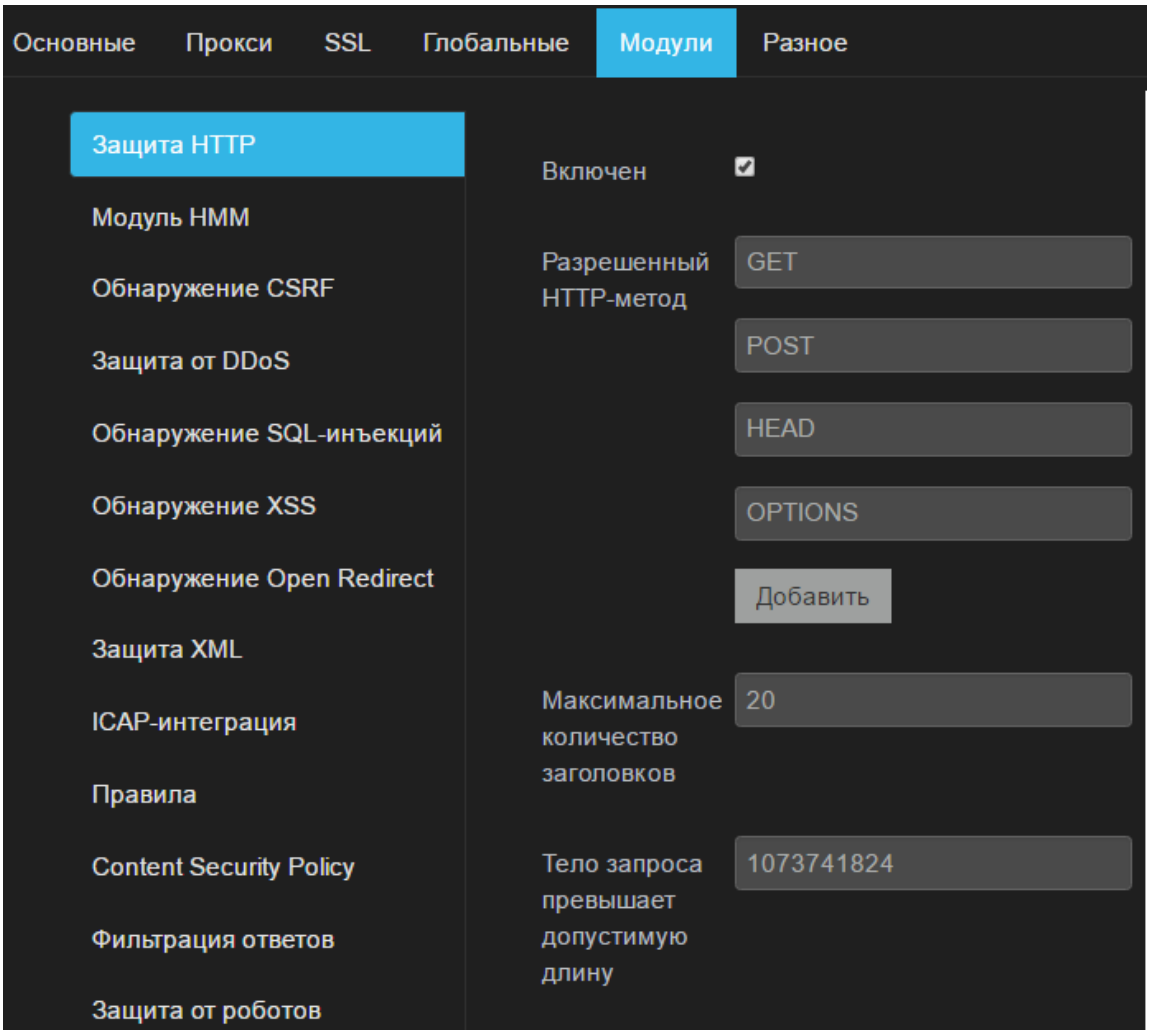


Рис. 169 – Защита HTTP

Модуль *Защита HTTP* предназначен для ограничения определенных параметров протокола HTTP:

- Ограничение на список используемых методов HTTP (противодействие разведке);
- Ограничение на количество HTTP-заголовков;
- Ограничение на длину HTTP-запроса;
- Ограничение на длину одного HTTP-заголовка;
- Проверка на содержание IP-адреса в поле Host (противодействие разведке);
- Отсутствие заголовка Host;
- Отсутствие заголовка Content-Type;
- Проверка содержимого заголовка Content-Type (ограничение на типы передаваемого контента);
- Таймаут ожидания заголовков и тела запроса;
- Проверка корректности Cookie-токена;
- Отсутствие Cookie-токена;
- Проверка названия заголовков;
- Проверка токена формы;
- Отсутствие токена формы.

При необходимости настройте *События*.

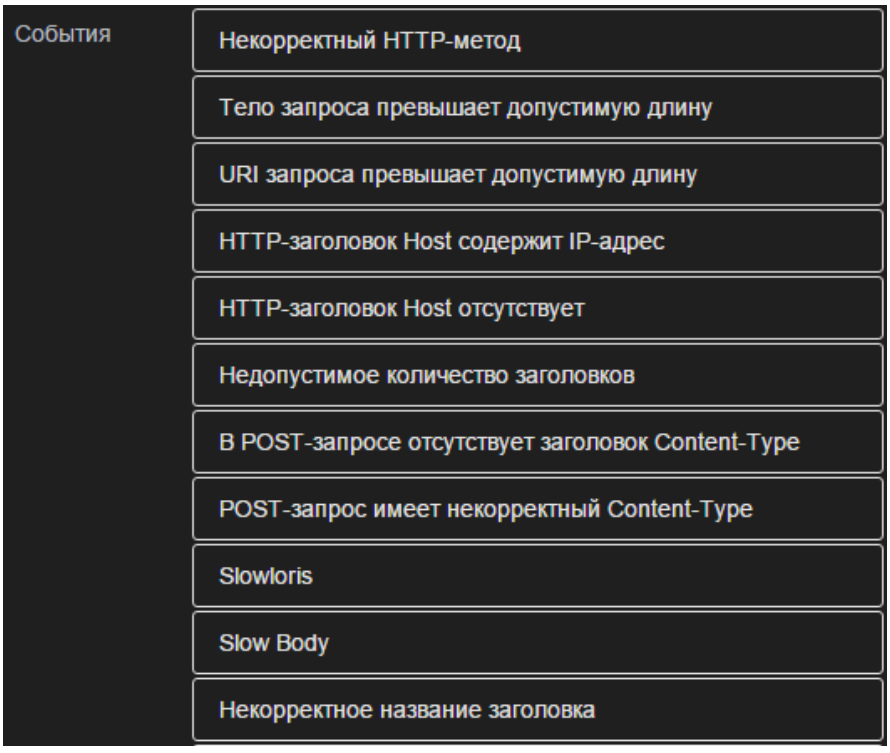


Рис. 170 – Настройка проверки заголовков

7.3.1.1. Некорректный HTTP-метод

Согласно RFC 2616 в протоколе HTTP 1.1 определено 7 методов: OPTIONS, GET, HEAD, POST, PUT, DELETE и TRACE. Из них для передачи данных между клиентом и сервером в большинстве случаев используются только GET и POST. Метод HEAD передает информации не больше, чем метод GET (согласно RFC метод HEAD должен отдавать ту же информацию, что и GET, прекращая передачу данных после отправки заголовка HTTP). Метод OPTIONS представляет запрос информации об опциях соединения, доступных в цепочке запросов/ответов, идентифицируемой запрашиваемым URI. Этот метод позволяет клиенту определять опции и/или требования, связанные с ресурсом, или возможностями сервера, но не производя никаких действий над ресурсом и не инициируя его загрузку. Остальные методы рекомендуется отключить так как они могут использоваться злоумышленником как минимум для целей разведки.

В поле *Разрешенный HTTP-метод* задается список методов, разрешенных к выполнению на стороне веб-сервера.

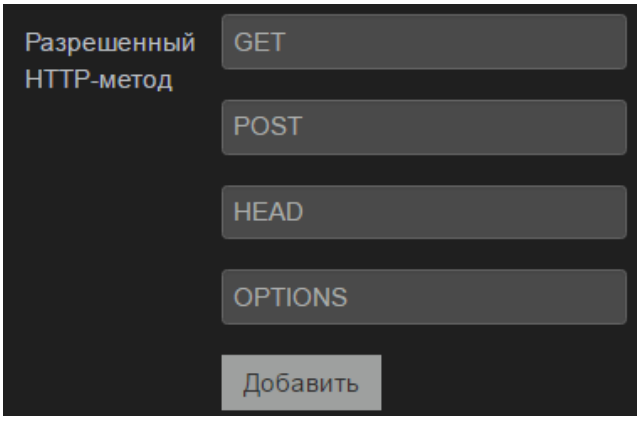


Рис. 171 – Разрешенный HTTP-метод

При некорректной настройке веб-сервера возможно нанести вред приложению: загрузить вредоносный код (метод PUT), удалить важные ресурсы (метод DELETE), получить доступ к чувствительной информации (метод TRACE).

В большинстве случаев, для взаимодействия с веб-приложением достаточно методов GET, POST, HEAD, OPTIONS.

7.3.1.2. Тело запроса превышает допустимую длину

В поле *Тело запроса превышает допустимую длину* задается максимальная длина HTTP-запроса, отправленная с методом POST. По умолчанию значение равно 1 ГБ.

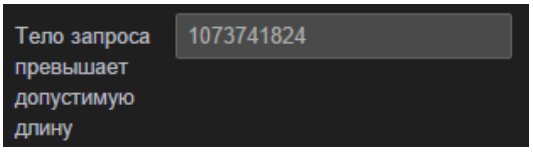


Рис. 172 – Тело запроса превышает допустимую длину

Этот параметр настраивается под конкретное приложение и предназначен для снижения нагрузки в случае DoS-атак.

7.3.1.3. Максимальная длина одного HTTP-заголовка («URI запроса превышает допустимую длину»)

В поле *Максимальная длина одного HTTP-заголовка* указывается ограничение на максимальный размер параметра, который может быть передан в HTTP-запросе веб-серверу.

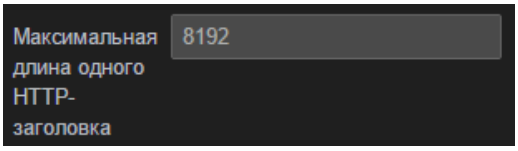


Рис. 173 – Максимальная длина одного HTTP-заголовка

Значение по умолчанию равно 8192 и совпадает с ограничением на длину ссылок в apache. При превышении этого порога сервер может непредсказуемо обработать запрос и вернуть злоумышленнику лишнюю информацию.

Для проверки возможного превышения длины значения заголовка используется событие «URI запроса превышает допустимую длину».

7.3.1.4. HTTP-заголовок Host содержит IP-адрес

В заголовке HTTP-запроса должно содержаться доменное имя веб-приложения. Если в этом поле указать IP-адрес сервера, он может вернуть лишнюю информацию, например, открыть другое веб-приложение, расположенное на том же физическом сервере.

7.3.1.5. HTTP-заголовок Host отсутствует

В HTTP-запросе стандарта HTTP 1.1 обязательно должен присутствовать заголовок *Host* (RFC 2616). Если это поле отсутствует, то с большой вероятностью запрос сгенерирован искусственно, и рекомендуется его заблокировать.

> **Примечание:** для стандарта HTTP 1.0 заголовок Host не требуется, но событие все равно будет срабатывать.

7.3.1.6. Недопустимое количество заголовков

Передача большого количества заголовков в запросе – один из вариантов DoS-атаки, а также возможный признак slowloris-атаки. Данное событие связано с настройкой *Максимально количество заголовков*.

7.3.1.7. В POST-запросе отсутствует заголовок Content-Type

Некоторые POST-запросы могут не содержать поля Content-Type. Однако, такие запросы могут вызвать ошибки в веб-приложении, поэтому их рекомендуется блокировать.

7.3.1.8. POST-запрос имеет некорректный Content-Type

В поле *Допустимые значения Content-Type в POST-запросах* задается список возможных значений заголовка Content-Type для POST-запросов.

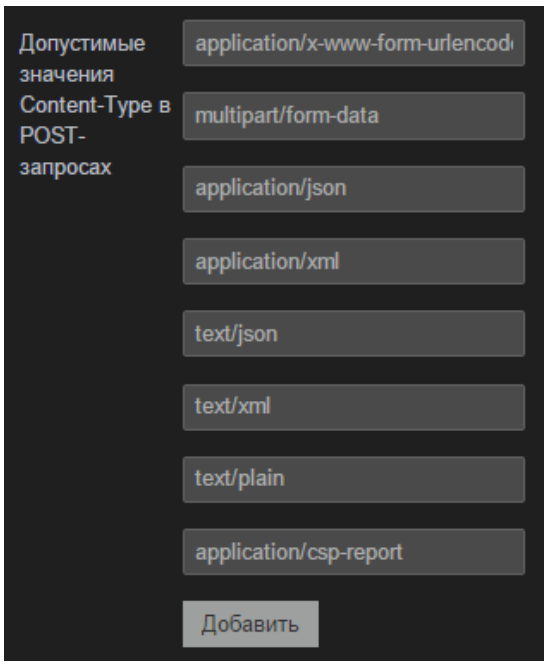


Рис. 174 – Допустимые значения Content-Type в POST-запросах

Эта настройка позволяет ограничить форматы, с которыми будет работать веб-приложение. Например, если веб-приложение будет отображать пользователю видео, к списку следует добавить «video/mp4» или «video/quicktime» в зависимости от формата видео.

7.3.1.9. Slowloris

Детектирование атаки типа Slowloris. Атака состоит в том, что злоумышленник открывает большое количество соединений с сервером и постоянно «досылает» заголовки, что заставляет сервер держать соединения открытыми, что, в свою очередь, приводит к исчерпанию ресурсов на нем. Эта атака имеет разную эффективность в отношении разных веб-серверов. Так в отношении сервера Apache 1.x, Apache 2.x, dhttpd, GoAhead WebServer и Squid она эффективна, а в отношении IIS6.0, IIS7.0, lighthttpd нет, т.к. последние могут удерживать практически неограниченное количество открытых соединений. Так или иначе, атака Slowloris не может пройти через PT AF в силу его архитектурных особенностей. Однако данное поле в модуле защиты может пригодиться для журналирования факта атаки.

7.3.1.10. Slowbody

Детектирование одной из разновидностей Slow-HTTP атак. Атака состоит в том, что злоумышленник открывает большое количество соединений с сервером и отправляет тело запроса на сервер с очень малой скоростью (несколько байт в секунду). Подробнее см. описание атаки Slowloris.

7.3.1.11. Некорректное название заголовка

Недопустимые символы в заголовке запроса могут привести к ошибкам и отказам от обслуживания при попытке разбора заголовков регулярными выражениями.

7.3.1.12. Invalid Cookie token

Cookie будут подписываться на стороне сервера, если в соответствующем профиле в модуле *Защита HTTP* установить опцию *Sign HMAC* и включить событие *Invalid Cookie Token*. В поле *Имя токена* следует указан постфикс, который будет добавляться к имени Cookie при именовании токена. По умолчанию указан постфикс __sig. Таким образом, если имя Cookie – pmaPass, то имя токена – pmaPass__sig.

Cookie подписываются на стороне сервера и указываются в *Set-Cookie*. Cookie, устанавливаемые на стороне клиента, не подписываются. Cookie и токены затем указываются клиентом в запросах.

Событие *Invalid Cookie Token* срабатывает в следующих случаях:

1. токен в запросе присутствует, но нет соответствующей ему Cookie;
2. указано неверное значение токена в запросе;
3. имя токена содержит неверное имя Cookie. Например, pma1Pass__sig не соответствует Cookie pmaPass. В данном случае срабатывает также событие *Missing Cookie Token*, т.к. для pmaPass не указана подпись.

Срабатывание события свидетельствует о попытке подделки или модификации легитимного запроса. Если веб-приложение не использует токены, событие рекомендуется отключить и отключить опцию *Sign HMAC*.

7.3.1.13. Missing Cookie token

Cookie будут подписываться на стороне сервера, если в модуле *Защита HTTP* установить опцию *Sign HMAC* и включить событие *Missing Cookie Token*.

Событие срабатывает в единственном случае, когда в запросе отсутствует токен, соответствующий некоторой Cookie. Отсутствие подписи свидетельствует о попытке модификации легитимного запроса с целью проведения некоторой атаки. Если веб-приложение не использует токены, событие рекомендуется отключить и отключить опцию *Sign HMAC*.

7.3.1.14. Invalid Form Token

Событие, которое срабатывает в случае, если токен пришел в запросе, но его значение не совпадает с ожидаемым (описание механизма защиты веб-форм см. в главе «Подпись форм»).

7.3.1.15. Missing Form Token

Событие, которое срабатывает в случае отсутствия токена в запросе (описание механизма защиты веб-форм см. в главе «Подпись форм»).

7.3.2. Модуль HMM

Модуль, использующий скрытые Марковские модели для адаптивного обучения формата параметров и обнаружения попыток внедрения нестандартных символов.

На основе анализа трафика, проходящего через AF, строятся определенные модели, например, в параметре «id» передаются только цифры, а в поле login не могут встречаться специальные символы.

Модуль HMM работает следующим образом. Из 6 возможных каналов (GET, POST, Cookie, Headers, XML и JSON) пришедшего запроса извлекаются все поступившие параметры со значениями и отправляются в соответствующие модели, где эти параметры "накапливаются" и модель становится кандидатом на стабильность. Пока модель является кандидатом на стабильность, она не применяется для блокирования аномальных запросов. Для того, чтобы модель стала стабильной и начала применяться, необходимо чтобы выполнились следующие условия: минимальное количество образцов - 500, при этом процент атак не должен превышать 0,1%. Данные значения установлены по умолчанию, их возможно настроить во вкладке *Настройки обучающего модуля*. В случае обнаружения ложных срабатываний на основе эвристики модель будет отправлена на переобучение, а инциденты безопасности будут удалены из консоли. Модель считается ошибочной, если у большинства пользователей на сайте за короткий промежуток времени данная модель вызывает срабатывания протектора. Основное условие – это равномерность срабатываний во времени, при этом алгоритм обнаружения ложных срабатываний может сам подстроиться под текущий трафик пользователя.

Необходимо настроить следующие коэффициенты (см. Рис. 175):

- Коэффициент порога для классификации параметра как подозрительного (значение по умолчанию – 1.0 σ) – определяет, насколько сильно значение параметра должно отличаться от тех данных, на которых модель обучилась, чтобы параметр считался подозрительным;
- Коэффициент порога для классификации параметра как потенциально опасного (значение по умолчанию – 2.0 σ) – определяет, насколько сильно значение параметра должно отличаться от тех данных, на которых модель обучилась, чтобы параметр считался потенциально опасным;
- Коэффициент порога для классификации параметра как крайне опасного (значение по умолчанию – 3.0 σ) – определяет, насколько сильно значение параметра должно отличаться от тех данных, на которых модель обучилась, чтобы параметр считался крайне опасным.

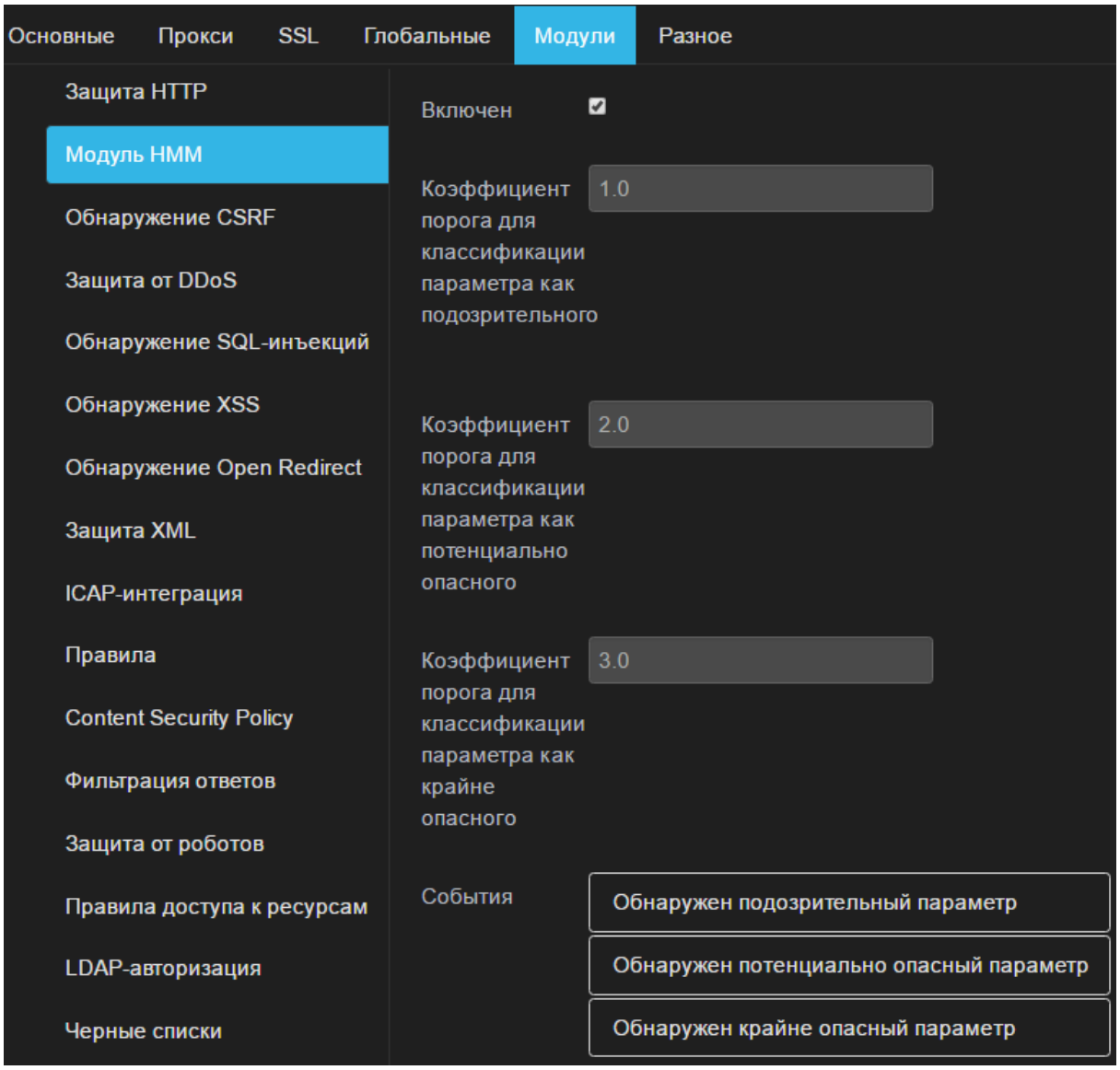


Рис. 175 – Настройка параметров модуля HMM

7.3.3. Обнаружение CSRF

Модуль для защиты от уязвимостей класса «Межсайтовая подделка запросов».

