Платформа Радар

Руководство по подключению источников

Версия 3.5.4

ООО Пангео Радар

Оглавление

1. Общее описание процесса подключения источников

Руководство по подключению источников содержит рекомендации и инструкции для настройки Платформы Радар для приема событий в пассивном и активном режимах, настройки источников, настройки лог-коллектора, а также обработки событий, включая фильтрацию и обогащение.

> **ВНИМАНИЕ! Перед внесением изменений в конфигурационные файлы не забудьте сделать их резервную копию.**

1.1. Пассивный сбор

В Платформе присутствует возможность приема событий от источников в пассивном режиме. Для этого необходимо в веб-интерфейсе Платформы настроить прием событий: включить поддерживаемые источники или создать и настроить новые источники, которые смогут самостоятельно отправлять данные. Подробное описание включения и создание источников дано в разделе «Работа с пассивными источниками событий»

1.2. Активный сбор

Для организации активного сбора необходимо использовать лог-коллектор. Он предназначен для организации сбора событий от активов, не имеющих возможности самостоятельной отправки данных в сторонние системы. Подробное описание настройки лог-коллектора дано в разделе «Руководство по настройке лог-коллектора. Активные источники событий».

1.3. Процесс подключения типового источника

Подключение типового источника осуществляется в три этапа:

1. Настройка Платформы на прием событий путем включения необходимого источника в веб-интерфейсе Платформы. После включения источника Платформа готова к приему событий в пассивном режиме. Подробнее о включении типовых источников в разделе «Работа с пассивными источниками событий».
2. Настройка лог-коллектора, если необходимо организовать активный сбор или реализовать цепочку по пересылке событий между двумя и более лог-коллекторами. Подробная настройка лог-коллектора описана в «Руководство по настройке лог-коллектора. Активные источники событий»
3. Настройка источника. Настройка производится согласно рекомендациям производителя или в соответствии с разделом с описанием поддерживаемых источников их настройки.

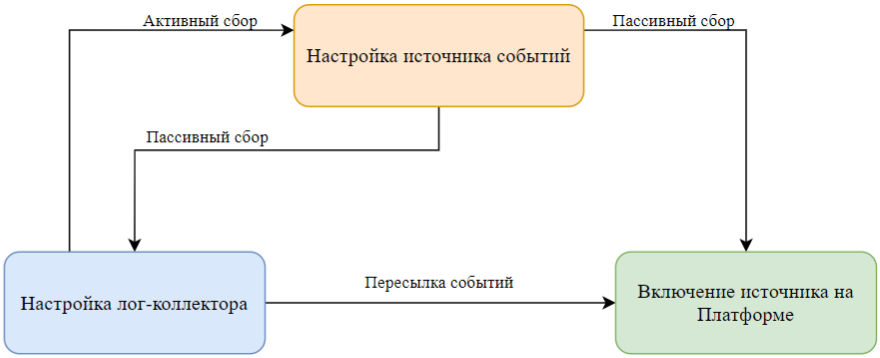

Рисунок 1 - Добавление типового источника

1.4. Процесс подключения нетипового источника

Подключение нетипового источника осуществляется в пять этапов:

1. Настройка Платформы на прием событий путем создания нового пассивного источника событий в веб-интерфейсе Платформы. После включения источника Платформа готова к приему событий в пассивном режиме. Подробнее о создании новых источников в разделе «Работа с пассивными источниками событий»
2. Настройка Лог-коллектора, если необходимо организовать активный сбор или реализовать цепочку по пересылке событий между двумя и более Лог-коллекторами. Подробная настройка Лог-коллектора описана в "Руководство по настройке лог-коллектора. Активные источники событий"
3. Настройка источника. Настройка производится согласно рекомендациям производителя или в соответствии с разделом с описанием поддерживаемых источников и их настройки.
4. Создание правил разбора для событий с нового источника. Подробнее в разделе про форматы правил разбора
5. Создание правил нормализации для событий с нового источника. Подробнее в разделе «Разработка правил разбора и нормализации событий»

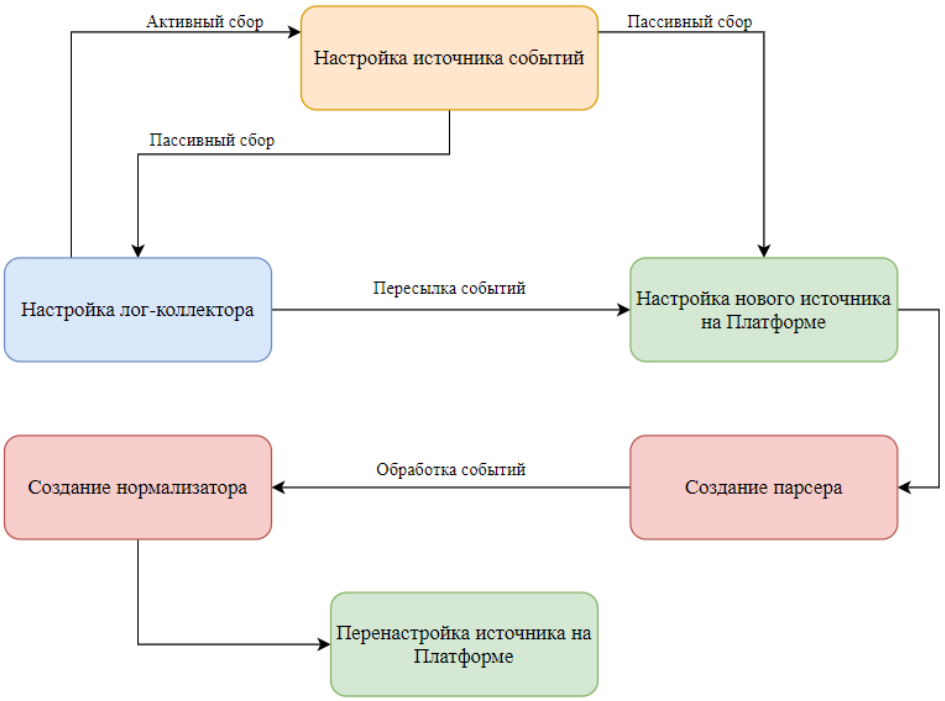

Рисунок 2 - Добавление нетипового источника

1.5. Проверка получения данных от источников

Выполнить проверку поступающих данных можно в веб-интерфейсе Платформы в разделе «**Инциденты**» — «**Просмотр событий**», выставив необходимые временные фильтры.