Суть CSRF сводится к выполнению каких-либо запросов на уязвимом сайте от лица жертвы (например, изменение пароля, почты, добавление администратора и т.д.). Если жертва заходит на сайт, созданный злоумышленником, от ее лица тайно отправляется запрос на другой сервер (например, на сервер платежной системы), осуществляющий некую вредоносную операцию (например, перевод денег на счет злоумышленника).

Необходимо настроить следующие параметры:

- Имя токена – имя параметра-токена, которое добавляется в HTML-формы. Представляет собой уникальное значение, предотвращающее непреднамеренную отправку формы (по умолчанию csrftoken);
- Разрешить отправку POST запросов хоста.

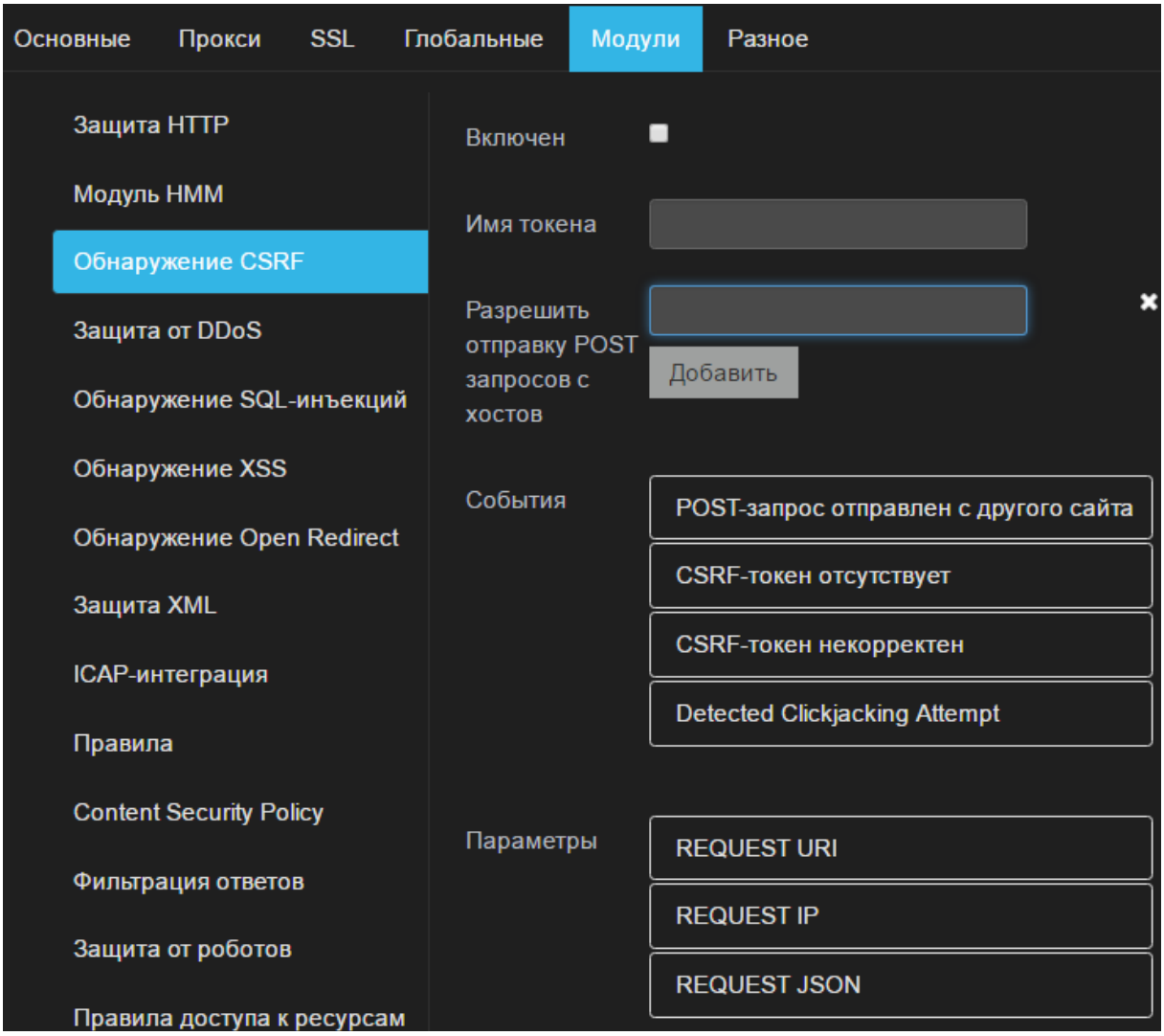


Рис. 176 – Обнаружение CSRF

7.3.3.1. Добавление CSRF-токена к запросам

Одним из механизмов защиты от атаки CSRF является усиление традиционного механизма подтверждения HTTP-сессии при помощи передачи HTTP cookie механизмом, схожим с одноразовыми паролями. Ко всем POST-запросам добавляется токен, который проверяется перед отправкой веб-приложению.

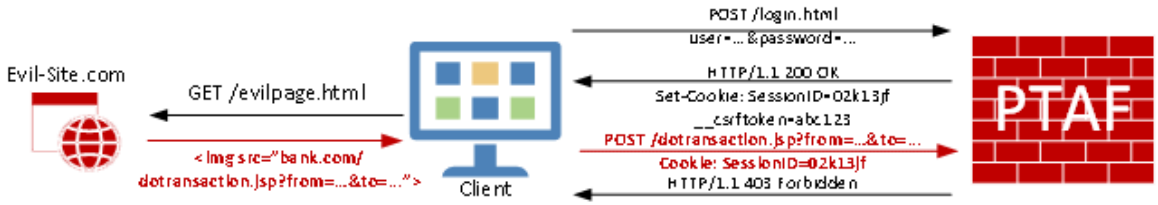


Рис. 177 –

Для этого в PT AF настраиваются следующие параметры:

- Имя токена – имя переменной, в которой будет храниться токен для запроса;
- Разрешить отправку POST-запросов с хостов – список исключений, для которых разрешается отправлять POST-запросы;

Запрос будет заблокирован PT AF в случае, если токен не верный, отсутствует, или запрос приходит с недоверенного источника.

7.3.3.2. CSRF-токен некорректен

Попытка использовать CSRF-токен с истекшим сроком действия или невалидный токен. Индикатор попытки подделать легитимный запрос. Если веб-приложение не использует CSRF-токены, модуль защиты рекомендуется отключить.

7.3.3.3. CSRF-токен отсутствует

Попытка прислать запрос без указания CSRF-токена, в то время как веб-приложение использует защиту от CSRF. Индикатор возможной атаки. Если веб-приложение не использует CSRF-токены, модуль защиты рекомендуется отключить.

7.3.3.4. POST-запрос отправлен с другого сайта

Это событие срабатывает в том случае, когда в POST-запросе присутствует заголовок Referer с URL, с которого отправлен запрос, и доменное имя в этом URL отличается от имени, указанного в заголовке Host. При этом считается, что домены совпадают, если у них совпадают домены второго уровня, например, следующий запрос не вызовет событие безопасности:

```
POST / HTTP/1.1
Host: mydomain1.example.ru
Referer: https://mydomain2.example.ru
…
```

7.3.3.5. Detected Clickjacking Attempt

Данное событие срабатывает при получении уведомления от waf.js в момент, когда обнаружена загрузка защищенной страницы сайта через iframe.

7.3.4. Защита от DDoS-атак

DDoS-атаки представляют собой класс атак, направленный на ввод серверного программного или аппаратного обеспечения в состояние отказа от обслуживания. Атаки обычно ориентированы на переполнение оперативной памяти или жесткого диска, переполнение пула соединений, занятие 100% процессорного времени, разрыв существующих соединений и т.п.

Характерно то, что большинство DDoS-атак подчиняются известным шаблонам, что способствует их достаточно быстрому выявлению.

Дополнительным средством для защиты от DDoS-атак являются решения Arbor Networks. Информация по интеграции представлена в главе «Send to Arbor».

7.3.5. Обнаружение SQL-инъекций

Модуль для защиты от уязвимостей класса «Внедрение операторов SQL».

Модуль позволяет предотвратить атаку на веб-системы, заключающуюся во взломе сайта или программы, работающей с базами данных, то есть блокирует внедрение в запрос произвольного SQL-кода, который, например, позволил бы злоумышленнику прочитать содержимое любых таблиц, удалить, изменить или добавить данные, выполнить произвольные команды на атакуемом сервере.

7.3.5.1. Обнаружена попытка SQL-инъекции

Данный модуль использует библиотеку libinjection. С ее помощью передаваемый параметр разбивается на токены, преобразуется в fingerprint и сравнивается со списком fingerprint'ов, характерных для SQL-инъекций.

7.3.6. Обнаружение XSS

Модуль для защиты от атак класса «Межсайтовое выполнение сценариев». Производит проверку HTTP-ответов на предмет внедрения данных, поступающих от пользователя. Настройте следующий параметр (см. Рис. 178):

- Минимальное количество символов в строке для проверки – минимальная длина значения параметра для проверки.

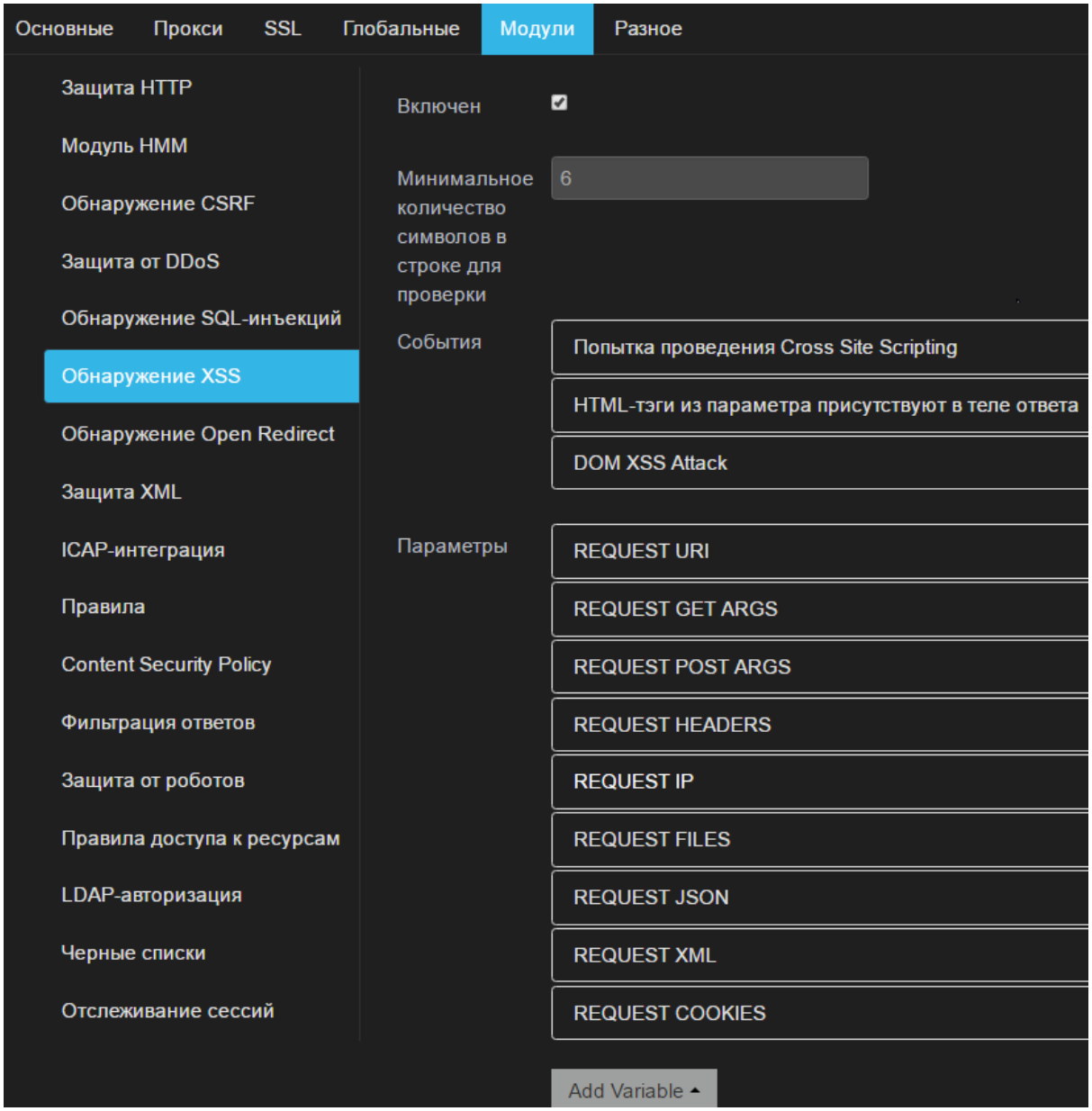


Рис. 178 – Обнаружение XSS

XSS-атака заключается в том, что злоумышленник получает возможность выполнить JavaScript-код в браузере жертвы. Это позволяет злоумышленнику украсть cookie и обойти механизмы авторизации на веб-серверах или выполнить другие атаки. Один из возможных способов осуществления этой атаки – внедрить свой JavaScript-код на страницу сервера. Например, отправить на форум сообщение «Hello world<script>alert("XSS!")</script>». Защищенный веб-сервер должен либо блокировать передачу таких данных, либо автоматически защищать свои страницы, например, автоматически заменяя «<» на «<», а «>» на «>». Другой разновидностью XSS является отраженная XSS. В этом случае JavaScript внедряется непосредственно в URI. Затем злоумышленнику остается при помощи социальной инженерии спровоцировать пользователя перейти по ссылке, которая содержит код. Ссылка не внедряясь на сервере вернется пользователю и код будет выполнен в его браузере.

7.3.6.1. HTML-теги из параметра присутствуют в теле ответа

Атаки типа XSS блокируются либо при помощи HMM, либо при помощи «Rule Engine» («Правила»). В модуле защиты *Обнаружение XSS* есть дополнительная возможность блокировать отраженные атаки, не сохраняющие код на стороне сервера. Для этого PT AF анализирует ответ сервера на предмет наличия параметров, переданных пользователем.

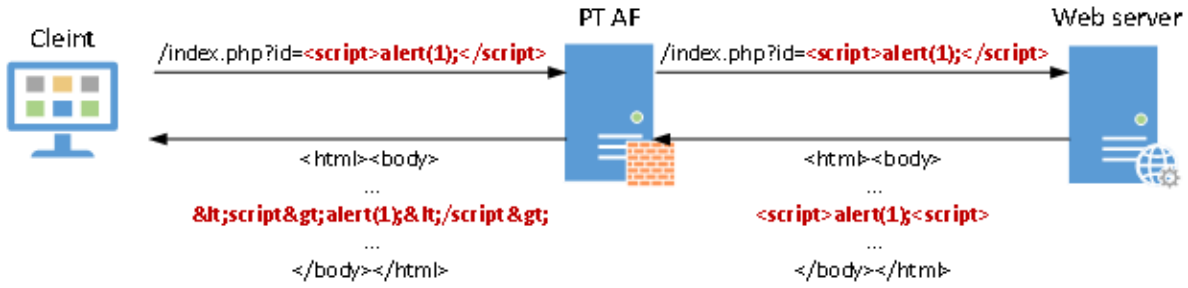


Рис. 179 –

Для этого модуля дополнительно задается параметр:

- Минимальное количество символов в строке для проверки – минимальная длина проверяемой строки, по умолчанию равна 6.

7.3.6.2. Попытка проведения Cross Site Scripting

Событие анализирует запрос и пытается найти в нем XSS по своему алгоритму. В отличие от события *HTML-теги из параметра присутствуют в теле ответа*, это событие анализирует только запрос, а не ответ от приложения.

7.3.6.3. DOM XSS Attack

Событие *DOM XSS Attack* срабатывает при обнаружении XSS-атак на стороне клиента с помощью waf.js, который пытается эту атаку предотвратить (путем фильтрации URL, cookies, localStorage и т.п.) и затем отправляет запрос с уведомлением об обнаруженной атаке. Данное событие может быть использовано только для журналирования факта обнаружения атаки. Событие работает через waf.js и отключается, если выключить waf.js во вкладке *Конфигурация -> Политики безопасности -> Профили -> Разное* (настройка *Inject waf.js into HTTP responses*).

7.3.7. Обнаружение Open Redirect

Модуль для защиты от атак класса «Открытое перенаправление».

Open Redirect перенаправляет пользователей на сайт, который указан в параметрах URI. При этом не осуществляется никакой валидации содержимого параметра. Такое перенаправление пользователей считается небезопасным, так как может применяться при «фишинге».

Необходимо настроить следующие события (см. Рис. 180):

- Попытка проведения Open Redirect через HTTP-заголовок Location – проверка URL в HTTP-заголовке Location, т.е. адрес страницы для перенаправления берется из URI;

- Попытка проведения Open Redirect через HTML-тег meta – проверка URL в HTML-метатегах с атрибутом «refresh». В тексте страницы может содержаться тэг вида <META http-equiv="refresh" content="5;URL=http://www.google.com">, который через 5 секунд перенаправит пользователя на указанную страницу. Если URL передается в параметрах запроса, злоумышленник может его подменить;
- Попытка проведения Open Redirect через HTTP-заголовок Refresh.

События срабатывают при перенаправлении на другой домен второго уровня, а также на тот же региональный домен. Если пользователь перенаправлен на тот же нерегиональный домен, события не срабатывают.

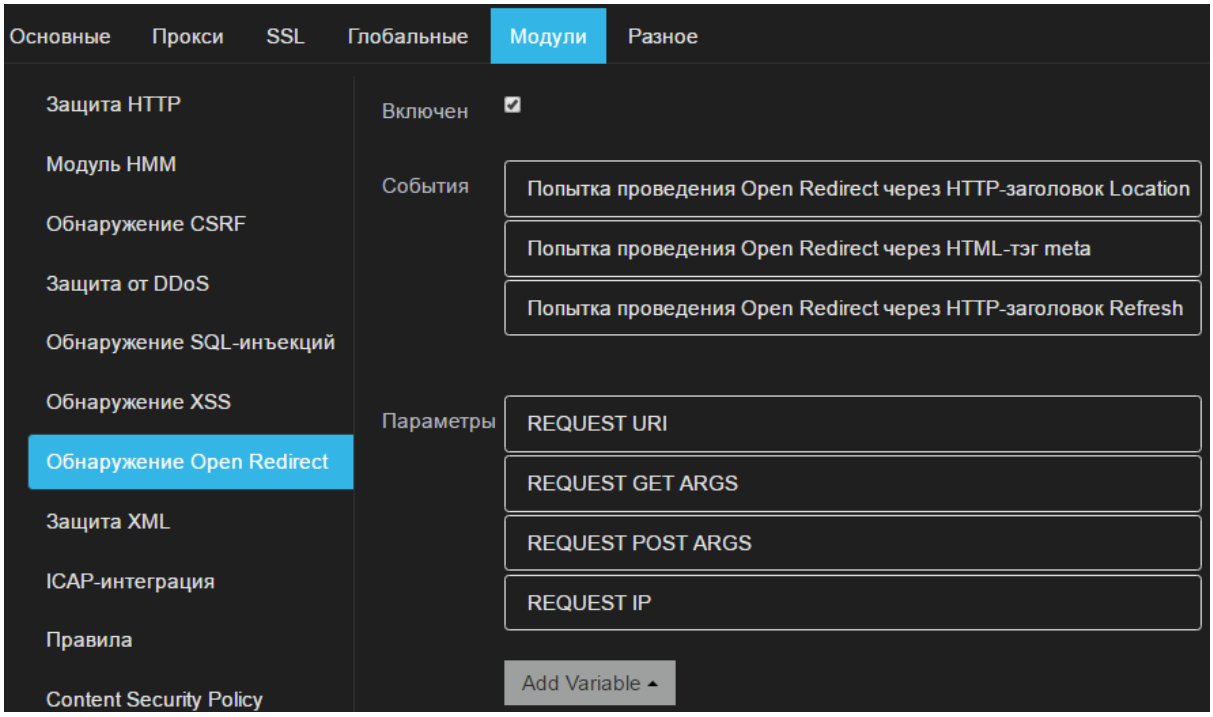


Рис. 180 – Обнаружение Open Redirect

7.3.8. Защита XML

Модуль для обработки XML-документов и SOAP-запросов.

7.3.8.1. Блокирование внешних XML-сущностей для предотвращения XXE-атак

Большая часть атак на XML сводится к загрузке внешнего модуля, который выполняет вредоносный код. События *Обнаружены внешние XML-сущности*, *External Entities Reference in XML* и *Обнаружены внешние Doctype в XML* позволяют блокировать внешние XML-сущности для предотвращения XXE-атак. Если XML начнет обращаться к внешнему файлу, сработает защита PT AF. Пример такого файла:

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE request [
<!ENTITY include SYSTEM "/etc/passwd">
]>
```

```
<request>
<description>&include;</description>
...
</request>
```

7.3.8.2. DOS-атака «Billion Laughs»

Когда событие обнаруживает DOS-атаку «Billion Laughs», оно срабатывает.

7.3.8.3. Ошибка валидации XSD

Событие срабатывает, когда на защищаемый URL отправляется XML, не соответствующий XSD-схеме. XSD-схема загружается во вкладке *Конфигурация -> Политики безопасности -> XML-схемы*. Чтобы определить защищаемый URL, нажмите кнопку *Добавить* в настройках модуля, а затем укажите в соответствующих полях URL, загруженную схему и тип ресурса – XSD.

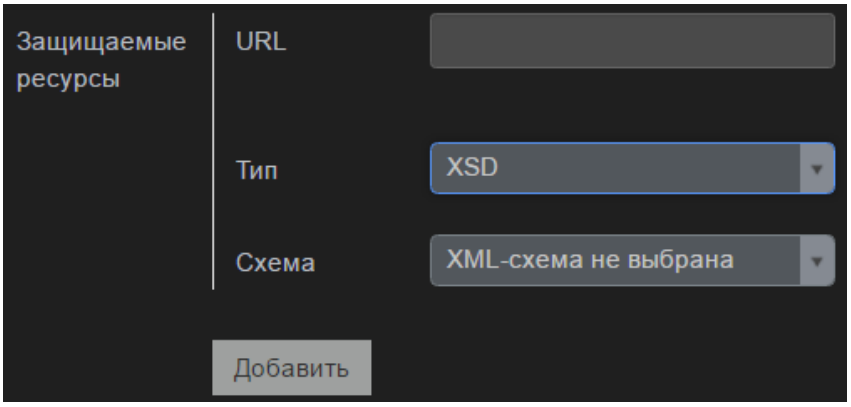


Рис. 181 – Настройка защищаемого ресурса

7.3.9. ICAP-интеграция

Интеграция с внешними системами возможна при помощи протокола ICAP. В первую очередь речь идет об интеграции с антивирусами, но это могут быть и другие системы, например, DLP. По умолчанию PT AF интегрирован с антивирусом ClamAV.

В данном модуле защиты возможны следующие действия:

- Malicious file upload attempt («Попытка загрузки зловредного файла»);
- Malicious file download attempt («Попытка скачивания зловредного файла»);
- ICAP Server Closed Connection («ICAP-сервер принудительно закрыл соединение»);
- ICAP Server sent error.

Нажмите кнопку *Управление ICAP-сервисами*, чтобы перейти на вкладку *Конфигурация -> ICAP-сервисы* и настроить онлайн-сервисы для проверки файлов на вирусы.

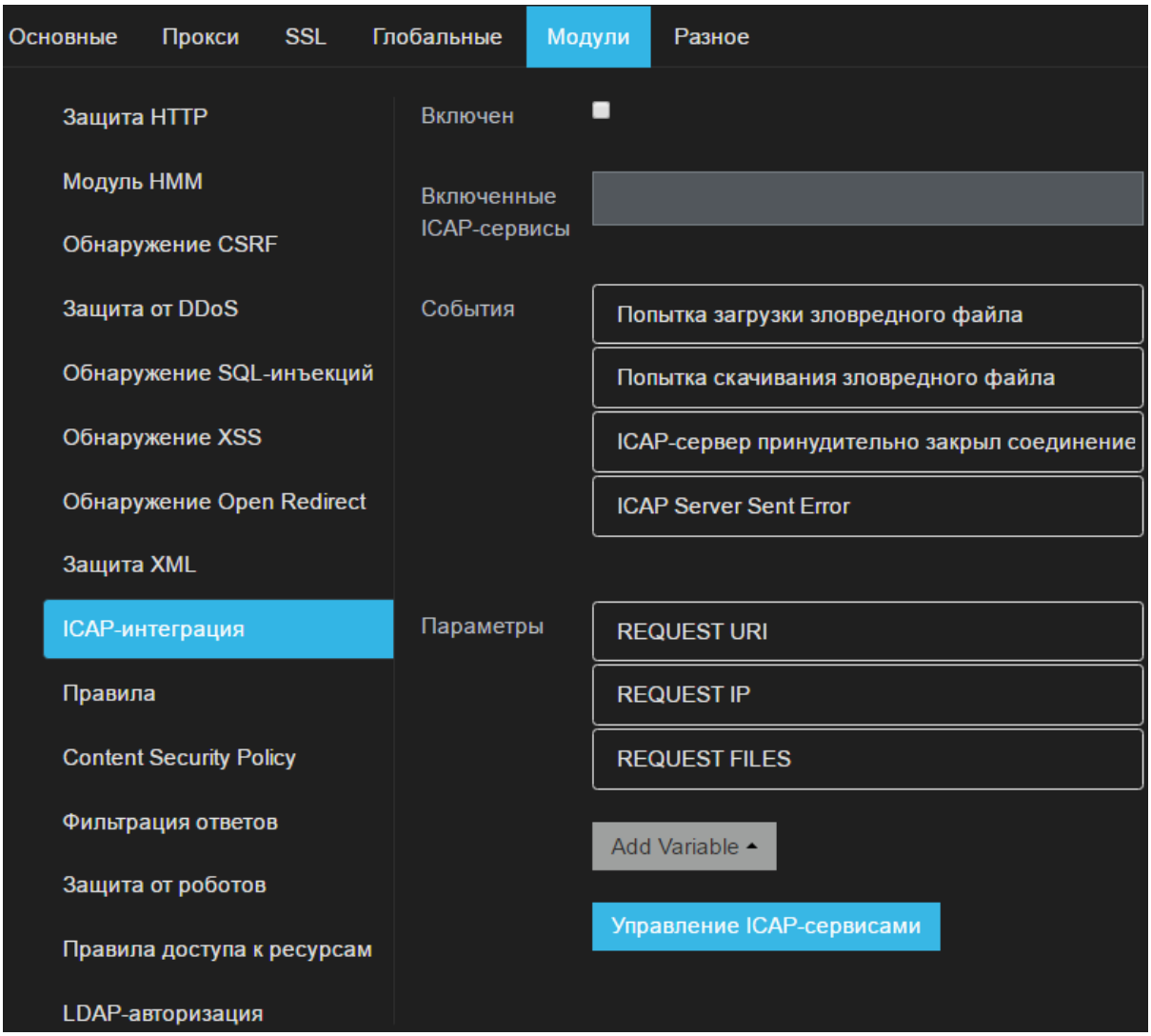


Рис. 182 – ICAP-интеграция

7.3.10. Правила

Модуль осуществляет проверку параметров на предмет совпадений с сигнатурами атак (пункт меню *Конфигурация -> Политики безопасности ->* [*Правила*](), позволяет идентифицировать вредоносные запросы.

Некоторые правила могут вызывать ложные срабатывания или обрабатываться слишком долго при работе с конкретным веб-приложением. В этом случае их рекомендуется отключить или модифицировать. Для редактирования правил нажмите соответствующую кнопку (Рис. 183).

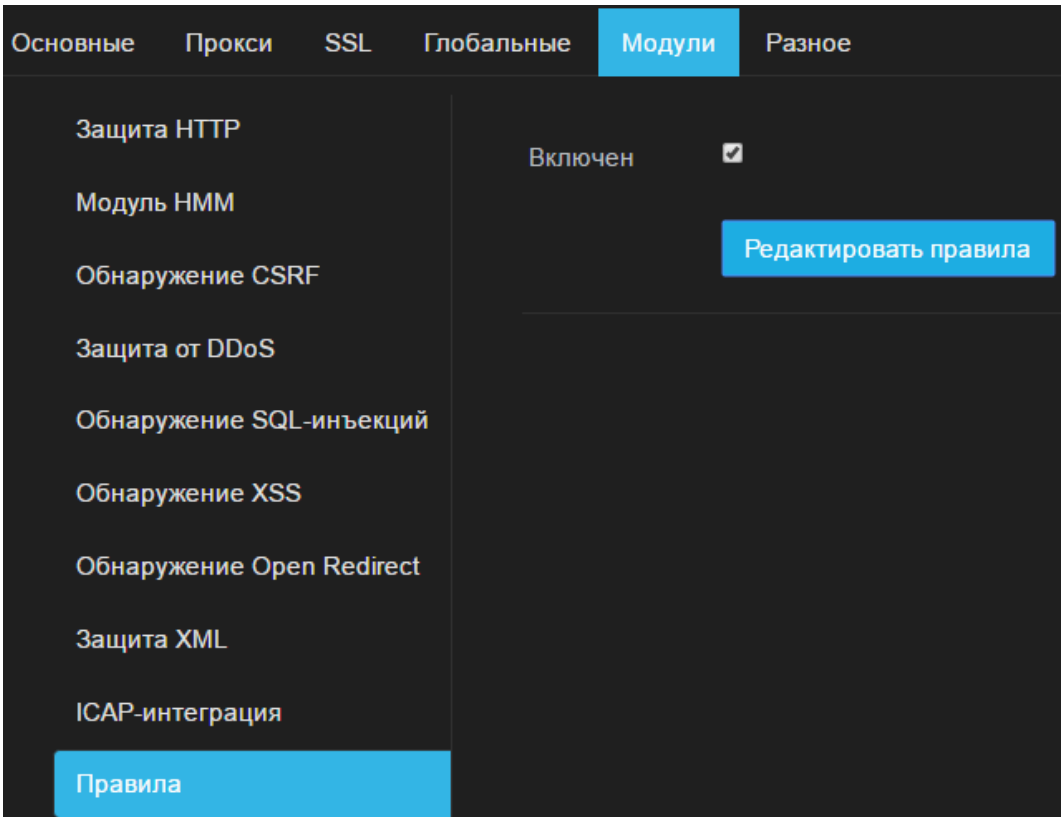


Рис. 183 – Правила

7.3.11. Content Security Policy

Модуль для автоматического обучения и применения политик стандарта Content Security Policy (более подробную информацию о стандарте можно найти на сайте http://www.w3.org/TR/CSP/).

Необходимо настроить следующие параметры:

- Путь для отправки CSP-отчетов (режим обучения) – путь, по которому обрабатываются отчеты о нарушениях политик Content Security Policy в режиме «Report Only» (стадия обучения);
- Путь для отправки CSP-отчетов (режим обнаружения) – путь, по которому обрабатываются отчеты о нарушениях политик Content Security Policy в режиме обнаружения;
- Защищаемые ресурсы – список защищаемых ресурсов, описываемых директивами CSP (script-src, frame-src, object-src, connect-src, font-src, img-src, media-src, style-src).

Дополнительные настройки модуля CSP доступны во вкладке *Конфигурация -> Политики безопасности -> Content Security Policy*.

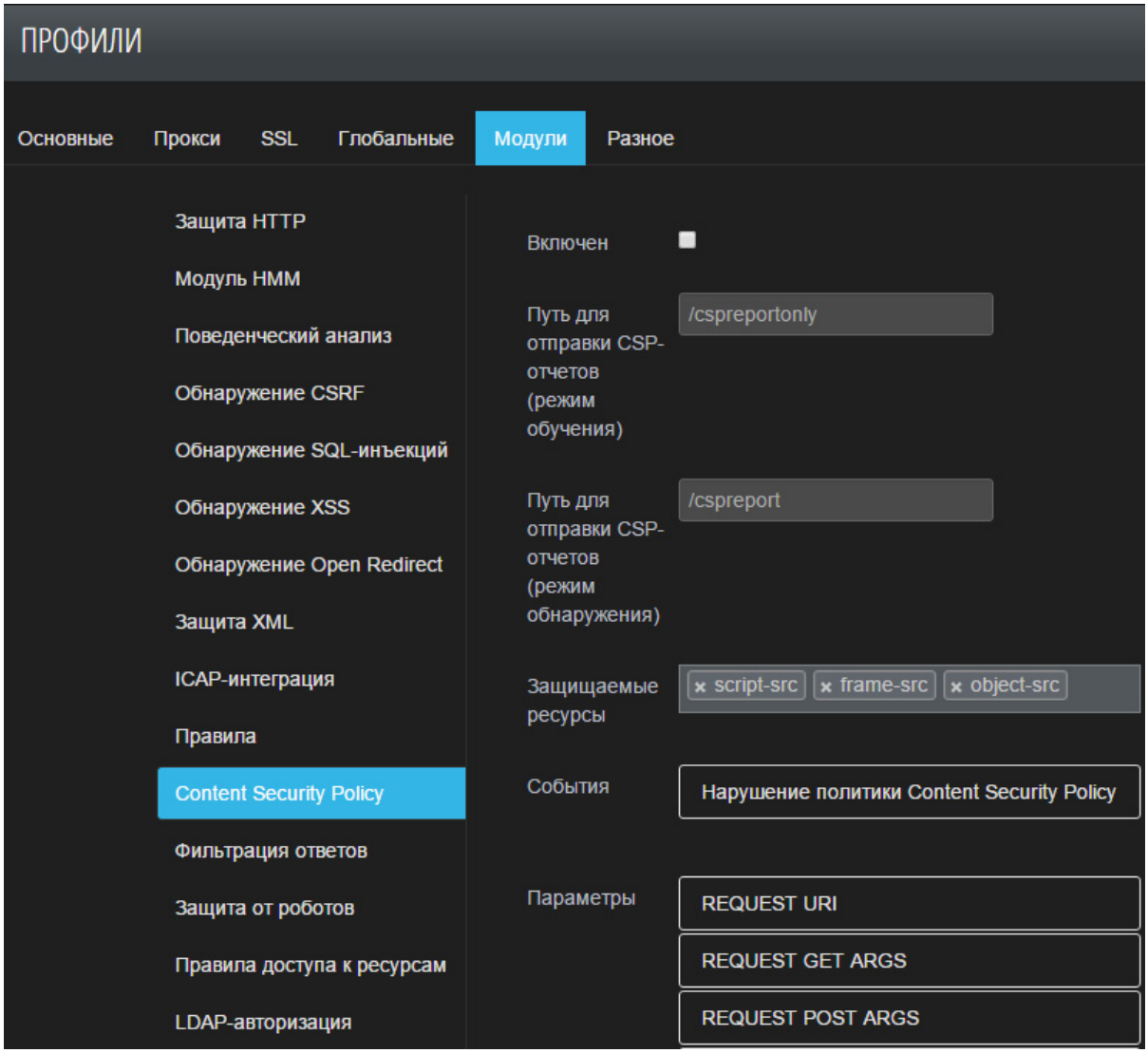


Рис. 184 – Модуль CSP

7.3.11.1. Нарушение политики Content Security Policy

Content Security Policy сообщает браузеру клиента, с каких внешних ресурсов, какие типы модулей разрешается загружать. При использовании CSP в запрос добавляется специальный заголовок Content-Security-Policy (для актуальных версий браузеров Chrome, Firefox, Opera) или X-Content-Security-Policy (для Internet Explorer 10). Этот заголовок содержит описание правил относительно того, что и откуда можно загружать.

По умолчанию, AF проверяет следующие директивы:

- script-src – определяет скрипты с каких ресурсов можно выполнять на страницах защищаемого ресурса;
- frame-src – определяет ресурсы, с которых можно загружать фреймы;
- object-src– определяет ресурсы, с которых можно загружать плагины.

7.3.12. Фильтрация ответов

Модуль для проверки и изменения HTTP-ответов с целью предотвращения утечки данных и обеспечения дополнительной защиты с помощью специальных HTTP-заголовков.

Для настройки доступны следующие параметры:

- Добавить флаг httpOnly в Cookie – добавление флага httpOnly для всех cookie-значений в заголовке Set-Cookie для защиты от XSS-атак. Такой флаг делает cookie недоступными для скриптов со стороны клиента (серверным скриптам доступ сохраняется). Это позволяет включить дополнительную защиту от XSS-атак;
- Значение HTTP-заголовка X-Frame-Options – значение заголовка X-Frame-Options для защиты от clickjacking-атак (по умолчанию SAMEORIGIN). Добавляет заголовок X-Frame-Options, который включает настройки отображения веб-сайта во фреймах. Существует несколько значений для этого заголовка (по умолчанию в настройках PT AF установлен SAMEORIGIN):
  - DENY – полный запрет просмотра сайта во фреймах, включая фреймы собственного сайта;
  - SAMEORIGIN – разрешает просмотр сайта в фреймах только на страницах своего сайта;
  - ALLOW-FROM uri – разрешает просмотр сайта во фреймах только на страницах указанного сайта.
- Значение HTTP-заголовка X-XSS-Protection – значение заголовка X-XSS-Protection, позволяющего настроить механизмы XSS-защиты в браузерах клиентов (по умолчанию 1 - включено);
- Значение HTTP-заголовка X-Content-Type-Options – значение заголовка X-Content-Type-Options, управляющего способами автоматического определения MIME-типов (по умолчанию nosniff) по содержимому файлов. Данный заголовок не допустит выполнение браузером JavaScript-кода, для которого сервером не выставляется соответствующий MIME-тип;
- Удалять HTTP-заголовок Server – удаление заголовка Server, раскрывающего данные о веб-сервере, если он был обнаружен в ответе сервера. Обычно, в заголовок Server пишется информация о используемом сервере и его модулях (например, Server: Apache/1.3.39 (Unix) PHP/5.3.27). Этот заголовок рекомендуется отключить, чтобы не сообщать злоумышленнику дополнительной информации;
- Удалять HTTP-заголовок X-Powered-By – удаление заголовка X-Powered-By, раскрывающего данные об интерпретаторе, если он был обнаружен в ответе сервера. Обычно, в этом заголовке пишется информация о технологиях, поддерживаемых веб-приложением (например, X-Powered-By: ASP.NET). Этот заголовок рекомендуется отключить, чтобы не сообщать злоумышленнику дополнительной информации;
- В поле *Шаблон утечки данных* указываются шаблоны, на которые проверяется ответ сервера. Например, если в ответе содержится «mySQL error with query», вероятно, что в ответ сервера включена информация с ошибкой базы данных. В ошибках может содержаться чувствительная информация, например, логины или пароли. Поэтому рекомендуется блокировать такие ответы, чтобы не давать злоумышленникам возможности получить доступ к чувствительной информации.

7.3.13. Защита от роботов

Модуль для настройки действий при обращении роботов и веб-пауков к указанным файлам и каталогам.

Необходимо настроить следующие параметры:

- *Filtered Paths* – список защищаемых страниц;

Примечание: для указания пути поддерживаются маски, перечисленные в Табл. 11.

Таблица 11. Маски

Маска	Описание