2. Работа с пассивными источниками событий

Все действия по управлению пассивными источниками в веб-интерфейсе Платформы выполняются в разделе «Администрирование» -> «Источники» -> «Управление источниками» (см. рисунок 3).

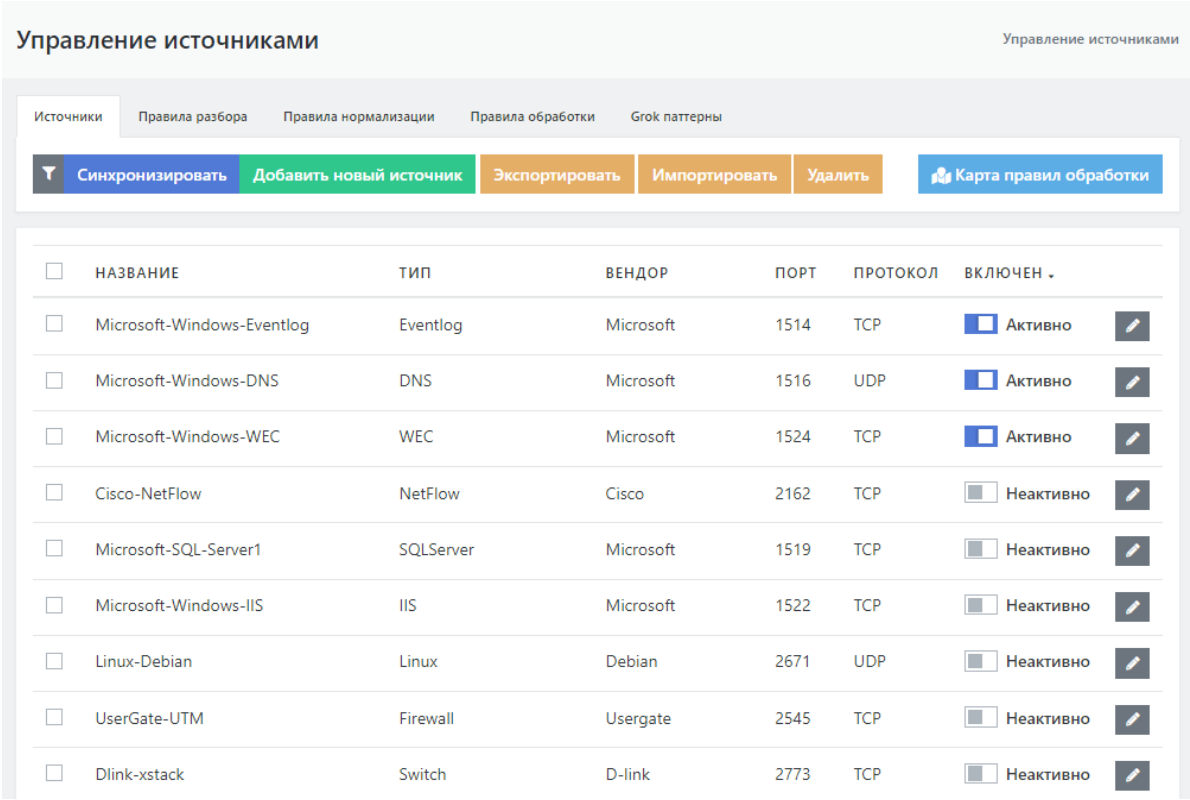

Рисунок 3 - Управление источниками

> ВНИМАНИЕ! За сбор и отправку событий отвечает служба `rsyslog`, расположенная на мастер-ноде, в случае установки на один сервер, или на ноде балансировщика, в случае распределенной установки. Убедитесь, что эта служба включена, в противном случае события в Платформу поступать не будут.

2.1. Включение/выключение пассивных источников и их синхронизация {#onoff_source}

Для включения/выключения источников необходимо выполнить следующие действия:

1. Выбрать источник и проверить его текущий статус работы в столбце **ВКЛЮЧЕН**: **синий фон** кнопки-переключателя и надпись **Активно** показывают, что источник включен, **белый фон** и надпись **Не активно**, что выключен (см. рисунок 4).

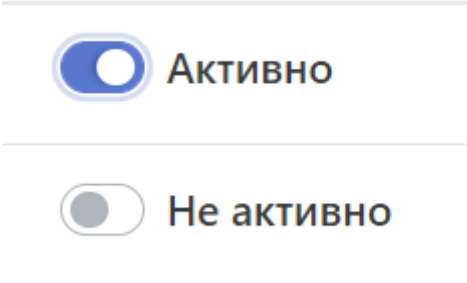

Рисунок 4 - Статус работы источника

2. Включить/выключить все необходимые источники. Включение и выключение источника осуществляется нажатием на кнопку-переключатель.
3. Выполнить синхронизацию источников, чтобы внесенные изменения вступили в силу, нажав кнопку **Синхронизировать** (см. рисунок 5).

> **Важно!** Необходимо синхронизировать источники после каждого изменения

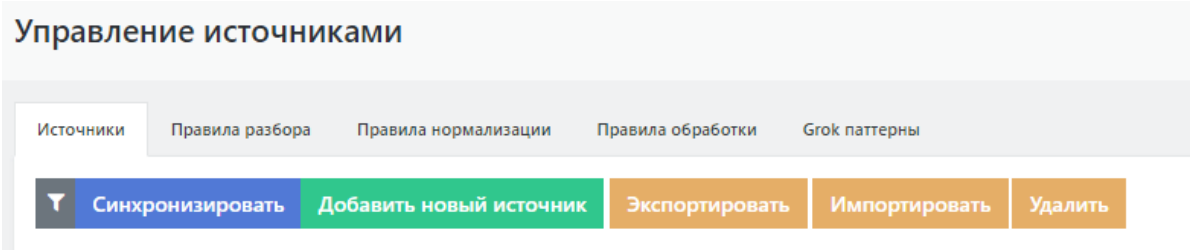

Рисунок 5 - Кнопки управления источниками

После выполнения вышеперечисленных действий Платформа готова к приему событий от включенных источников в пассивном режиме.

2.2. Экспорт, импорт и удаление источника

Кнопки управления источниками (см. рисунок 5) позволяют:

- Экспортировать выбранные источники в файл архива формата ZIP.
- Импортировать источники из файла архива формата ZIP.
- Удалять выбранные источники.

> **Важно!** Необходимо синхронизировать источники после каждого изменения

2.3. Заведение нового пассивного источника

Для заведения нового пассивного источника:

1. Нажать кнопку **Добавить новый источник** в верхней части страницы «Управление источниками» (см. рисунки 1,3).
2. Заполнить форму (см. рисунок 6). Расшифровка полей формы дана в следующем разделе *«Описание полей формы»*.
3. При необходимости проверить работу выбранных парсера и нормализатора в правой части формы, подставив сырое событие от источника и запустив проверку. Результаты проверки появятся во всплывающем окне.
4. Сохранить данные, нажав кнопку **Сохранить** в нижней левой части формы. Для выхода из формы без сохранения данных нажмите кнопку **Отменить**.
5. Включить новый источник.
6. Синхронизировать источники.

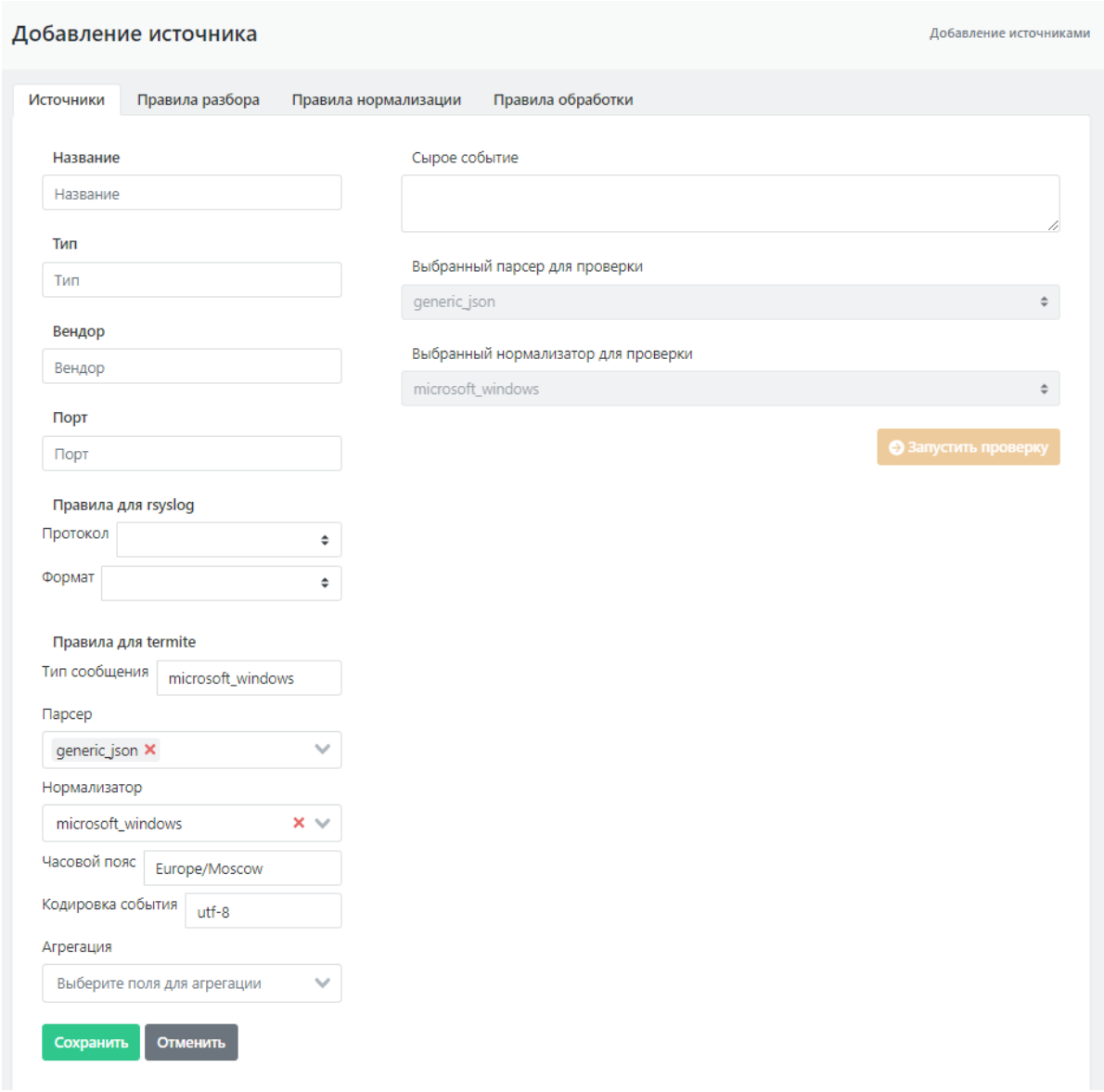

Рисунок 6 - Форма добавления нового типа источника

2.4. Описание полей формы создания/ редактирования пассивного источника {#fields}