*	все символы в любом количестве
?	один любой символ
[seq]	любой символ seq
[!seq]	любой символ, кроме символов seq.

- *User-Agent для фильтрации* – список блокируемых User-Agent.

7.3.13.1. Робот посетил защищенную страницу

В поле *Filtered Paths* задаются разделы веб-приложения, к которым будет применяться данный модуль.

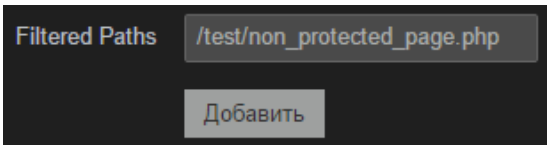


Рис. 185 –

В поле *User-Agent для фильтрации* указывается список идентификаторов роботов, по которым осуществляется фильтрация.

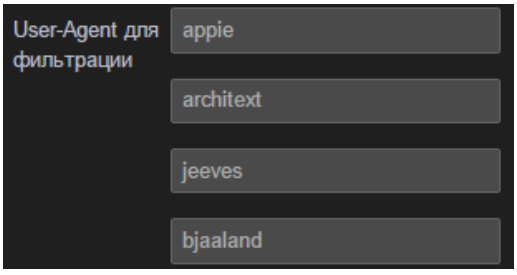


Рис. 186 –

7.3.13.2. Hacking Tool has been detected

Событие, которое срабатывает при обнаружении использования средств взлома, таких как Acunetix, BeEF, Burp Suite, OWASP ZAP, Fiddler и др. Событие работает через waf.js и отключается, если выключить waf.js во вкладке *Конфигурация -> Политики безопасности -> Профили -> Разное* (настройка *Inject waf.js into HTTP responses*).

7.3.13.3. Automated Web Bot Activity has been detected

Событие, которое срабатывает при обнаружении использования ботов (Selenium и headless-браузеров на основе PhantomJS). Событие работает через waf.js и отключается, если выключить waf.js во вкладке *Конфигурация -> Политики безопасности -> Профили -> Разное* (настройка *Inject waf.js into HTTP responses*).

7.3.14. Правила доступа к ресурсам

Веб-приложение передает PT Application Firewall информацию об авторизованных пользователях и позволяет осуществить контроль доступа для пользователей или групп пользователей.

На вкладке *Правила доступа к ресурсам (ACL Protector)* настраивается метод определения сессии (например, определенный параметр Cookie) и способ интеграции с защищаемым приложением (см. Рис. 188). На данный момент поддерживается взаимодействие по протоколу JSON.

Таким образом, PT Application Firewall может ограничить доступ конкретного пользователя к разделам приложения, так как знает, какая сессия ему соответствует, и к какой группе пользователей он относится.

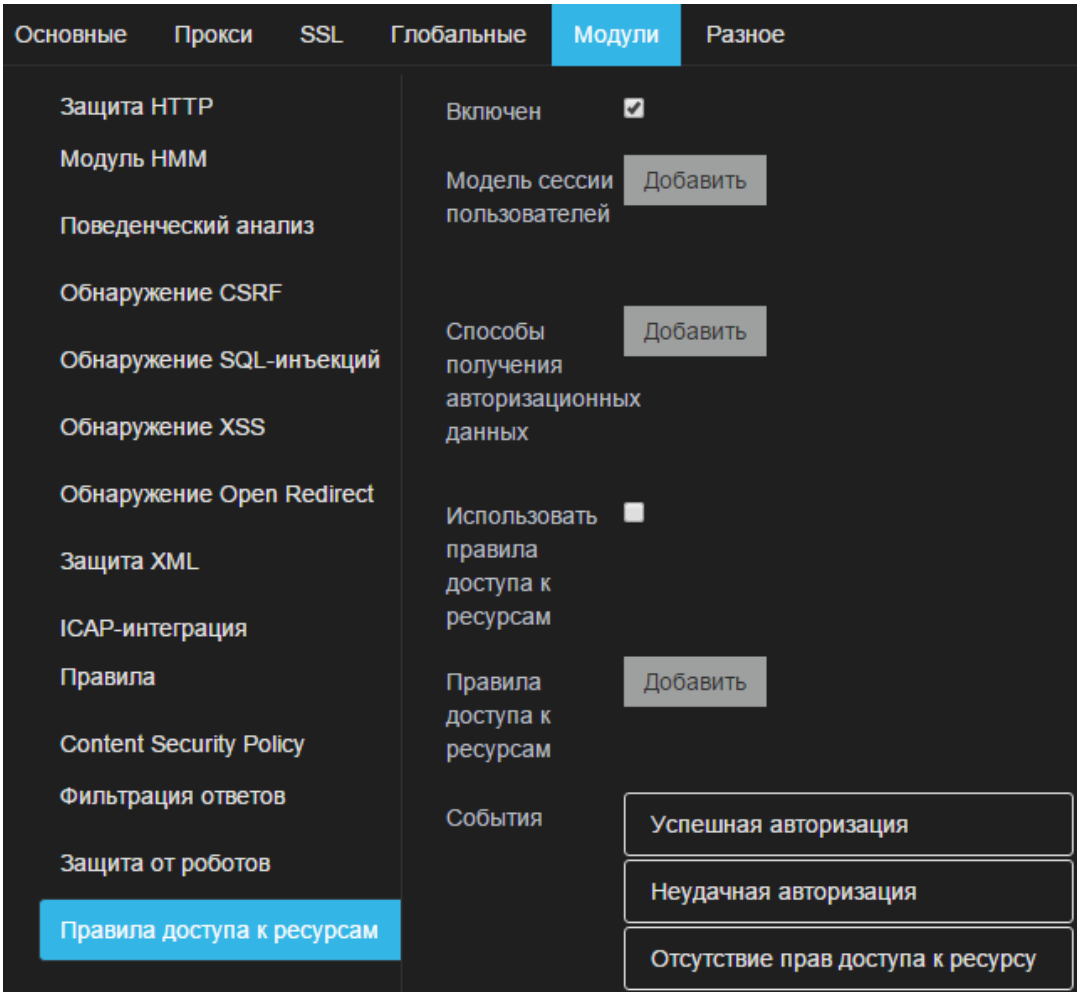


Рис. 187 – Правила доступа к ресурсам

Для настройки ограничений потребуется заполнить поля:

- *Путь* – раздел приложения, для которого будет применяться контроль доступа;
- *Приоритет* – уровень приоритета для обрабатываемых событий;
- *Страница авторизации* – присутствует ли по указанному пути форма авторизации;
- *Разрешенные/запрещенные ресурсы* – разрешить или запретить доступ к указанному разделу:
  - *Пространство имен* – пространство имен пользователей и групп;
  - *Тип* – выбор типа параметра (пользователь или группа);
  - *Имя* – название параметра;
  - *Права* – тип ограничений (чтение, запись, чтение и запись).

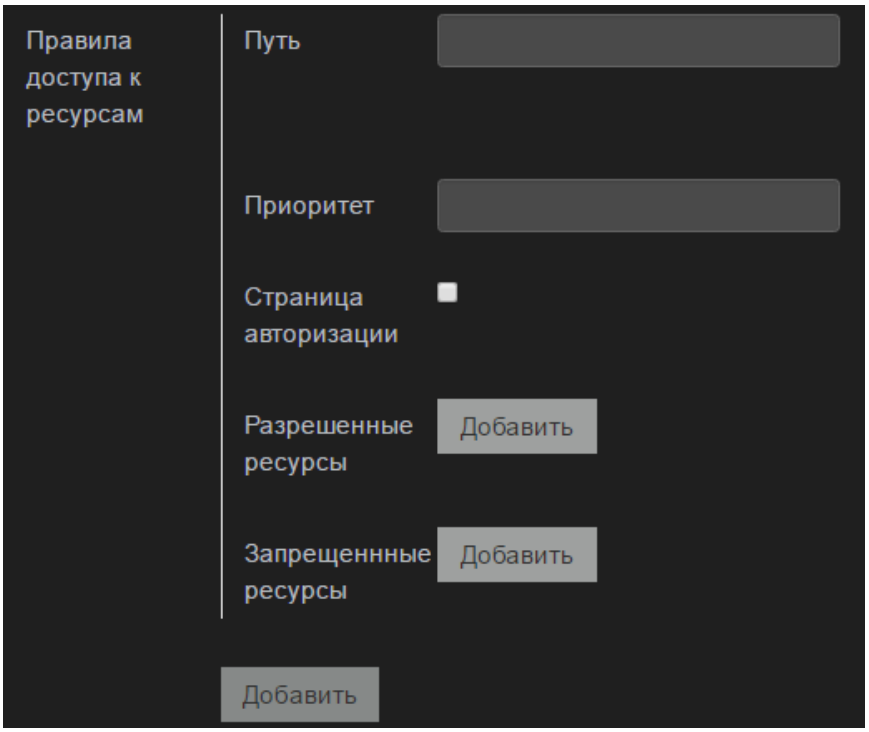


Рис. 188 – Поля настроек ограничений правила доступа к ресурсам

7.3.15. LDAP-авторизация

На данной вкладке следует задать параметры LDAP-авторизации. Настройки позволяют при работе с XML получить данные об авторизации пользователя и проверить их в Active Directory.

Необходимо настроить следующие параметры:

- Опция *Включен* отвечает за включение/выключение модуля.
- В поле *Таймаут кеша* следует установить время хранения кэша в секундах. Данный кэш хранится в nginx и используется для повторяющихся запросов.
- Параметр *Используемые LDAP-сервисы* отвечает за выбранные LDAP-сервисы, которые будут использовать в модуле. Добавить LDAP-сервис можно на вкладке *Конфигурации -> LDAP-сервисы* (см. главу «LDAP-сервисы»).

• URL параметры: переменные, получаемые из xml-запроса по xpath и по имени передаваемые в URL сервисов (Name, %user, %group и т.д.). Можно добавлять несколько параметров, при этом необходимо, чтобы имена переменных были в добавленных LDAP-сервисах.

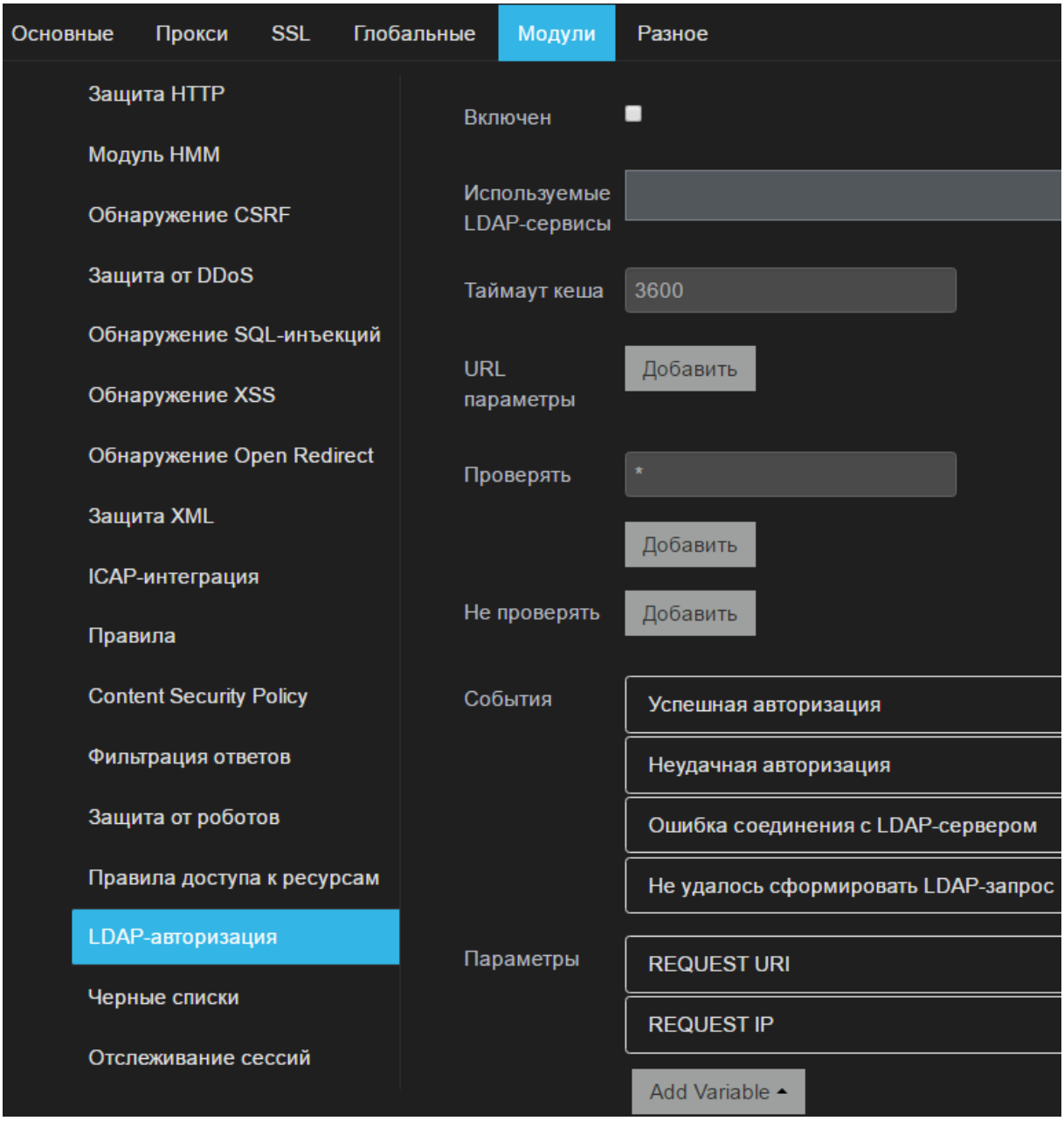


Рис. 189 – LDAP-авторизация

7.3.16. Черные списки

Данная настройка дает возможность работы с репутационными сервисами.

Здесь следует задать поведение для событий, если пользователь приходит с IP-адресов из списка Blacklist IP (см. главу «Черный список IP-адресов») или с сайта из Blacklist (см. главу «Черный список хостов»).

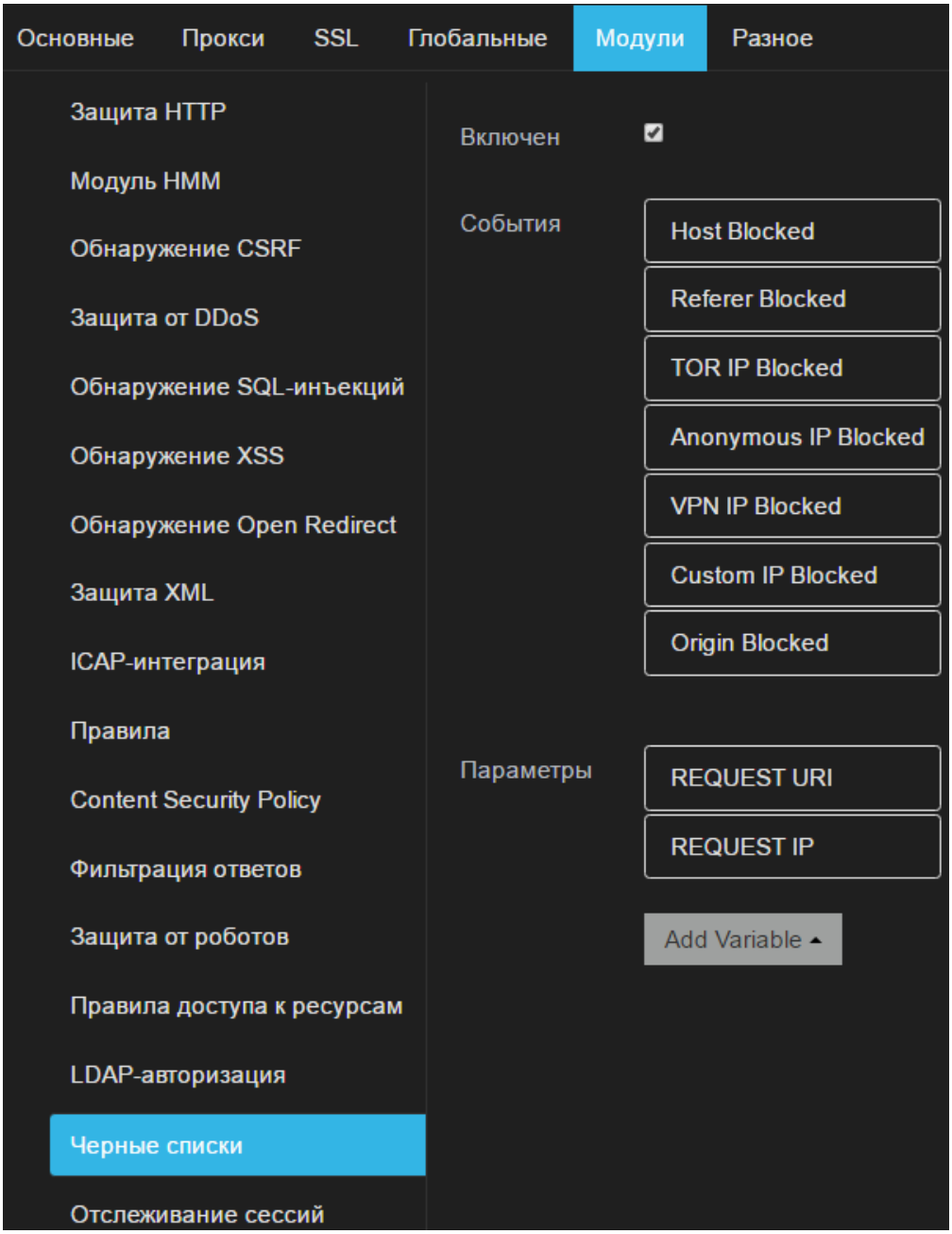


Рис. 190 – Черные списки

7.3.17. Отслеживание сессий

Данный модуль предназначен для идентификации пользователей. Модуль защиты позволяет выполнять более целенаправленную блокировку пользователей, по сравнению с блокировкой по IP, определять кражу выданной Cookie по нескольким критериям, а также позволяет обнаружить попытки подделки Cookie.

При получении запроса, не содержащего правильной Cookie, данный модуль создает новую Cookie и отправляет ее клиенту вместе с ответом от веб-сервера. Клиент должен использовать полученную Cookie в последующих обращениях к серверу. При создании Cookie генерируется событие *Creating Session*.

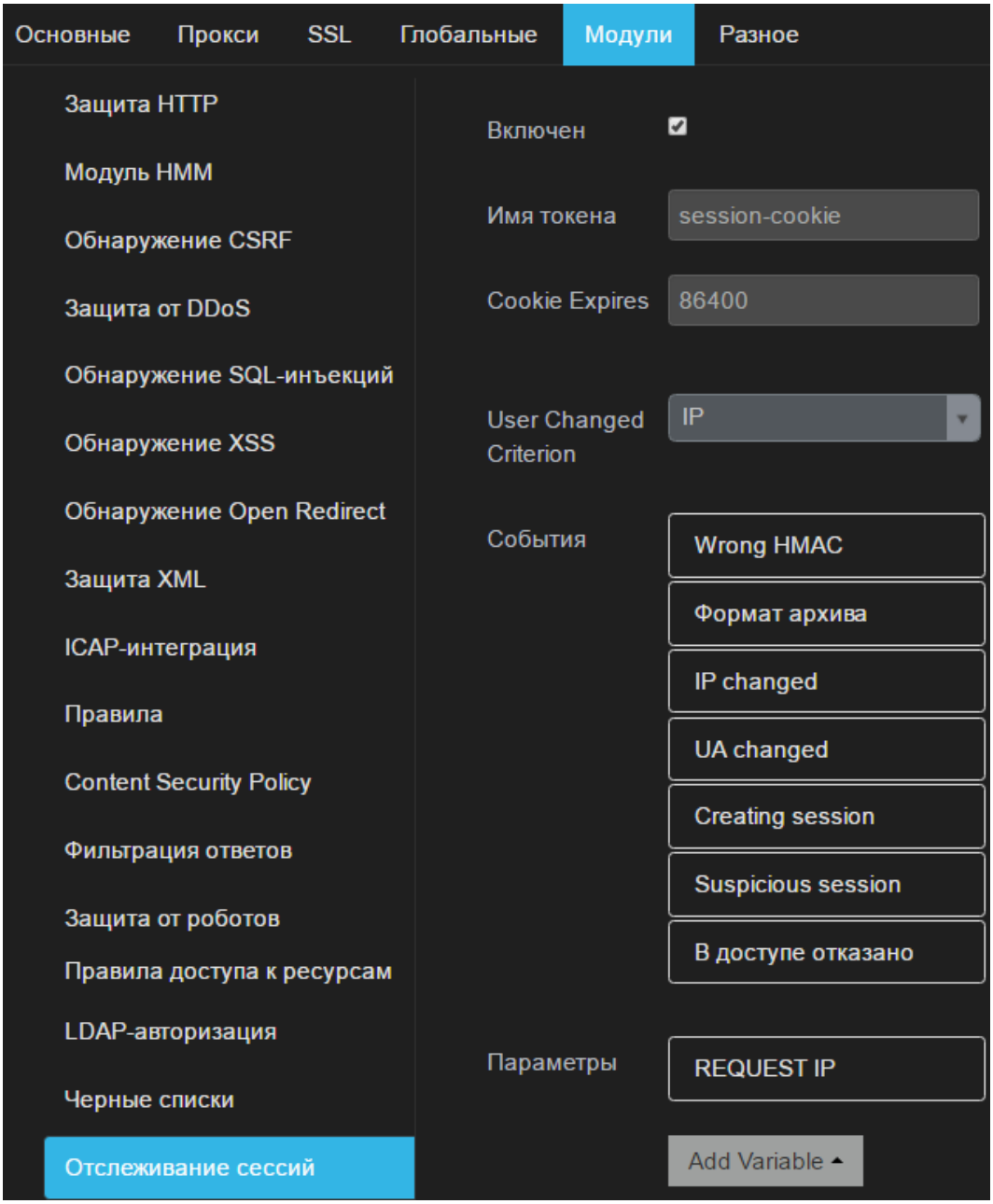


Рис. 191 – Отслеживание сессий

В данном модуле предусмотрены следующие настройки:

- Включен – отвечает за включение/выключение модуля;
- Имя Cookie – имя, которое модуль защиты будет присваивать Cookie, выдаваемую клиенту;
- Cookie Expires – время жизни Cookie;
- User Changed Criterion – настройка, позволяющая определить, по какому критерию для сессии будет создаваться событие IP changed. Возможные значения: IP, Город, Регион, Страна.

При необходимости настройте события:

- Wrong HMAC – срабатывает, если запрос содержит поврежденную Cookie. Поврежденная Cookie имеет правильную длину и содержит допустимые символы, но при ее формировании использован другой алгоритм создания Cookie;
- Wrong cookie format – срабатывает, если запрос пришел с неверным форматом Cookie: неверная длина или значение Cookie содержит недопустимый символ;
- IP changed – срабатывает в зависимости от критерия (см. User Changed Criterion), по которому следует определять, изменилось ли местонахождение пользователя;
- UA changed – срабатывает, если клиентский запрос содержит значение заголовка User-Agent отличное от того, для которого создавалась сессия. При обновлении версии User-Agent клиента обеспечено предотвращение ложного срабатывания;
- Creating session – срабатывает, если запрос пришел без Cookie, и создается новая Cookie;
- Suspicious session – срабатывает, если сессия находится в списке *Suspicious Sessions* (см. главу «Подозрительные сессии»). Действие *Session is suspicious - cookie* заносит сессию в список подозрительных.
- Session expired - срабатывает, если клиент прислал запрос, содержащий Cookie с истекшим временем жизни.

В модуле можно настроить переменную:

- Request IP - фильтр по IP.

7.4. Настройка пользовательских правил обработки трафика

В данном разделе представлена информацию о создании и настройке правил обработки трафика (см. Рис. 192). Необходимо задать следующие параметры:

- Профиль – профиль, с которым ассоциировано правило;
- Теги – метки, позволяющие группировать правила по типам атак и уязвимостей;
- Описание – текстовое описание правила;
- Регулярное выражение – регулярное выражение, по которому производится поиск;
- Воздействие – вес правила в числовом выражении;
- Включен – включение/отключение правила.

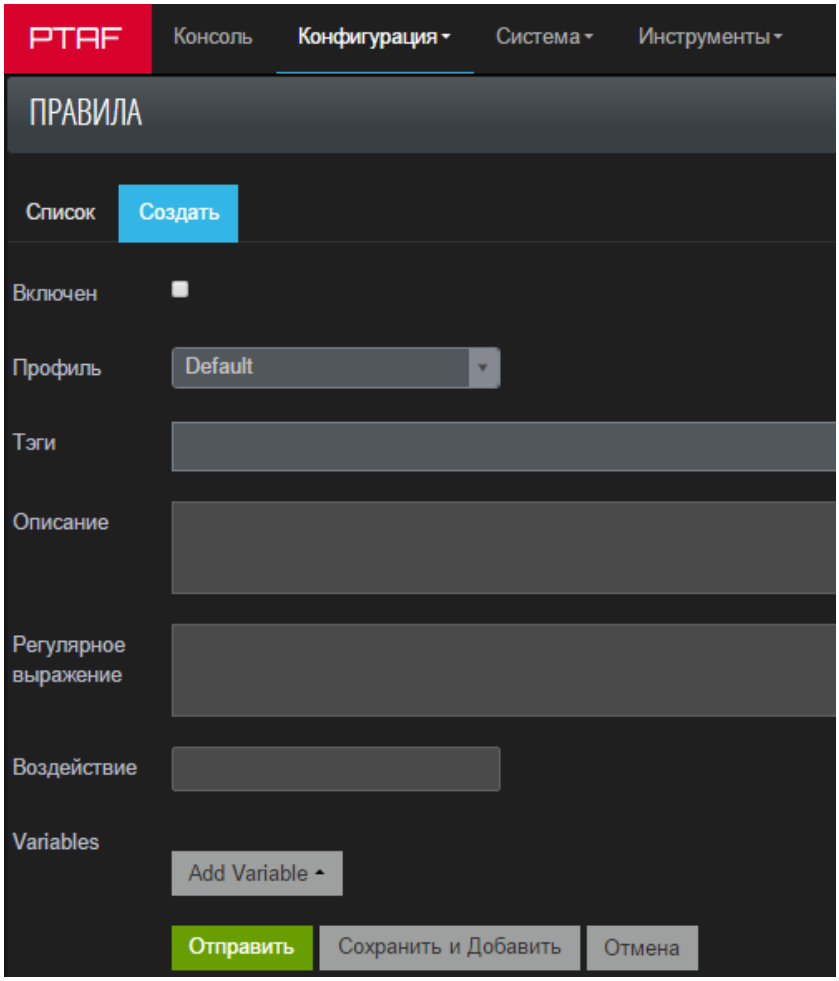


Рис. 192 – Создание и редактирование правил

8. Примеры конфигурации

В данной главе представлено описание некоторых сценариев использования системы и дана краткая информация по настройкам интерфейса. Полное описание функциональности см. в главе «Интерфейс и работа с системой».

8.1. Конфигурация работы

8.1.1. Настройка мониторинга (Sniffer)

- Переведите интерфейс в режим SPAN:

```
wsc
wsc> if span eth1
wsc> config commit
wsc> if mark eth1
wsc> config sync
```

- Добавьте роль SPAN на вкладке *Конфигурация -> Сеть -> Алиасы сетевых интерфейсов;*

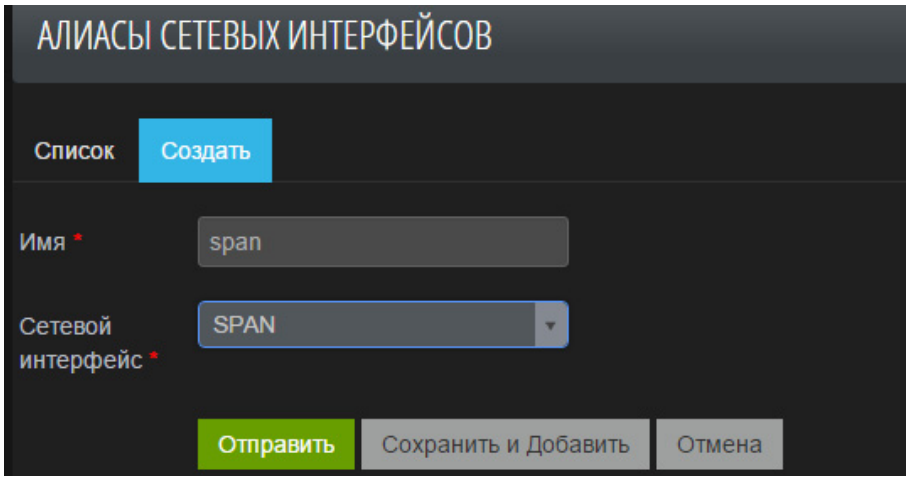


Рис. 193 – Добавление роли

- Добавьте созданную роль для интерфейса eth1 на вкладке К*онфигурация -> Сеть -> Шлюзы -> Сеть;*

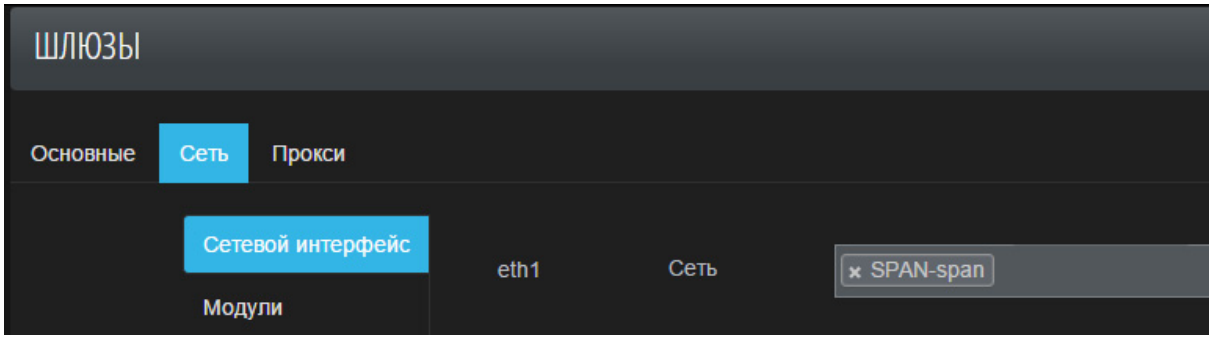


Рис. 194 – Добавление роли для интерфейса

• Настройте параметры мониторинга на вкладке *Конфигурация -> Сеть -> Сниффер*:

- Интерфейс, с которого следует снимать трафик (опция *Сетевой интерфейс*);
- IP-адреса приложений, если необходимо анализировать все адреса назначения;
- Порты приложений;
- SSL-ключ, если используется HTTPS;

Примечание: расшифровка сниффером SSL-трафика возможна только в то случае, когда на стороне защищаемого приложения в качестве протокола обмена ключами используются определенные наборы шифров, список которых представлен в главе 5.1.2.

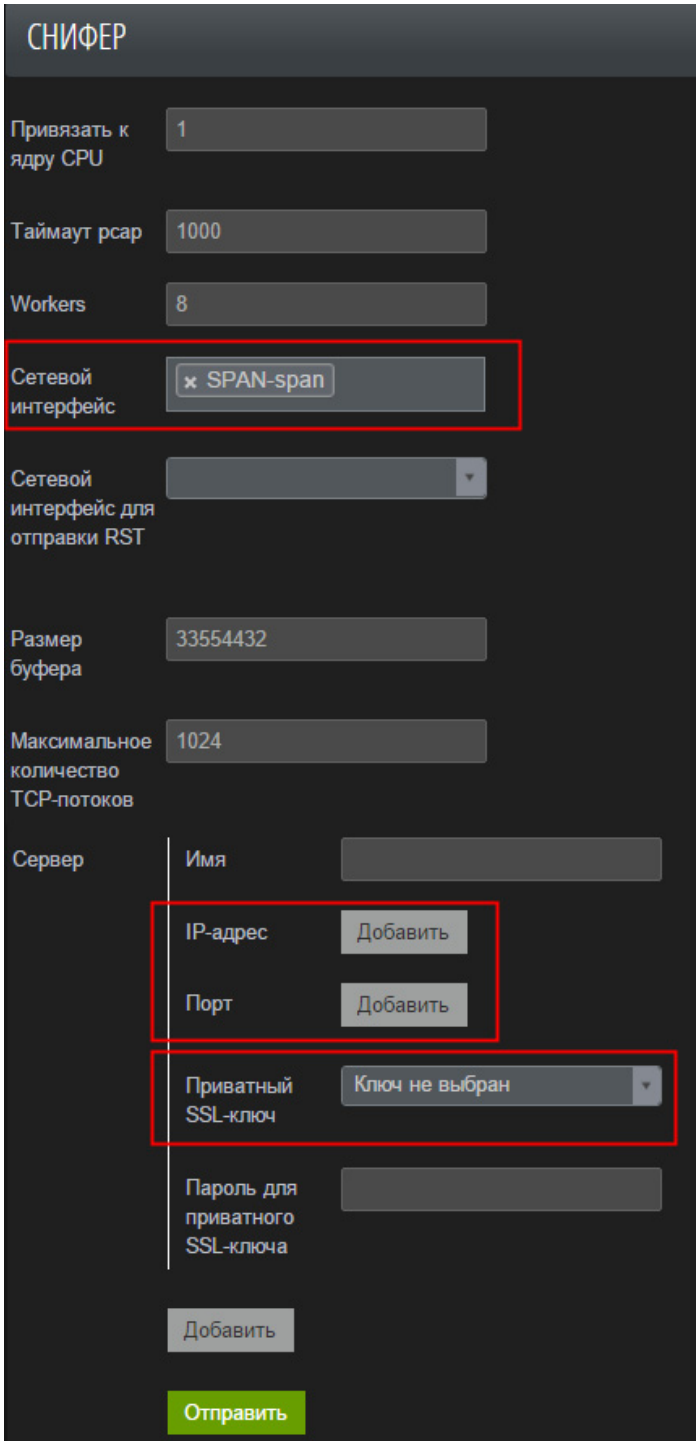


Рис. 195 – Настройка параметров мониторинга

• На вкладке *Конфигурация -> Политики безопасности -> Профили -> Основные* включите опцию *Сниффер подключен*.

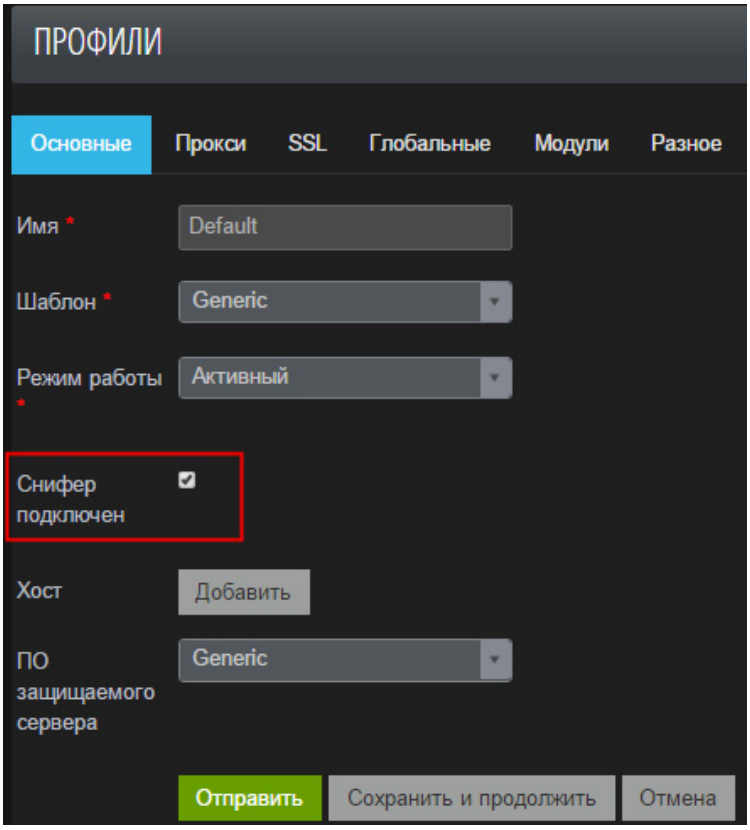


Рис. 196 – Редактирование профиля

8.1.2. Настройка режима реверс-прокси

- Добавьте роль WAN на вкладке *Конфигурация -> Сеть -> Алиасы сетевых интерфейсов*;

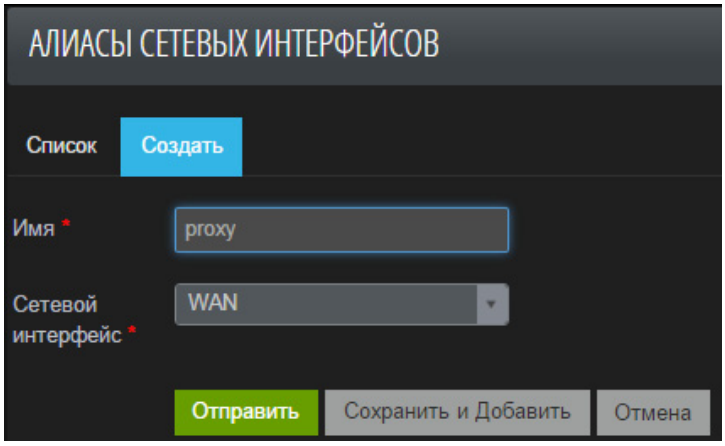


Рис. 197 – Добавление роли

• Добавить порты, разрешенные для подключения пользователей, отредактировав созданный алиас;

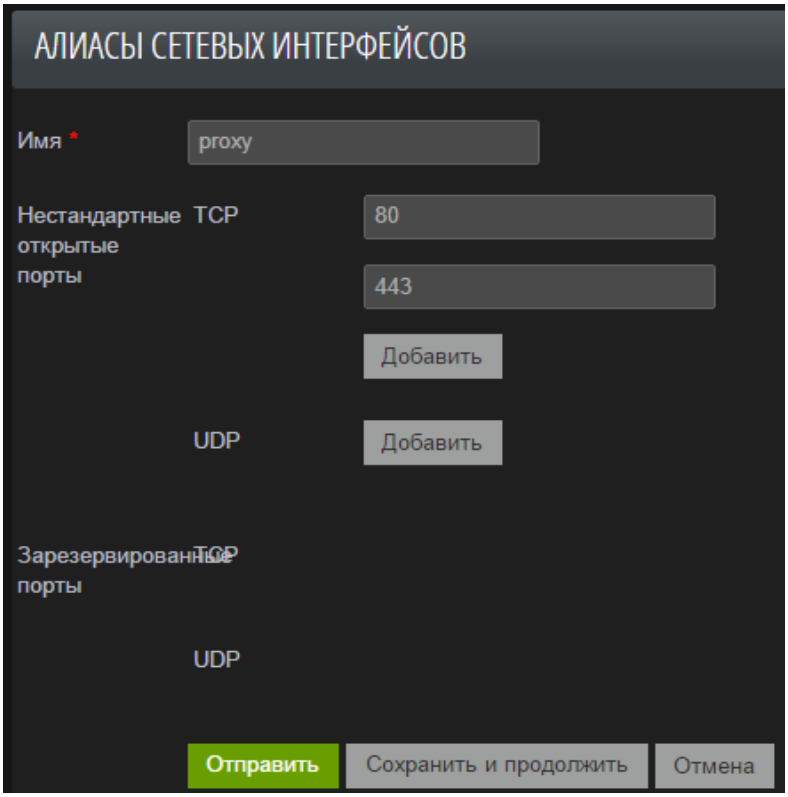


Рис. 198 – Добавление нестандартных открытых портов

• Добавьте новый сервер приложения на вкладке *Конфигурация -> Сеть -> Группы серверов*:

- Укажите имя для группы серверов;
- Укажите IP-адрес;

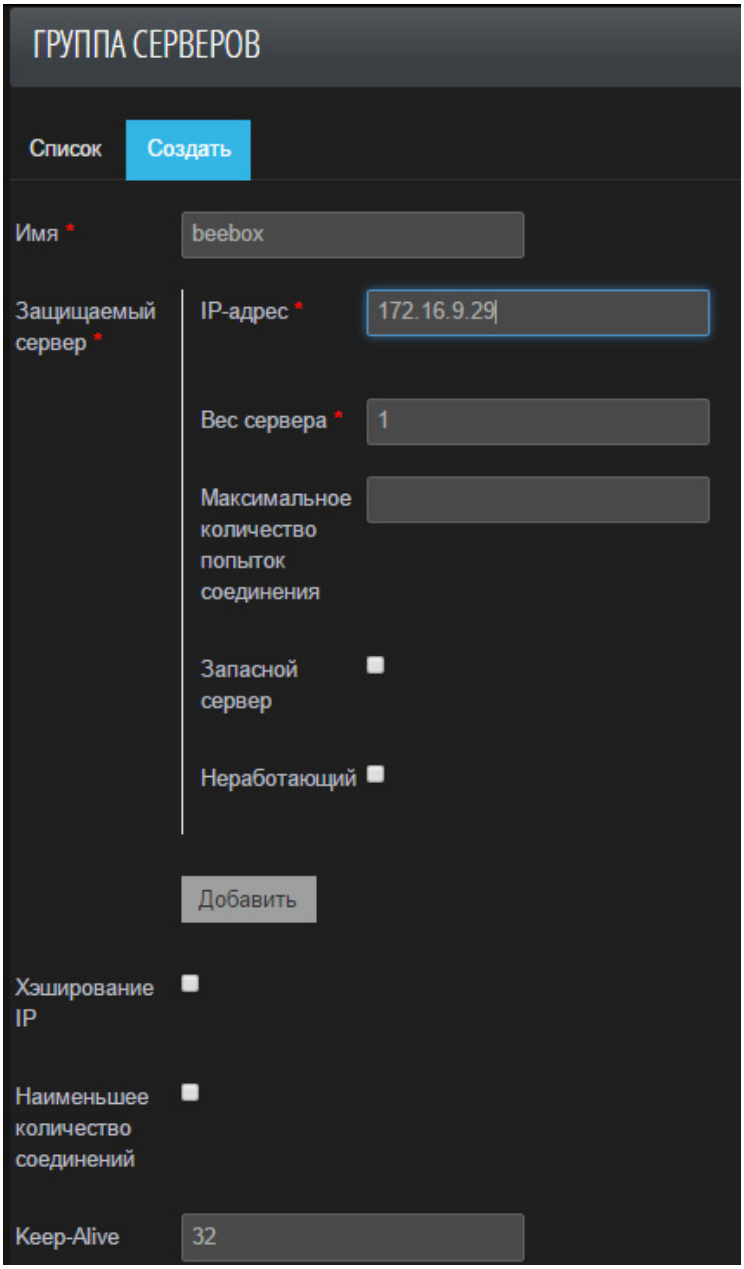


Рис. 199 – Добавление нового сервера приложения

• На вкладке *Конфигурация -> Политики безопасности - > Профили -> Прокси* укажите:

- Интерфейс, на котором будет открыт порт: WAN;

- Порт, который будет открыт на PTAF для входящих соединений (должен быть в списке открытых портов для WAN);

- Включите опцию *Default Server*, если профиль должен обрабатывать запросы на указанный порт и все имена узлов, не заданные в других профилях;

- Группу серверов, на которую будут перенаправляться запросы.

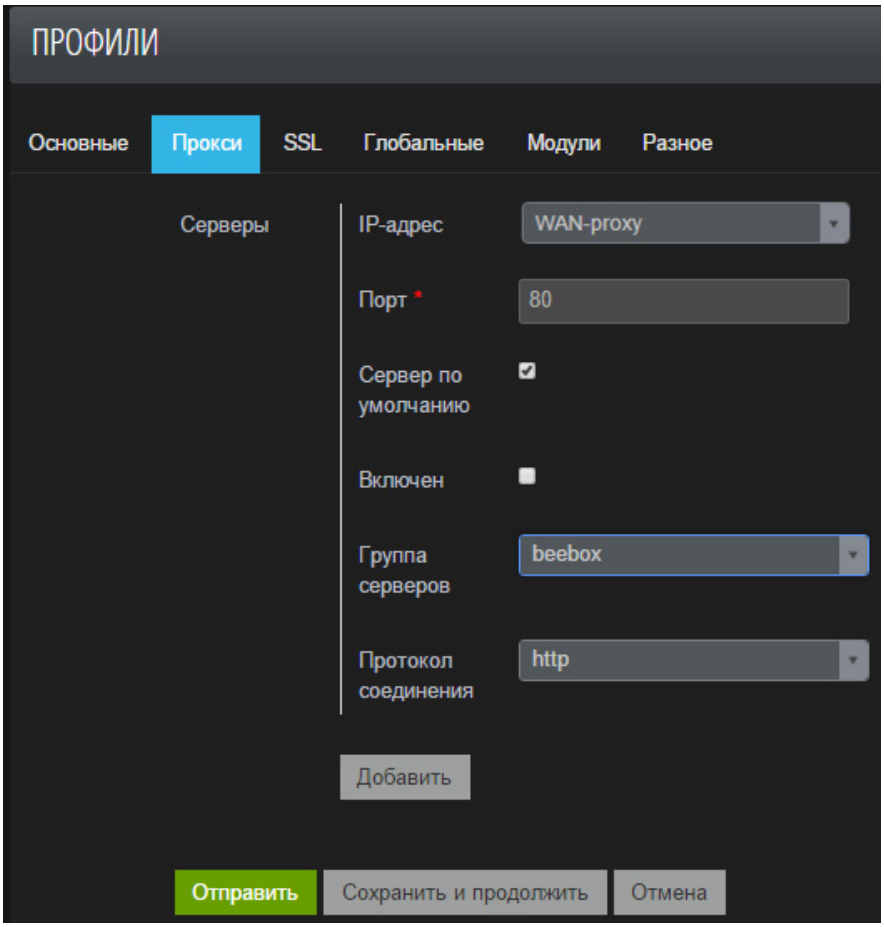


Рис. 200 – Редактирование профиля

8.1.3. Включение SSL для реверс-прокси

- При необходимости, сгенерируйте сертификат, приватный ключ:

```
sudo openssl req -x509 -nodes -days 365 -newkey rsa:4096    -keyout
nginx.key -out nginx.crt

Country Name (2 letter code) [AU]:RU
State or Province Name (full name) [Some-State]:Moscow
Locality Name (eg, city) []:Moscow
Organization Name (eg, company) [Internet Widgits Pty Ltd]:PT
Organizational Unit Name (eg, section) []:
Common Name (e.g. server FQDN or YOUR name) []:
Email Address []:
```

- Перейдите на вкладку *Конфигурация -> SSL-сертификаты и ключи*;
- Загрузите в каталог certificates сертификат;
- Загрузите в каталог keys ключ;

• Перейдите на вкладку *Конфигурация -> Сеть -> Группа серверов* и задайте SSL-порт приложения;

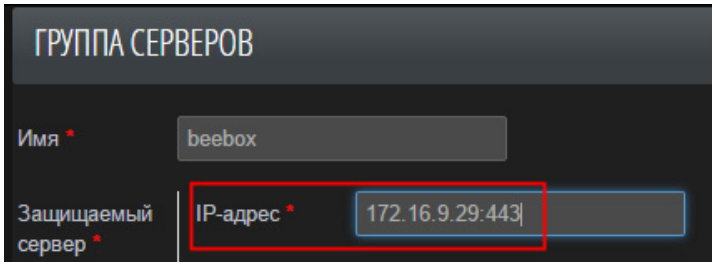


Рис. 201 – Редактирование группы серверов

• Перейдите на вкладку *Конфигурация -> Политики безопасности -> Профили -> Прокси*:

- Укажите порт для SSL;
- Включите опцию *Включен* (SSL enabled) для порта приложения.

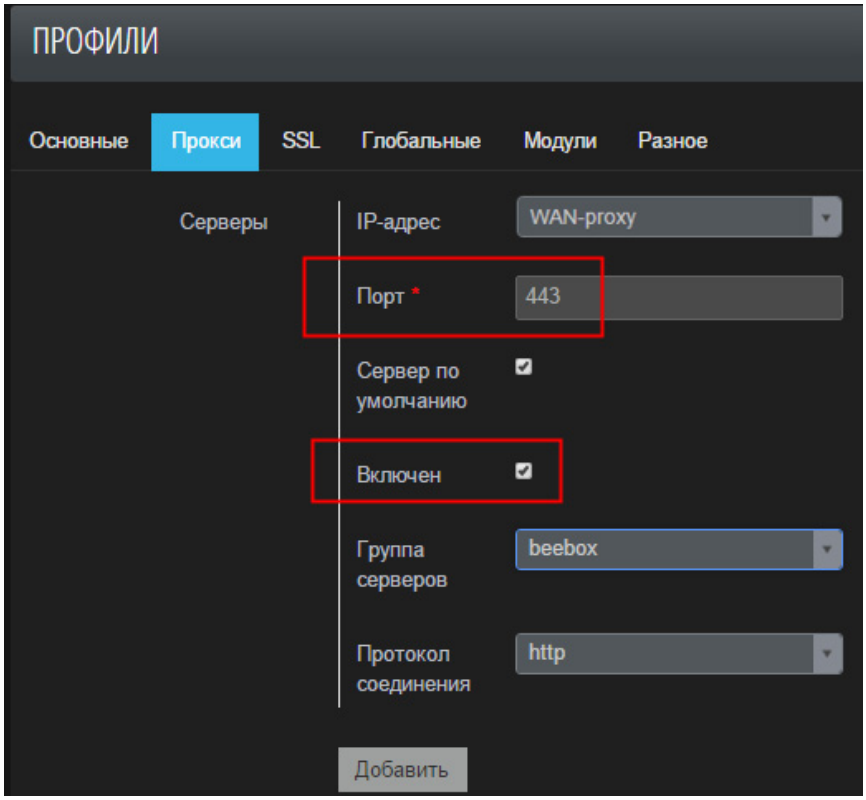


Рис. 202 – Редактирование профиля

• Перейдите на вкладку *Конфигурация -> Политики безопасности -> Профили -> SSL*, и укажите загруженные сертификаты и ключи, нажмите «Использовать рекомендованные настройки», чтобы заполнить параметры SSL.

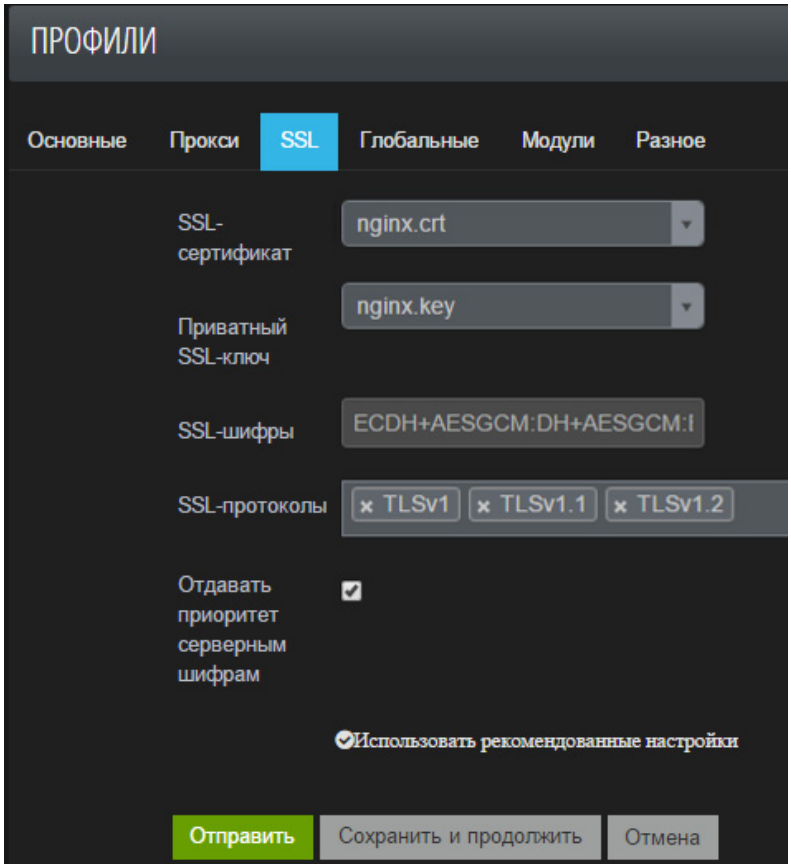


Рис. 203 – Редактирование SSL настроек профиля

8.1.4. Анализ журналов в режиме Forensics

- Перейдите на вкладку *Инструменты -> Анализ файлов журналирования;*
- Загрузите журнал;
- Нажмите *Определить*, чтобы определить формат журнала;

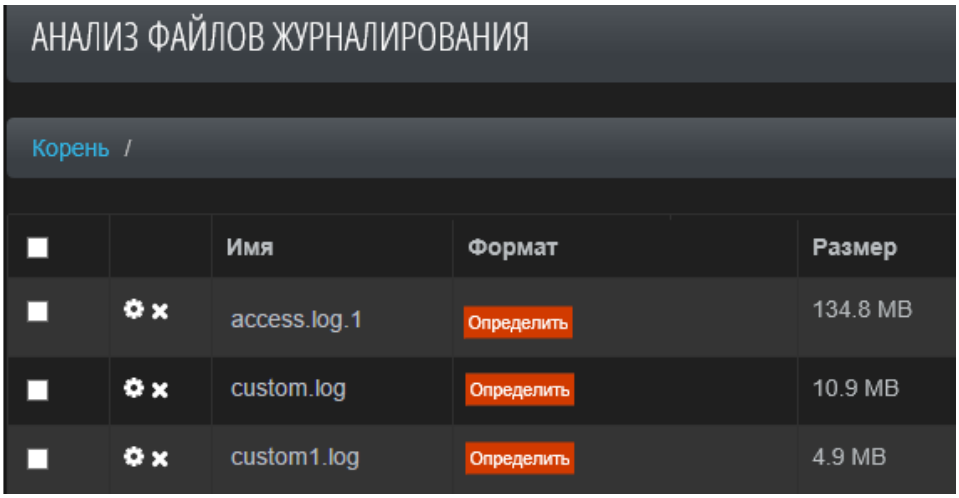


Рис. 204 – Загрузка журнала

- Если формат не определился, выполните следующие шаги:
  - Перейдите в настройки;

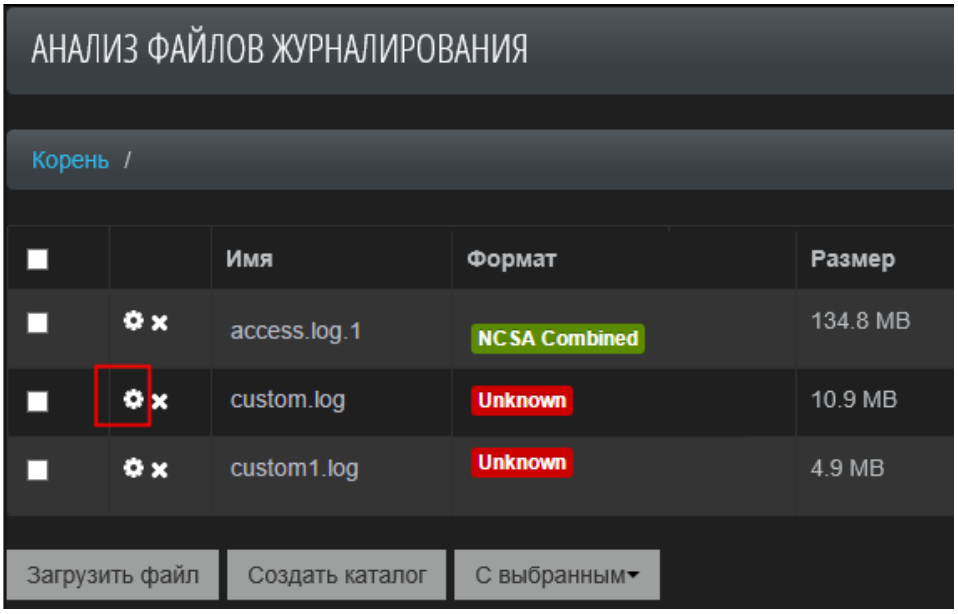


Рис. 205 – Переход к настройкам журнала

  - Добавьте описание и задайте формат;
  - Подтвердите правильность формата нажатием кнопки *Проверить формат*.

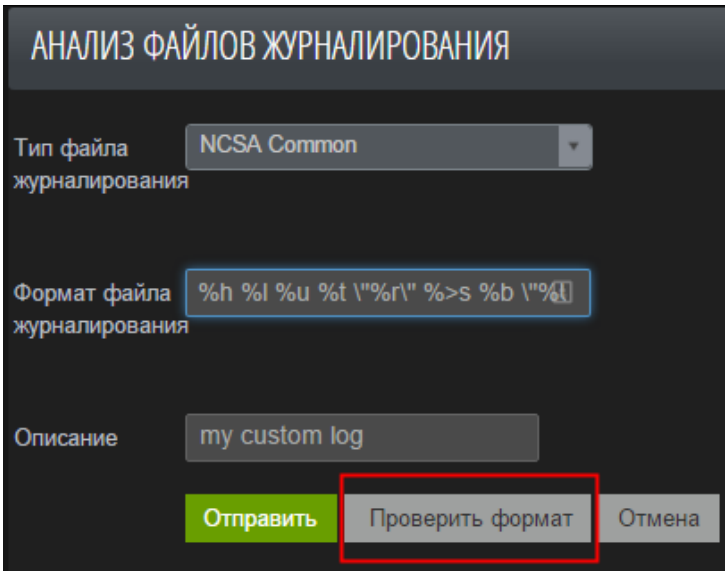


Рис. 206 – Настройка журнала

Примечание. Для записи формата используется apache mod_log_config, подробнее с описанием возможных полей можно ознакомиться в документации.
Для примера, типу журнала:
1.2.3.4 - - [24/Oct/2015:03:27:31 +0400] "GET / HTTP/1.1" 200 27015 "-" "Mozilla/5.0 (Windows NT 6.1; rv:21.0) Gecko/20100101 Firefox/21.0" "-"
Соответствует такой формат:

%h %l %u %t \"%r\" %>s %b \"%U\" \"%{User-agent}i\" \"-\"
Для этого типа журнала:
[14/Apr/2015:03:12:55 +0300] 1.7.4.3 - - - my.testapp.ru to: 5.5.3.3:84: HEAD /register/step1 HTTP/1.1 upstream_response_time 0.004 msec 1434240775.028 request_time 0.004
Подходит следующий формат:
%t %h - - - %{Host}o to: %a:%p: %m %U %H upstream_response_time %T msec %{msec}T request_time %T

- Выберите журналы для анализа;
- Выполните команду *С выбранным -> Обработать*;

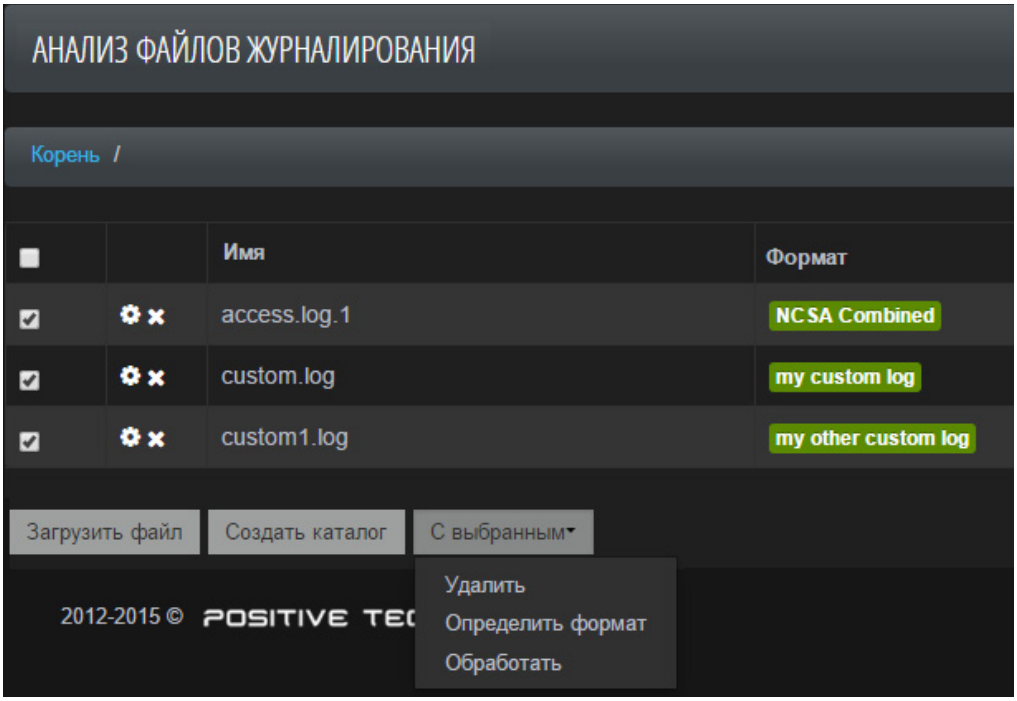


Рис. 207 – Выбор команды обработки файлов журналирования

- Для просмотра атак, обнаруженных в процессе анализа журналов, перейдите по ссылке, размещенной в поле *Идентификатор задания*.

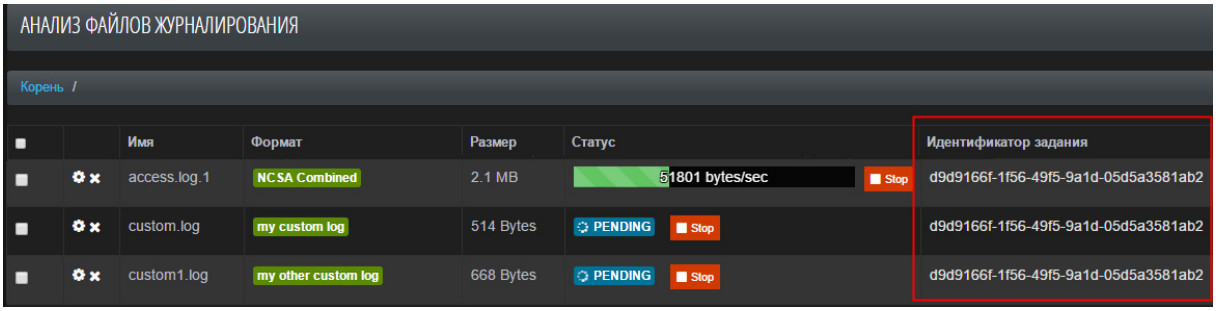


Рис. 208 – Отображение идентификатора задания

8.1.5. Добавление исключения

- Нажмите кнопку *Исключить* на панели *Attacks* в строке выбранной атаки;

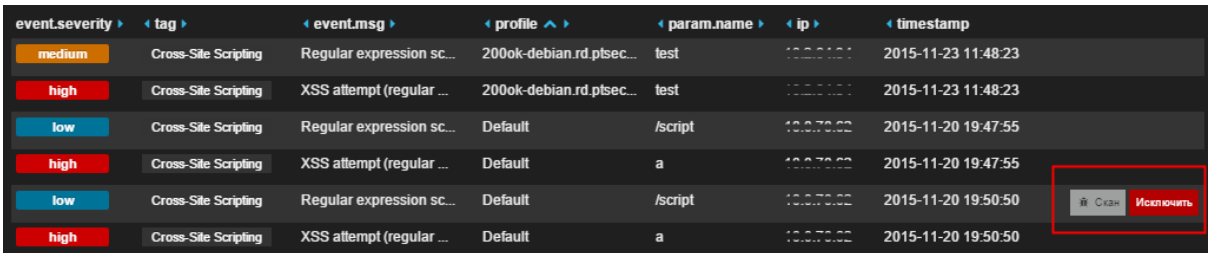


Рис. 209 – Панель атак

- В окне *Attack Exclude Dialog* выберите список исключений для IP-адреса:
  - Для одного модуля защиты (Protectors excludes);
  - Для всех модулей (Global excludes);
  - Для отдельных правил (Rule excludes) – применительно только к событиям защитного механизма Rule engine.
- Выберите список исключений для параметра:
  - Для одного модуля защиты (Protectors excludes);
  - Для всех модулей (Global excludes);
  - Для отдельных правил (Rule excludes) – применительно только к событиям защитного механизма Rule engine;
- Выберите HMM model actions (справедливо только для событий HMM):
  - Выключить модель;
  - Переобучить модель;
- Выберите параметры исключения атак в базе данных:
  - Путь URL (path) – поддерживает regex;
  - Профиль (profile);
  - Защитный механизм (module);
  - Имя параметра (param.name) – поддерживает regex;
  - Источник параметра (param.src).
- Если требуется удалить атаки из БД, а не просто пометить их как исключенные (с возможностью восстановления), включите опцию *Исключить навсегда*.

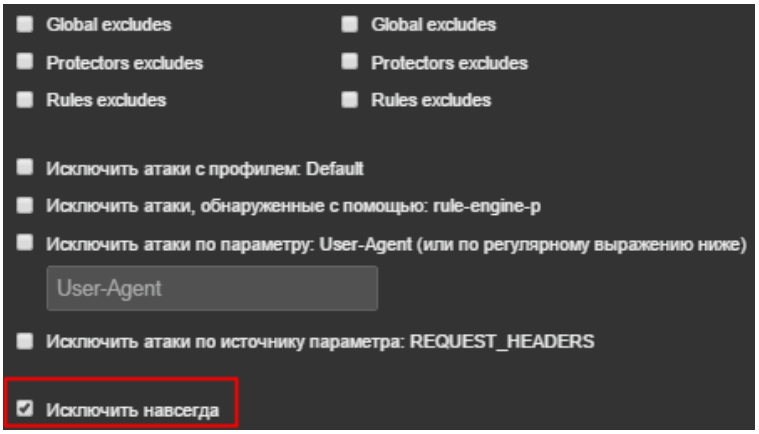


Рис. 210 – Удаление атаки из БД

8.1.6. Blackbox-сканер

• В строке с атакой в панели *Attacks* нажмите кнопку *Скан*;

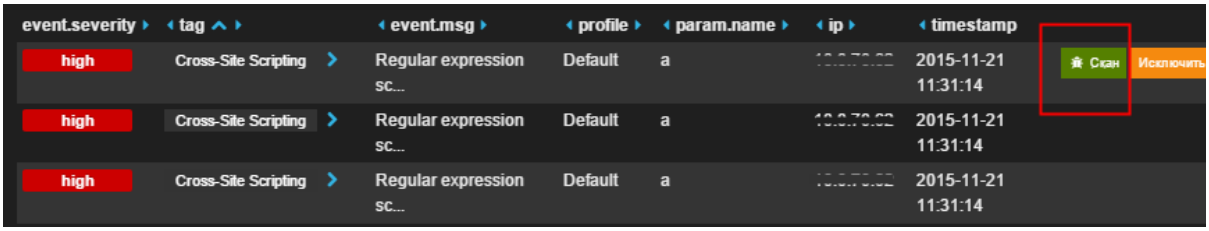


Рис. 211 – Запуск сканирования

• В появившемся окне укажите адрес приложения без защиты PTAF (например, тот же, что указан в группе серверов);

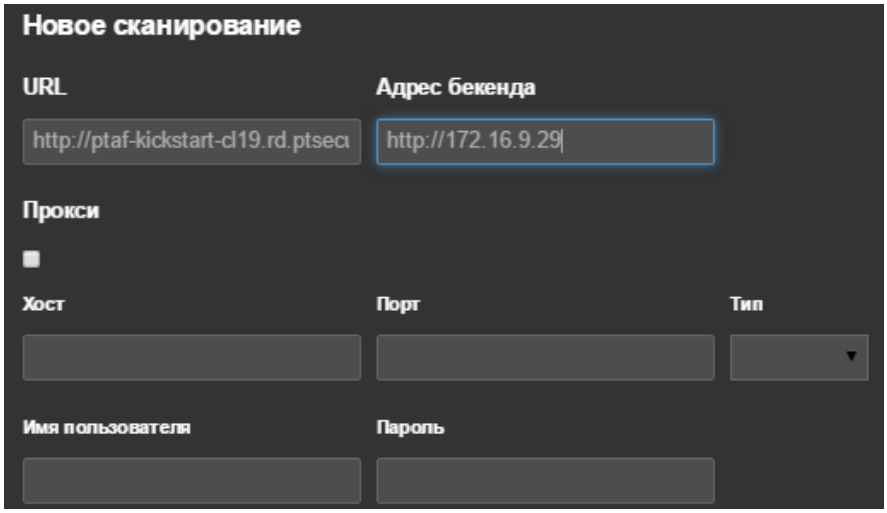


Рис. 212 – Настройка нового сканирования

• Нажать кнопку *Сканировать*, и посмотрите результат:

- Стоп – сканирование в процессе;

- Safe – уязвимость не обнаружена;
- Alert – уязвимость обнаружена;
- Error – произошла ошибка в процессе сканирования.

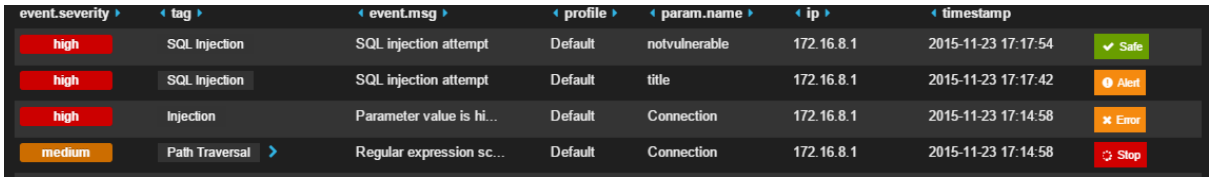


Рис. 213 – Просмотр результата сканирования

8.1.7. Обработка файлов журналирования веб-сервера и pcap-файлов

PT AF может анализировать трафик в автономном режиме путем обработки файла журнала веб-сервера или журнала сетевых данных в формате pcap. Такой режим позволяет проводить расследования инцидентов с веб-приложениями, которые не были защищены PT AF или были защищены другим веб Application Firewall.

Для осуществления оффлайн-анализа файлов журналирования веб-сервера или pcap-файлов необходимо выполнить следующие шаги:

1. Перейти в раздел меню *Инструменты -> Анализ файлов журналирования*, а затем нажать кнопку *Загрузить файл* (см. Рис. 214);

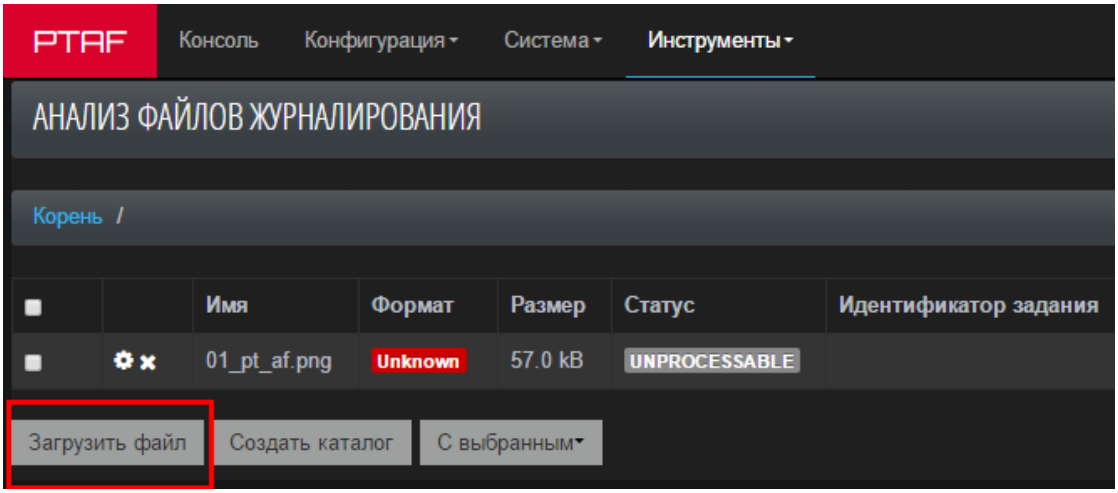


Рис. 214 – Анализ файлов журналирования

2. Загрузить файл, нажать кнопку *Отправить*.

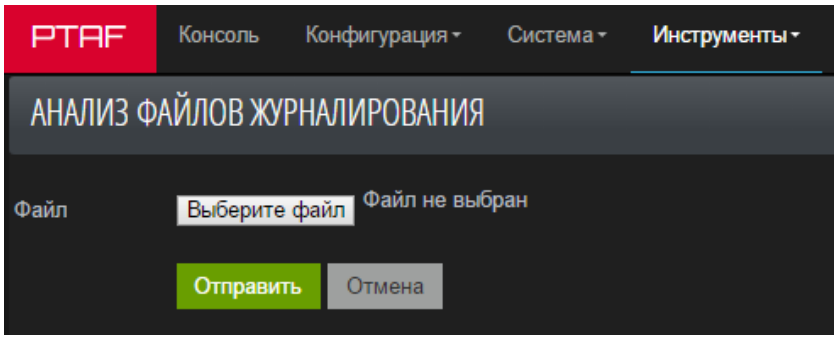


Рис. 215 – Загрузка файла

3. Определить формат файла, нажав кнопку *Определить* или кнопку *С выбранным* и выбрав пункт выпадающего меню *Определить формат*. В системе предусмотрены следующие форматы:

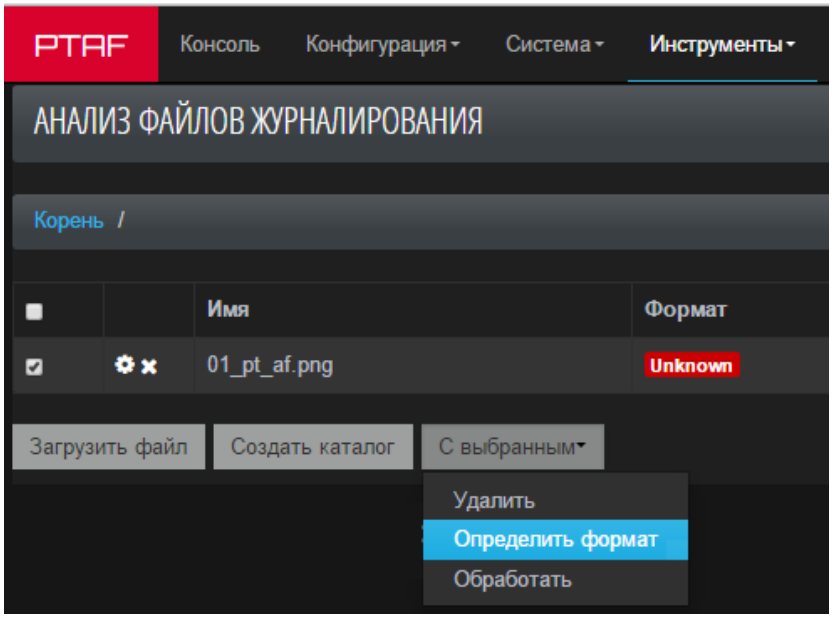


Рис. 216 – Определение формата

- Pcap file – файл дампа трафика в формате, используемом библиотекой libpcap;
- NCSA Common Log Format – файл журналирования в формате Common Log Format, который используется большинством веб-серверов: Apache, nginx, lighthttpd;
- W3C Extended Log Format – файл журналирования в формате, разработанном консорциумом W3C (http://www.w3.org/TR/WD-logfile.html), используемый веб-сервером IIS.

4. Отправить файл на обработку, выбрав пункт *Обработать* выпадающего меню *С выбранным*;

5. Просмотреть отчет, перейдя по ссылке, находящейся в поле *Индентификатор задания*.

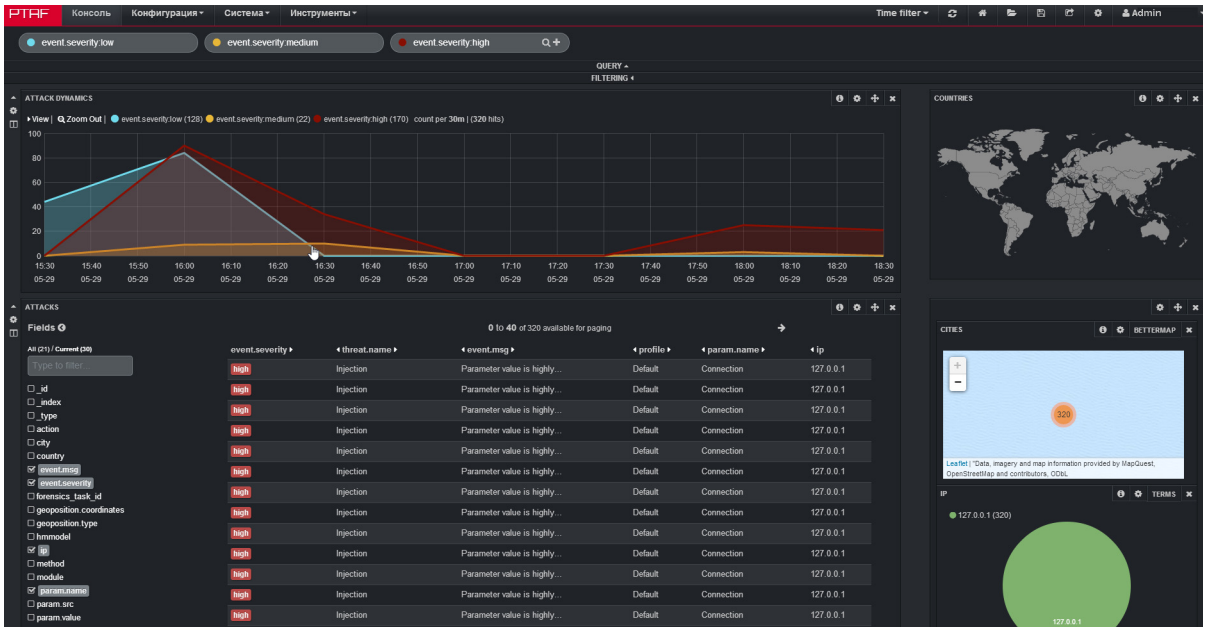


Рис. 217 – Просмотр отчета

8.2. Настройка интерфейса

8.2.1. Статистика по полям elasticsearch

- Перейдите в расширенный режим интерфейса;
- В столбце *Поля* рядом с атаками нажмите на имя интересующего поля (например, param.name)
- Нажмите кнопку *Ключевые слова* и выберите тип отображения:
  - Диаграмма со столбцами;
  - Круговая диаграмма;
  - Таблица.

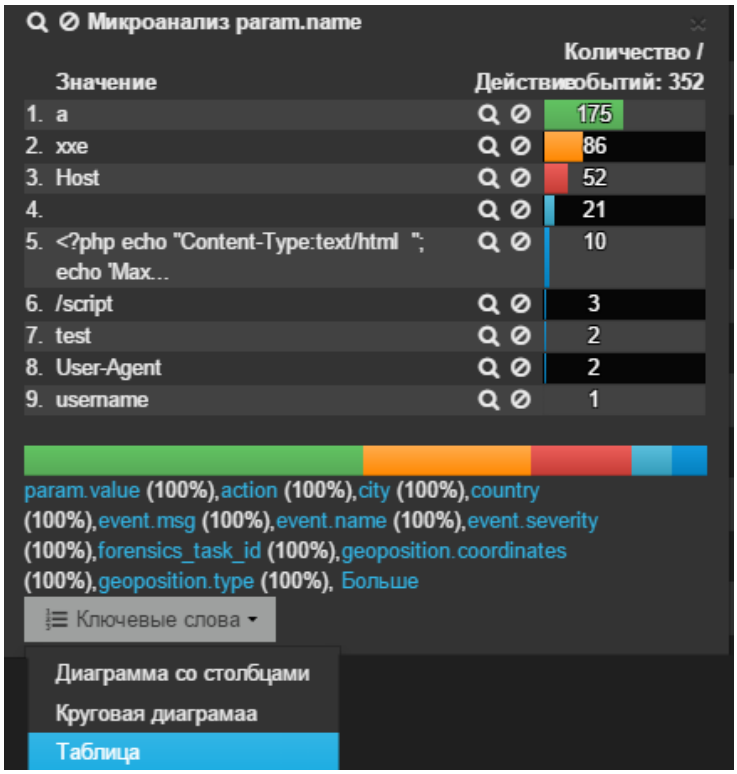


Рис. 218 – Выбор типа отображения

- Проанализируйте полученную статистику.

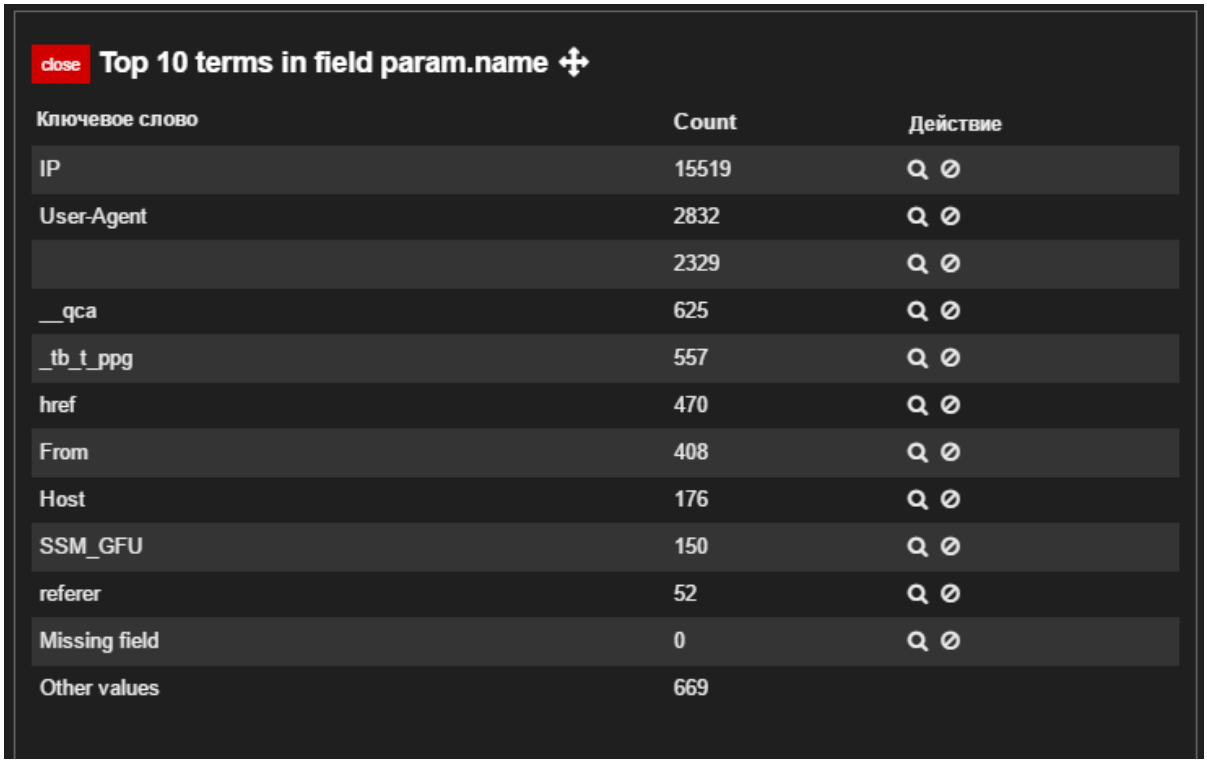


Рис. 219 – Отображение списка ключевых слов

8.2.2. Настройка графиков

- Перейдите в расширенный режим интерфейса;
- Перейдите в настройки графика, нажав кнопку *Настроить*;

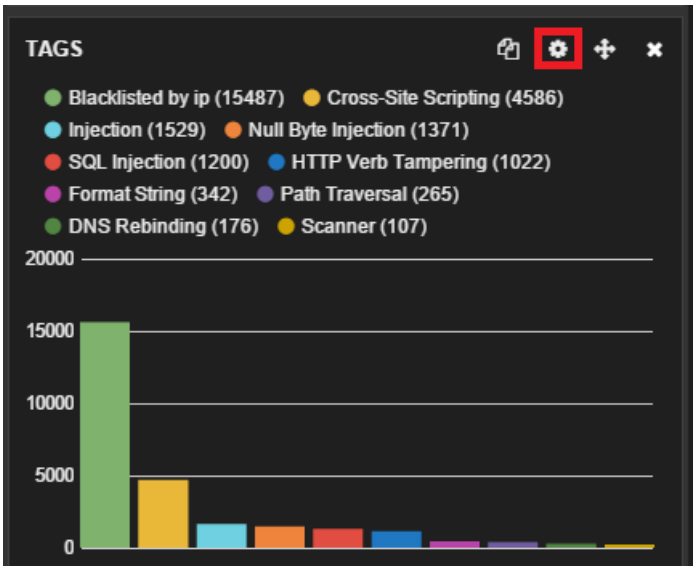


Рис. 220 – Переход к настройкам графика

- В открывшемся окне *Теги: Настройки* перейдите на вкладку *Панель* и измените необходимые параметры. Например, количество выводимых параметров и тип диаграммы;

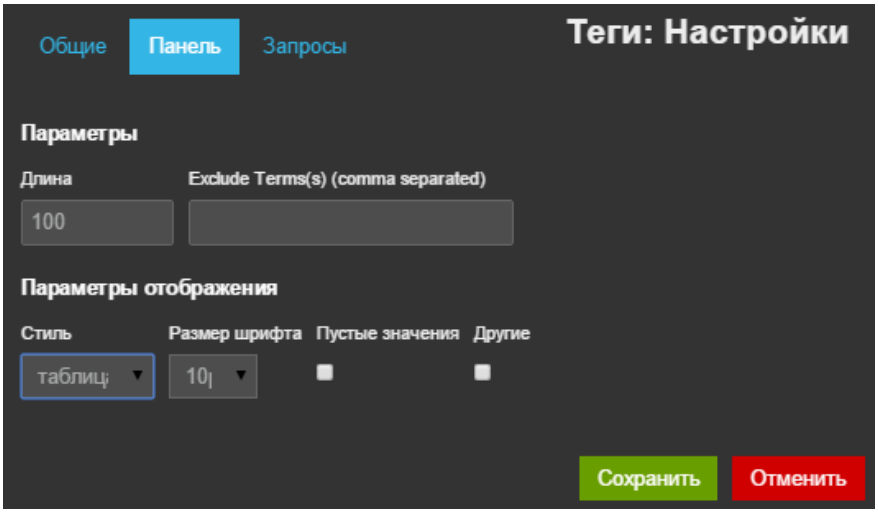


Рис. 221 – Изменение настроек графика

- Посмотрите результат.

Рис. 222 – Просмотр результата

9. Восстановление заводских настроек

Скрипт при установке waf создает резервные копии:

- MongoDB;
- Правил iptables;
- Конфигурационных файлов:

```
/etc/mongod.conf
/etc/ntp.conf
/etc/diamond/diamond.conf
/etc/elasticsearch/elasticsearch.yml
/etc/ferm/*
/etc/monit/monitrc
/etc/network/interfaces
/etc/nginx/nginx.conf
/etc/uwsgi/apps-enabled/graphite.ini
/etc/uwsgi/apps-enabled/ui.xml
/opt/waf/conf/*
```

При вызове `sudo waf_reset_tool` происходит:

- Восстановление MongoDB;
- Восстановление правил iptables;
- Очистка журнала событий;
- Восстановление конфигурационных файлов из резервной копии.

Внимание! После окончания процесса восстановления PT AF необходимо сконфигурировать заново, т.е. задать параметры сетевых интерфейсов, настроить Алиасы, Шлюзы и т.д. Подробное описание базовой настройки и конфигурации представлено в главах «Начальная настройка» и «Примеры конфигурации».

10. Поиск и устранение неисправностей

Рассмотрим несколько примеров решения типовых проблем в PTAF. В процессе обработки событий данные проходят несколько этапов:

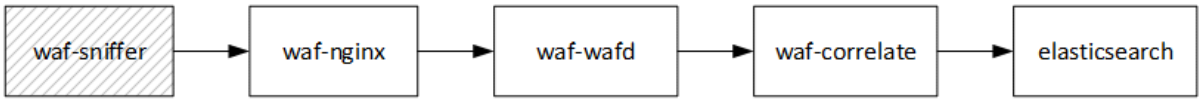


Рис. 223 – Этапы процесса обработки событий

- waf-sniffer снимает с интерфейса события, и отправляет их по внутреннему сокету в waf-nginx (только для режима работы snifer);
- waf-nginx передает события для анализа ядру waf-wafd;
- waf-wafd анализирует данные об атаках и передает их в waf-correlate;
- waf-correlate при необходимости объединяет атаки в коррелированные события и отправляет их в БД elasticsearch.

Если работа описанной цепочки нарушается, атаки не попадают в базу данных. Для диагностики придется пройти каждый этап обработки данных. Для проверки обнаружения атак из командной строки PTAF можно использовать следующие запросы (подставьте IP-адрес и порт сетевого интерфейса, который слушает PT AF):