Название — наименование типа источника (пример: «Linux Debian»)
Тип — тип источника (пример: «Linux»)
Вендор — производитель системы, которая выступает в качестве источника (пример: «Debian»)
Порт — необходимо указать один из свободных портов, который будет использоваться для отправки события с нового источника (+- диапазон 6000-8000)

Область **Правила для rsyslog**:

- Поле **Протокол** — протокол, по которому будут приниматься события. Возможные форматы:
  - **TCP**,
  - **PTCP** (Plain TCP),
  - **UDP**.
- Поле **Формат** — правила приема и обработки события. Возможные форматы:
  - **RAW** — не изменять входящее событие
  - **RAW-JSON** — обогатить сообщение дополнительной технической информацией и упаковать в пакет json

- **JSON-JSON** — обогатить существующую структуру json дополнительными полями с технической информацией

Область **Правила для termite**:

- **Тип сообщения** — указывается тип события из правила нормализации.
- **Парсер** — указывается правило разбора данного типа событий.
- **Нормализатор** — указывается правило нормализации.
- **Часовой пояс** — указывается необходимая временная зона (пример: «Europe/Moscow»).
- **Кодировка событий** — указывается необходимая кодировка (пример: «utf-8»).
- **Агрегация** — позволяет выполнить агрегацию однотипных событий. Необходимо указать поля, которые могут меняться. Расшифровка полей для агрегации дана в разделе «Описание полей нормализации».

2.5. Изменение параметров пассивного источника

Для изменения данных об источнике необходимо выполнить следующие действия:

1. Нажать кнопку редактирования  в строке выбранного источника.
2. Внести необходимые изменения в форму редактирования источника (см. рисунок 7).
3. Сохранить данные, нажав кнопку **Сохранить** в нижней левой части формы. Для выхода из формы без сохранения данных нажать кнопку **Отменить**.
4. Синхронизировать источники нажав кнопку **Синхронизировать** на вкладке "Источники".

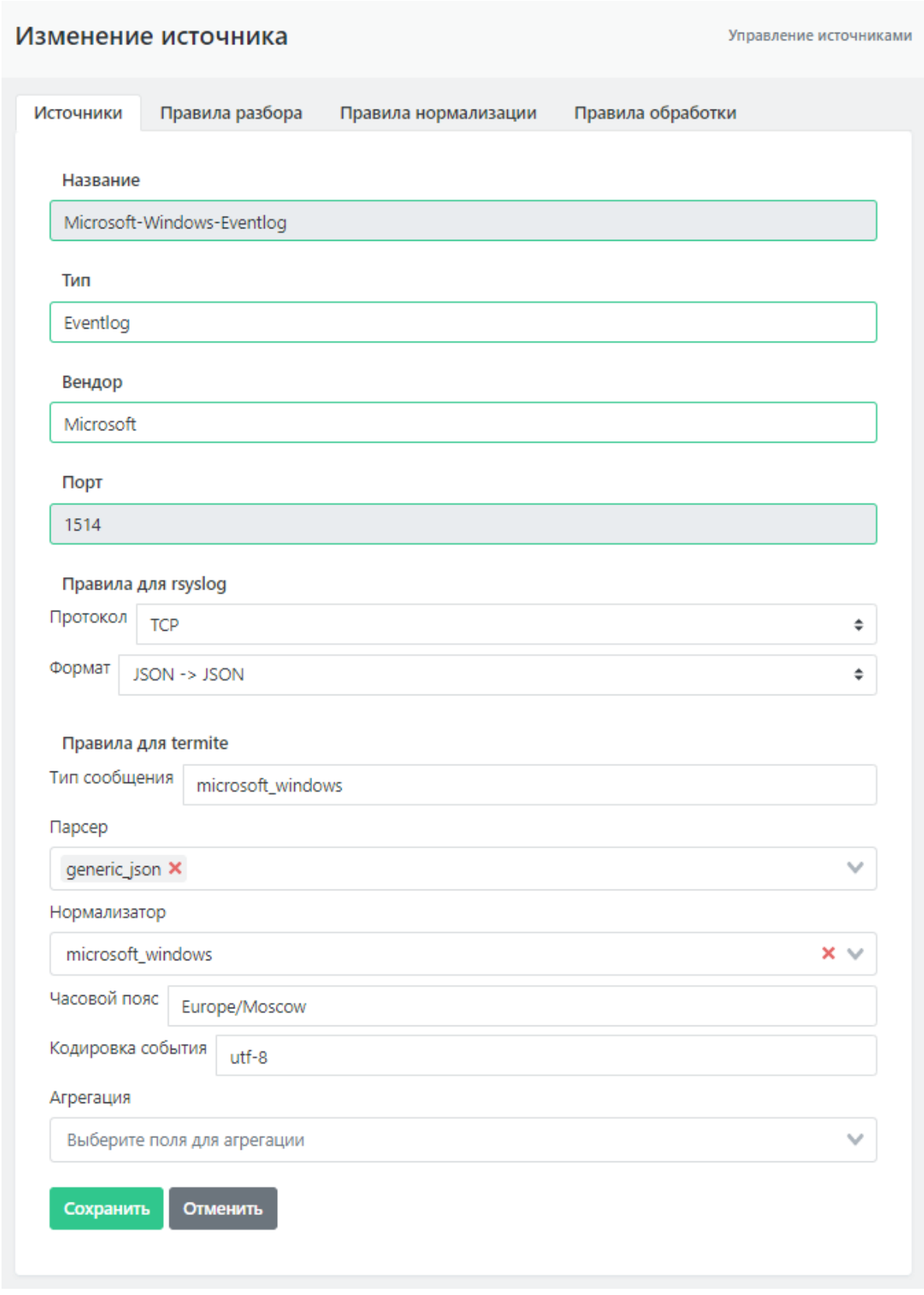

Рисунок 7 - Форма редактирования типа источника

3. Список поддерживаемых источников

Данный раздел содержит перечень систем, которые могут быть подключены к **Платформе Радар** в качестве источников событий

3.0.1. Операционные системы

Наименование	Версия	Примечание
Astra Linux		
CentOS Linux	7, 8	
Debian Linux	8, 9, 10	
Fedora Linux	30, 31	
IBM AIX	7.1, 7.2	
Microsoft Windows	XP, 7+	
Microsoft Windows Server	2003, 2008+	
Microsoft Windows Event Forwarding	7+, 2008+	
Oracle Solaris	10, 11	
Red Hat Enterprise Linux (RHEL)	6, 7, 8	
SUSE Linux Enterprise	11.3, 12	
Ubuntu Linux	16.04+	

3.0.2. Решения Endpoint Security

Наименование	Версия	Примечание
FireEye HX		
Kaspersky Security Center	10, 11	
Kaspersky Web Traffic Security		
McAfee ePolicy Orchestrator	5.9, 5.10	
PaloAlto Traps		
Symantec Endpoint Protection	14	

3.0.3. Решения Network Security

Наименование	Версия	Примечание
Barracuda Firewall		
Bluecoat Proxysg	6, 7	
Checkpoint NGFW	77, 80	log export(syslog)
Cisco ASA		

Наименование	Версия	Примечание
Cisco Firepower		
Cisco snort		
Citrix ADC (Netscaler)		
Fortinet Fortigate	5, 6	
McAfee Web Gateway		
nGate Firewall		
OPSEC LEA		
PaloAlto NGFW	7, 8	
pfSense Firewall		
Radware DefencePro		
SecurityCode Continent	3.7, 3.9	
SecurityCode Continent IDS		
Suricata IDS		
Trend Micro TippingPoint		
Usergate UTM Firewall	6	
СКДПУ НТ		

3.0.4. Решения Application Security

Наименование	Версия	Примечание
F5 BIG-IP	15	

3.0.5. Сетевые устройства

Наименование	Версия	Примечание
Cisco IOS		
Cisco Netflow		
D-link xStack		
Huawei Switch		
Infoblox Trinzic		

3.0.6. Системы управления базами данных

Наименование	Версия	Примечание
Microsoft SQL Server	2014+	
PostgreSQL	9+	
Oracle Database		
Oracle MySQL		
Oracle NetListener		

3.0.7. Системы защиты электронной почты

Наименование	Версия	Примечание
FortiSandbox		
IBM Postfix		
Kaspersky Secure Mail Gateway		
Microsoft Exchange Server	2013/2016/2019	
SEPPmail Secure Email	9	

3.0.8. Системы контроля привилегированного доступа

Наименование	Версия	Примечание
CyberArk PAM		
RSA SecurID		
SearchInform DLP		
SmartLine DeviceLock DLP	8x	
Solar Dozor	7.9	
Staffcop Enterprise		

3.0.9. Инфраструктурные системы

Наименование	Версия	Примечание
Dell IDRAC		
HAProxy		
ISC Bind DNS	9	
Linux NFS Server		

Наименование	Версия	Примечание
Microsoft DNS	2008+	
Microsoft DHCP	2008+	
vGate		
VMware ESXi		
VMware vCenter		

3.0.10. Web-серверы

Наименование	Версия	Примечание
Apache HTTP server		
Apache Tomcat		
Lighttpd		
Microsoft IIS		
Nginx		

3.0.11. Proxy-серверы

Наименование	Версия	Примечание
Squid	3.5+	
Solar WebProxy	3.8.x	

3.0.12. Другое

Наименование	Версия	Примечание
Confident Dallaslock	8.0-K	
Linux Auditd		
Microsoft Sysmon		

4. Операционные системы

4.1. Microsoft Windows 7+/2008+ {#win}

4.1.1. Настройка источника

1. Создание учетной записи для сбора событий.
    - Если источник находится в домене, то на контроллере домена необходимо создать учетную запись и добавить ее в группу Event Log Readers.
    - Если источник не находится в домене, то необходимо создать локальную учетную запись с аналогичным набором прав.
      Процесс создания учетной записи приведен в разделе *Создание учетной записи Microsoft Windows*.
2. При использовании межсетевого экрана на узле, необходимо сделать правило для входящих соединений.
   Настройка расширенного аудита представлена в разделе *Настройка расширенных политик аудита Windows*.

4.1.2. Включение источника в Платформе Радар {#turnwin}

Для информации! Включение источника в **Платформе Радар** подробно представлено в разделе *Управление источниками в **Платформе Радар** — Включение/выключение источников и их синхронизация*.

1. Зайдите в веб-консоль **Платформы Радар**, перейти в раздел «Источники» — «Управление источниками».
2. Найдите в списке доступных источников «Microsoft-Windows-Eventlog» и включить его.
3. Нажмите на кнопку «Синхронизировать».

4.1.3. Настройка коллектора событий {#lcwin}

Для информации! Пример конфигурационного файла с настройкой данного источника представлен в разделе *Пример конфигурационного файла лог-коллектора*. Более подробная информация о настройках лог-коллектора представлена в разделе *Руководство по настройке лог-коллектора*.