```
curl "http://10.0.210.59/?testattack=<script>alert(1)</script>"
curl "http://10.0.210.59/?testattack='+or+1=1+--"
```

10.1. Диагностика waf-sniffer

- Проверить корректность настройки на вкладке *Конфигурация -> Сеть -> Сниффер*:

  - указан верный порт для отправки (тот же, что и в *Конфигурация -> Политики безопасности -> Профили*);

  - указаны верные IP-адреса приложений или * (для любых приложений);

  - указан верный список портов приложений;

- Убедиться, что процесс tapered запущен:

```
pt@ptaf:~$ ps aux | grep tapered
root 4 0.0 0.1 398320  5284 ? Ssl Sep13 0:00 /opt/waf/bin/tapered
```

  Если не запущен:

  - проверить журнал /var/log/monit.log на наличие ошибок запуска waf-sniffer;

  - попробовать запустить вручную /opt/waf/bin/tapered и проанализировать ошибки в журнале /var/log/waf/tapered.INFO

- Проверить, разбираются ли события с интерфейса:

```
pt@ptaf:~$ /opt/waf/bin/tapered-client
```

```
/opt/waf/bin/tapered-client
('IPC_CHECKIN_RESPONSE_MSG', {'code': 1})
('IPC_CONN_OPEN_IPV4_MSG', 1, ('10.0.68.41', 56693), ('10.0.210.59',
80), 0)
('IPC_CONN_CLIENT_SENDS_MSG', 1, 'GET / HTTP/1.1\r\nHost:
10.0.210.59\r\nConnection: keep-alive\r\nAccep...')
('IPC_CONN_SERVER_SENDS_MSG', 1, 'HTTP/1.1 502 Bad Gateway\r\nDate:
Mon, 14 Sep 2015 10:49:42 GMT\r\nC...')
('IPC_CONN_SERVER_SENDS_MSG', 1, '- a padding to disable MSIE and
Chrome friendly error page -->\r\n...')
```

Если событий нет:

- Проверить наличие трафика на интерфейсе:

```
pt@ptaf:~$ sudo ngrep -d eth2 port 80
interface: eth2 (10.0.208.0/255.255.240.0)
filter: (ip or ip6) and ( port 80 )
#
T 10.0.68.41:56411 -> 10.0.210.59:80 [AP]
  GET / HTTP/1.1..Host: 10.0.210.59..Connection: keep-alive..Cache-
Control: max-age=0..Accept: text/html,application/xhtml+xm
  l,application/xml;q=0.9,image/webp,*/*;q=0.8..Upgrade-Insecure-
Requests: 1..User-Agent: Mozilla/5.0 (Windows NT 6.3; Win64;
   x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/45.0.2454.85
Safari/537.36..DNT: 1..Accept-Encoding: gzip, deflate, sdc
  h..Accept-Language: en-US,en;q=0.8,ru;q=0.6..
```

Если отображается в виде:

```
pt@ptaf:~$ sudo ngrep -d eth2 port 80
interface: eth2 (10.0.208.0/255.255.240.0)
filter: (ip or ip6) and ( port 80 )
##############################################################################
################################################
```

- Проверить наличие в трафике запросов на приложение и ответов с него:

```
root@ptaf:/home/pt# tcpdump -ni eth2 port 80
tcpdump: verbose output suppressed, use -v or -vv for full protocol
decode
listening on eth2, link-type EN10MB (Ethernet), capture size 65535
bytes
13:56:49.336868 IP 10.0.68.41.57015 > 10.0.210.59.80: Flags [S],
seq 4147737156, win 8192, options [mss 1460,nop,wscale
2,nop,nop,sackOK], length 0
13:56:49.336969 IP 10.0.210.59.80 > 10.0.68.41.57015: Flags [S.],
seq 1810370147, ack 4147737157, win 14600, options [mss
1460,nop,nop,sackOK,nop,wscale 6], length 0
```

Если в трафике только запросы на приложение: завернуть весь трафик включая ответы с приложения и проверить снова.

- Проверить ошибки в файле `/var/log/waf/waf-sync.log`

К примеру, если интерфейс, с которого предполагается разбирать трафик отсутствует, или для него не назначен IP-адрес, отобразится следующая ошибка:

```
2015-09-14 14:27:30 ERROR    waf_sync.utils  Cannot get ip address on
interface eth0, error: 2
```

10.2. Диагностика waf-nginx

- Проверить, что запросы попадают в журнал /var/log/waf/access.log:

```
pt@ptaf:~$ sudo tail -0f /var/log/waf/access.log
10.0.68.41 - - [14/Sep/2015:14:14:53 +0300] "GET / HTTP/1.1" 200 286 "-
" "Mozilla/5.0 (Windows NT 6.3; Win64; x64) AppleWebKit/537.36 (KHTML,
like Gecko) Chrome/45.0.2454.85 Safari/537.36"
10.0.68.41 - - [14/Sep/2015:14:14:53 +0300] "GET /bWAPP/images/
evil_bee.png HTTP/1.1" 200 24952 "http://10.0.210.59/" "Mozilla/5.0
(Windows NT 6.3; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/45.0.2454.85 Safari/537.36"
```

Если событий нет:

- Служба waf-nginx запущена и слушает правильный порт:

```
pt@ptaf:~$ sudo netstat -tunlp | grep waf-nginx
tcp 0 0 10.0.210.59:80  0.0.0.0:* LISTEN 36665/waf-nginx
tcp 0 0 127.0.0.1:18083 0.0.0.0:* LISTEN 36665/waf-nginx
tcp 0 0 127.0.0.1:18084 0.0.0.0:* LISTEN 36665/waf-nginx
```

Если не запущена: запустить, и посмотреть на вывод ошибок в консоль и журнале /var/log/waf/error.log:

```
sudo service waf-nginx start
```

Если сервис не запускается: перезапустить waf-wafd и проверить, нет ли ошибок с ключом.

- Проверить доступность порта waf-nginx:

```
pt@ptaf:~$ curl 10.0.210.59:80
200
```

Если недоступен: перезапустить сервис waf-nginx и проверить на наличие нескольких master-процессов (следующий шаг):

```
sudo service waf-nginx restart
```

- В процессах waf-nginx только один мастер-процесс:

```
pt@ptaf:~$ ps aux | grep waf-nginx
root     36665  0.0  0.0 132356  3224 ?        Ss   Sep11   0:00
waf-nginx: master process /opt/waf/bin/waf-nginx
112      62534  0.0  1.0 695328 41136 ?        S    14:47   0:00
waf-nginx: worker process
112      62534  0.0  1.0 695328 41136 ?        S    14:47   0:00
waf-nginx: worker process
```

```
112      62534  0.0  1.0 695328 41136 ?        S    14:47   0:00
waf-nginx: worker process
112      62534  0.0  1.0 695328 41136 ?        S    14:47   0:00
waf-nginx: worker process
```

Если в списке присутствуют несколько master-процессов: выполнить последовательность команд:

```
sudo service waf-wafd stop
sudo service waf-nginx stop
rm -fr /dev/shm/sem*
rm -fr /tmp/*.cache
sudo service waf-wafd start
sudo service waf-nginx start
```

- Проверить правильность настроек на вкладке *Конфигурация -> Политики безопасности -> Профили*:

  - на вкладке *Прокси* указан верный порт сниффера (тот же, что и на вкладке *Конфигурация -> Сеть -> Сниффер, опция Сервер -> Порт*);

  - Включена опция *Включен (Sniffer)*;

10.3. Диагностика waf-wafd

- Проверить, что сервис запущен:

```
sudo service waf-wafd status
```

Если не запущен: запустить

```
sudo service waf-wfad start
```

Если при запуске возникает ошибка ключа:

- Для локального ключа – проверить, что он проброшен в VM или подключен напрямую к серверу;

- Для сетевого ключа – проверить, что сервер лицензий доступен по 6001,6002 (TCP/UDP) портам;

- Выполнить проверку сервисной программой (см. главу «Диагностика USB-ключа Guardant»)

10.4. Диагностика waf-correlate

- Проверить, что служба запущена:

```
sudo service waf-correlate status
```

Если не запущена: запустить, `service waf-correlate start`

Если не запускается: проверить журнал /var/log/waf/waf-correlate.log

10.5. Диагностика elasticsearch

- Проверить, что служба запущена:

```
sudo service elasticsearch status
```

Если не запущена: запустить, `service elasticsearch start`

Если не запускается: проверить журнал /var/log/elasticsearch/elasticsearch.log

- Проверить, что на диске достаточно свободного места:

```
pt@ptaf:~$ df -h
Filesystem              Size  Used Avail Use% Mounted on
rootfs                   47G  7.9G   37G  18% /
```

10.6. Недоступен UI (502 Bad gateway)

- Проверить, что служба запущена:

```
sudo service uwsgi status ui
```

Если не запущена: запустить, `service uwsgi start`

10.7. Недоступен UI (ошибка сети)

- Проверить, что служба запущена:

```
sudo service nginx status
```

Если не запущена: запустить, `service nginx start`

Если не запускается:

- убедиться, что порт UI не используется другими процессами:

```
pt@ptaf:~$ sudo netstat -tunlp | grep nginx
tcp 0 0 0.0.0.0:8080   0.0.0.0:* LISTEN      27596/nginx
tcp 0 0 127.0.1.1:8082 0.0.0.0:* LISTEN      27596/nginx
```

- проверить на наличие ошибок журнал `/var/log/nginx/error.log`
- проверить правильность настроек порта и SSL в конфигурационном файле `/etc/nginx/sites-enabled/ui.conf`

10.8. Недоступно защищаемое приложение (в режиме reverse-proxy, ошибка 502 Bad gateway)

- Проверить доступность приложения из командной строки:

```
pt@ptaf:~$ curl http://10.0.221.89
<!DOCTYPE html><html><body>…
```

Если недоступно:

- проверить, что приложение доступно с другого источника;
- проверить правильность сетевых настроек:

```
ifconfig
netstat -r
```

- проверить доступность хоста с приложением: ping 10.0.221.89
- проверить настройки firewall: iptables-save

10.9. Не отображаются новые атаки

• Отключить стандартный фильтр и проверить, не появились ли новые события на вкладке *Консоль*:

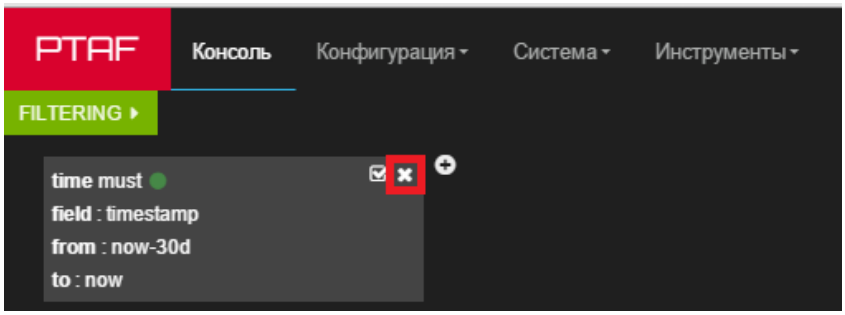


Рис. 224 –

• Сравнить дату/время/timezone на PTAF с текущими:

```
pt@ptaf:~$ date
Mon Sep 14 17:34:05 MSK 2015
```

10.10. Пропал график с динамикой атак (после отключения фильтра)

• Перейти в *Расширенный* режим:

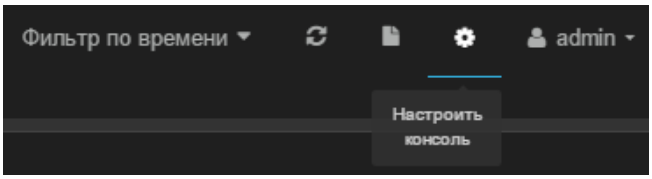


Рис. 225 – Переход в расширенный режим

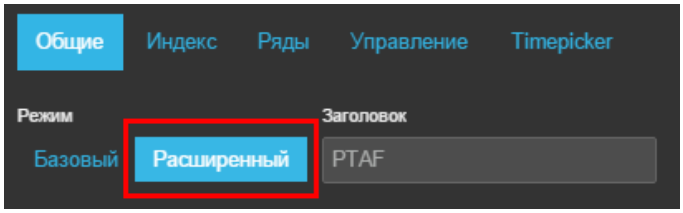


Рис. 226 – Выбор расширенного режима

• Открыть настройки графика:

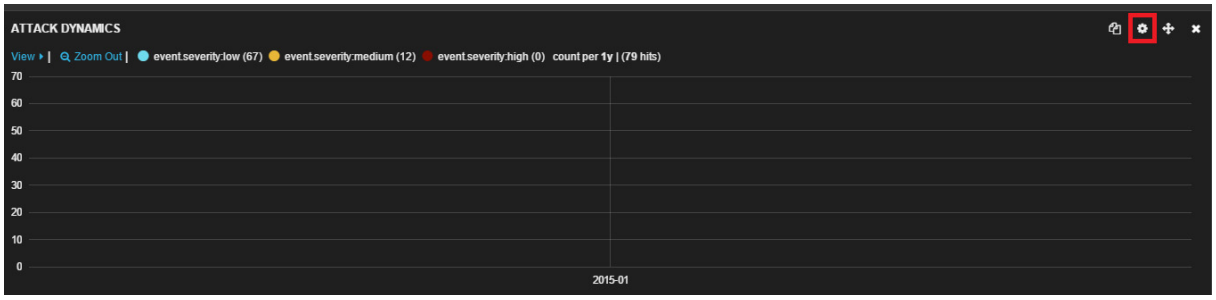


Рис. 227 – Настройки графика

- Перейти на вкладку *Панель*, отключить *Автоматический интервал* и вручную задать период в поле *Интервал* (например, 1h).

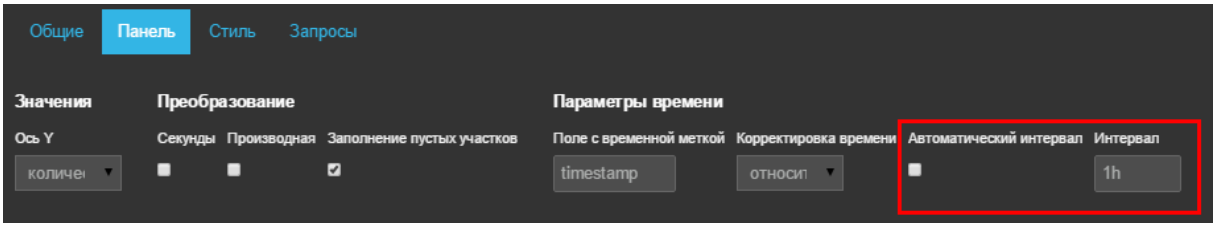


Рис. 228 – Вкладка Панель

- Сохранить настройки.

10.11. Диагностика USB-ключа Guardant

Симптомы:

- Ошибка в UI на входе в систему:

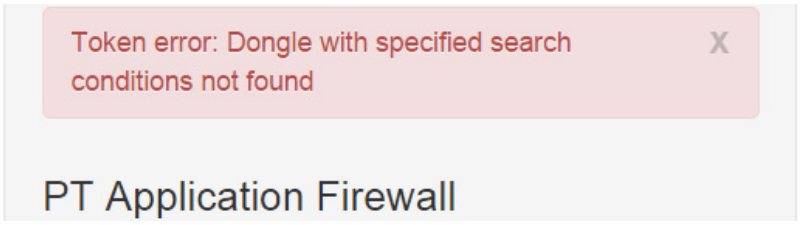


Рис. 229 – Ошибка в UI

- Ошибки при запуске служб:

```
pt@ptaf:/home/pt# sudo service waf-wafd restart
[....] Restarting waf-wafd: waf-wafdERROR: Bad configuration: please,
activate token (No such file or directory, errno=2) failed!
```

Диагностика:

- Проверить, что ключ подключен (корректно проброшен):

```
root@ptaf:/home/pt# lsusb
Bus 001 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
Bus 002 Device 001: ID 1d6b:0001 Linux Foundation 1.1 root hub
```

```
Bus 002 Device 002: ID 0e0f:0003 VMware, Inc. Virtual Mouse
Bus 002 Device 003: ID 0e0f:0002 VMware, Inc. Virtual USB Hub
Bus 002 Device 004: ID 0a89:0009
```

- Выполнить проверку сервисной программой:
  - Для локального ключа: ./ServiceProgram –d
  - Для сетевого ключа: ./ServiceProgram -d –n

    Примечание: в случае сетевого ключа убедиться, что файл настроек gnclient.ini расположен в одной папке с сервисной программой.

    Результатом успешного выполнения служит отсутствие ошибок и вывод «System Data was checked!!!»

    Например, в случае отсутствия файла лицензии в выводе сервисной программы появится строка:

    [system_msg] Trying to read the "/opt/waf/data/license" file : ERROR(File "/opt/waf/data/license" not found)

    А ошибка «[system_msg] Error! The hardware algorithm with specified number has not been found in the dongle.» говорит о том, что ключ залочился физически и потребуется перепрошивка.
- В случае ошибки «Data verification failed» (только локальный ключ):
  - Выполнить:

    ```
    ./ServiceProgram -d -c
    ./ServiceProgram -d
    ```
  - Убедиться, что нет ошибок
  - Перезапустить waf-wafd:

    ```
    sudo service waf-wafd restart
    ```

10.12. Просмотр списка блокируемых IP-адресов

Если по какой-то из причин невозможно просмотреть список заблокированных IP-адресов через интерфейс PT AF, то воспользуйтесь следующим методом.

Блокировка осуществляется через модуль ipset межсетевого экрана. Соответствующее правило в netfilter выглядит так:

```
iptables -A INPUT -m set --match-set wafdset src -j REJECT --reject-with icmp-port-unreachable
```

Список блокируемых IP-адресов можно увидеть, просмотрев набор wafdset при помощи команды ipset (может не поставляться в составе PT AF, т.к. PT AF управляет наборами самостоятельно через соответствующие API).

```
# ipset -L wafdset
Name: wafdset
Type: hash:ip
Header: family inet hashsize 1024 maxelem 4000001 timeout 600
Size in memory: 16568
References: 1
Members:
192.0.2.12 timeout 280
```

```
192.0.2.123 timeout 3572
```

Здесь видно, какие адреса заблокированы и сколько времени осталось до конца блокировки.

Positive Technologies – лидер европейского рынка систем анализа защищенности и соответствия стандартам. Компания входит в число наиболее динамично развивающихся участников российской IT-отрасли, демонстрируя ежегодный рост более 50%. Офисы и представительства Positive Technologies расположены в Москве, Лондоне, Риме, Сеуле и Тунисе.

Разработанные экспертами компании программные продукты заслужили международное признание в сфере практической информационной безопасности.

Продукты

Система контроля защищенности и соответствия стандартам MaxPatrol помогает обеспечивать безопасность корпоративных информационных систем и формировать комплексное представление о реальном уровне защищенности IT-инфраструктуры организации. Система позволяет контролировать выполнение требований государственных, отраслевых и международных стандартов, таких как Федеральный закон № 152-ФЗ «О персональных данных», СТО БР ИББС, ISO 27001/27002, SOX 404, PCI DSS. В MaxPatrol объединены активные механизмы оценки защищенности, включая функции системных проверок, тестирования на проникновение, контроля соответствия стандартам – в сочетании с поддержкой анализа различных операционных систем, СУБД и веб-приложений.

Система анализа защищенности XSpider более 10 лет является признанным лидером среди средств сетевого аудита ИБ. На сегодняшний день это один из лучших интеллектуальных сканеров безопасности в мире. Более 1000 международных компаний успешно используют XSpider для анализа и контроля защищенности корпоративных ресурсов.

Услуги

Компания Positive Technologies специализируется на проведении комплексного аудита информационной безопасности, на оценке защищенности прикладных систем и веб-приложений, тестировании на проникновение и внедрении процессов мониторинга информационной безопасности. Статус PCI DSS Approved Scanning Vendor позволяет проводить работы по проверке соответствия данному стандарту.

Клиенты

В числе заказчиков Positive Technologies – более 1000 государственных учреждений, финансовых организаций, телекоммуникационных и розничных компаний, промышленных предприятий России, стран СНГ и Балтии, а также Великобритании, Германии, Голландии, Израиля, Ирана, Китая, Мексики, США, Таиланда, Турции, Эквадора, ЮАР и Японии.

Вклад в индустрию

Принимая активное участие в развитии IT-отрасли, Positive Technologies выступает организатором международного форума по информационной безопасности Positive Hack Days и развивает SecurityLab.ru – самый популярный ИБ-портал на русском языке.

Более подробную информацию можно получить на сайте www.ptsecurity.ru