1. В конфигурационный файл лог-коллектора (config.yaml) необходимо добавить input компонента Eventlog. Пример настройки по умолчанию можно найти в документации: *Руководство по настройке лог-коллектора — Компонент Eventlog*.
   Основные параметры, которые необходимо указать:

   ```
   channel: ['<название журнала, который нужно собирать>']
   ```

   Например:

   ```
   channel: ['Security', 'System']
   ```

   Заполнить вкладку remote, по следующему принципу:

```
enabled: true (включение удаленного сбора)
user: <"username в открытом или зашифрованном виде"> (имя пользователя с
правами на чтение журнала событий)
password: <"password в открытом или зашифрованном виде"> (пароль
пользователя)
domain: <"домен пользователя"> (если машина не в домене - ".")
remote_servers: [<"ip-адрес удаленного узла">] (адрес/список адресов
серверов для сбора событий)
auth_method: <"метод авторизации"> (выбрать один из доступных методов
авторизации: Negotiate, Kerberos, NTLM)
```

2. После настройки компонента сбора событий (input) - необходимо настроить компонент отправки событий (output). В качестве компонента отправки событий для данного источника предусмотрено использование отправки по протоколу TCP. Пример настройки по умолчанию можно найти в документации: *Руководство по настройке лог-коллектора, Компонент отправки событий по протоколу TCP.*
   Основные параметры, которые необходимо указать:

```
target_host: <"ip адрес или имя удаленного узла"> (адрес Платформы Радар)
port: <"порт"> (стандартный порт для данного источника 1514)
```

3. Далее необходимо включить компоненты сбора (collectors) и отправки (senders). Пример настройки по умолчанию можно найти в *Руководство по настройке лог-коллектора — Включение компонентов*.
   Основные параметры, которые нужно указать при включении компонентов сбора:

```
collectors:
event_log:
- <<: *<"id компонента сбора"> (ID компонента сбора, который указывали при
объявлении компонента сбора)
```

   Основные параметры, которые нужно указать при включении компонентов отправки:

```
senders:
tcp:
- <<: *<"id компонента отправки"> (ID компонента отправки, который указывали
при объявлении компонента отправки)
```

4. После чего необходимо произвести настройку маршрутизации событий.
   Пример настройки по умолчанию можно найти в *Руководство по настройке лог-коллектора — Маршрутизация событий*.
   Основные параметры, которые нужно указать при настройке маршрута:

```
route_1: &route_1

collector_id:

- <"id компонента сбора">

sender_id:

- <"id компонента отправки">
```

5. После нужно включить маршрут в разделе routers. Пример включения маршрута:

```
routers:

- <<: *<название маршрута> (например - <<: *route_1)
```

4.2. Создание учетной записи Microsoft Windows. {#create_account}

4.2.1. Создание учетной записи

Для создания учетной записи необходимо выполнить следующие действия:

1. В панели управления Windows открыть консоль Computer Management (Управление компьютером).
2. В консоли открыть раздел:

   System Tools (Служебные программы) → Local Users and Groups (Локальные пользователи и группы) → Users (Пользователи).
3. В контекстном меню раздела Users (Пользователи) выбрать функцию New User (Новый пользователь) для создания нового пользователя (см. рисунок 8).

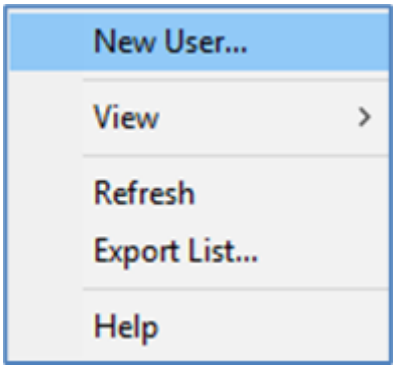

Рисунок 8 - Выбор функции создания нового пользователя.

4. В открывшемся окне New User (Новый пользователь) ввести следующие данные (см. рисунок 9):
   - В поле Name (Имя) ввести имя нового пользователя.
   - Установить пароль в поле Password (Пароль) и подтвердить его в поле Confirm Password (Подтвердить).
   - При необходимости выставить настройки в пунктах:
   - User cannot change password (Запретить смену пароля пользователем).

- Password never expires (Срок действия пароля неограничен).

5. Для создания пользователя с заданными параметрами нажать кнопку Create (Создать - см. рисунок 9).

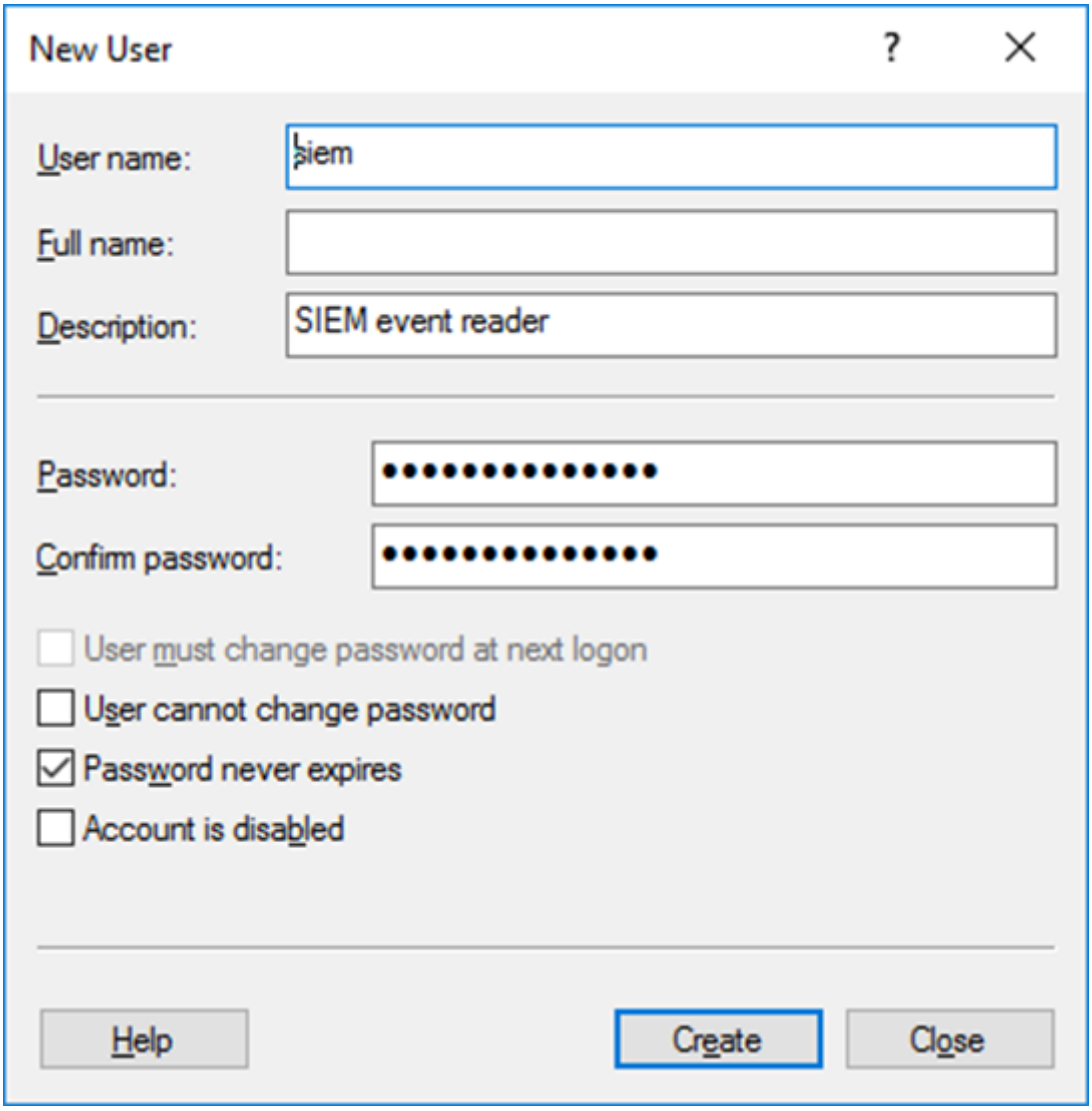

Рисунок 9 - Ввод данных нового пользователя.

4.2.2. Предоставление пользователю прав доступа к журналу событий

Для добавления пользователя в группу Event Log Readers (с правом доступа к журналам событий) необходимо выполнить следующие действия:

1. В консоли Computer Management (Управление компьютером) открыть раздел:

   System Tools (Служебные программы) → Local Users and Groups (Локальные пользователи и группы) → Groups (Группы)

2. Выбрать в списке группу Event Log Readers (Читатели журнала событий) (см. рисунок 10).

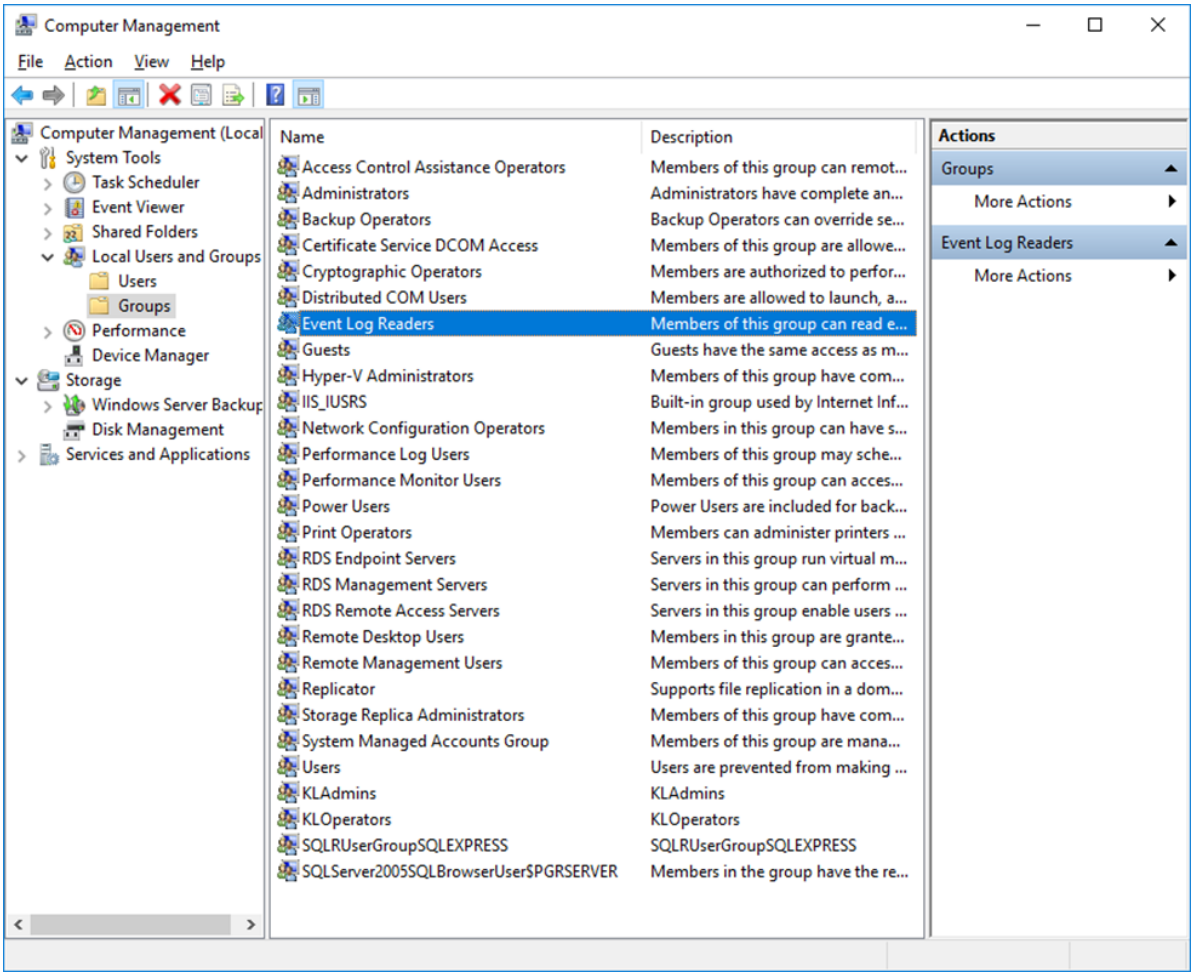

Рисунок 10 - Выбор группы Event Log Readers для включения учетной записи.

3. Открыть правой кнопкой мыши контекстное меню группы Event Log Readers (Читатели журнала событий) и выбрать пункт Add To Group (Добавить в группу). Откроется окно Event Log Readers Properties (Свойства: Читатели журнала событий) (см. рисунок 11).
4. Для добавления пользователя в группу:
    - Нажать кнопку Add (Добавить).
    - В открывшемся окне Select Users (Выбор: Пользователи) выбрать в списке пользователя, созданного ранее, и добавить его в группу, нажав кнопку OK.
5. Для сохранения введенных настроек в окне Event Log Readers Properties (Свойства: Читатели журнала событий) нажать кнопку OK (см. рисунок 11).

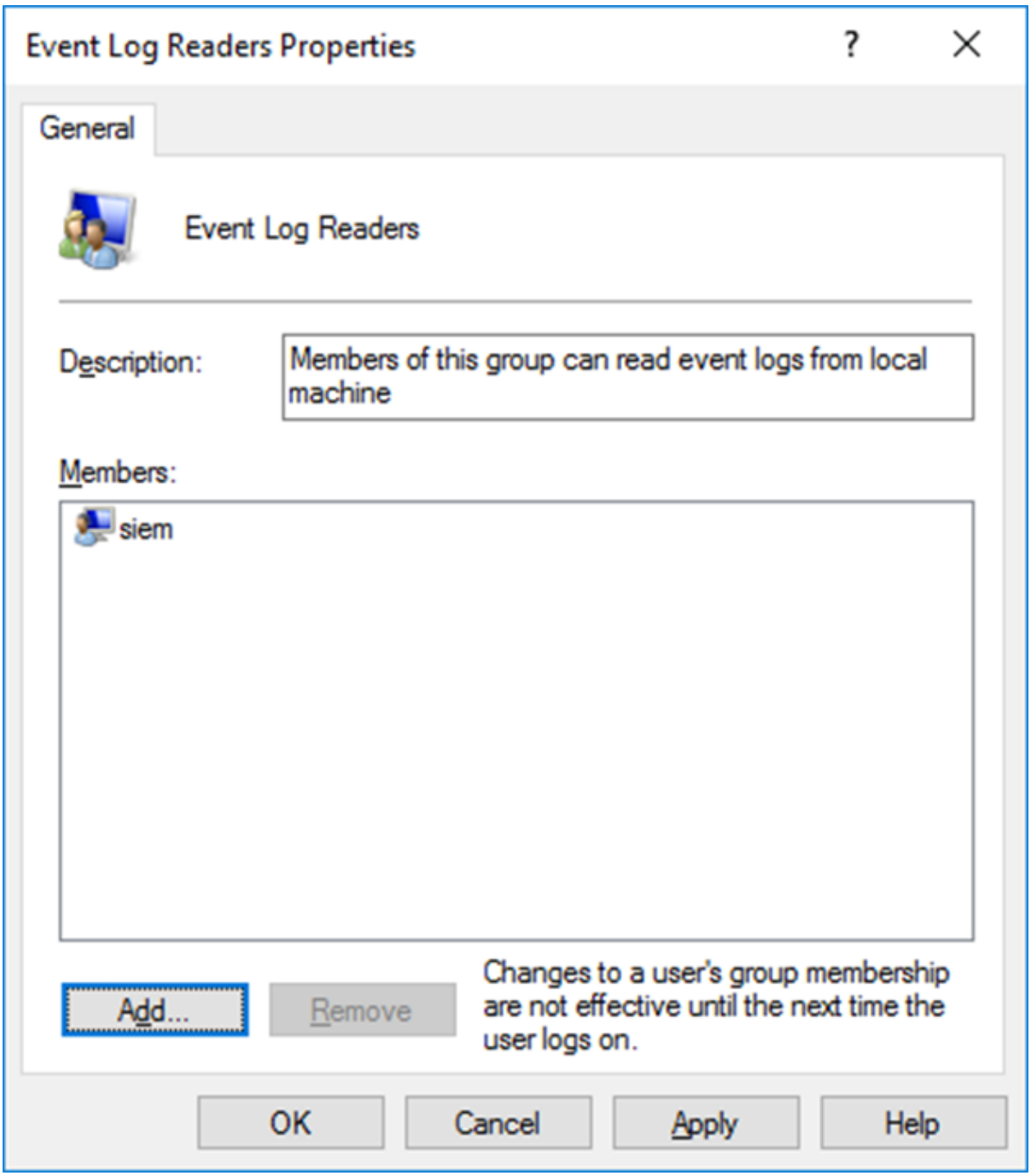

Рисунок 11 - Добавление пользователя в группу Event Log Readers.

Внесенные изменения вступают в действие при следующем входе нового пользователя в систему.

4.3. Настройка расширенных политик аудита Windows {#audit}

Для настройки политик аудита на контроллерах домена используются групповые политики домена, которые необходимо сконфигурировать в соответствии с представленной инструкцией:

В групповой политике, применяемой для контроллеров домена, необходимо включить политику использования расширенной конфигурации политики аудита «Audit: Force audit policy subcategory settings (Windows Vista or later) (Аудит: принудительно переопределяет параметры категории политики аудита параметрами подкатегории политики аудита (Windows Visa или следующие версии))».

Данную политику необходимо включить в разделе «Computer Configuration (Конфигурация компьютера)» → «Windows Settings (Конфигурация Windows)» → «Security Settings (Параметры безопасности)» → «Local Policies (Локальные политики)» → «Security Options (Параметры безопасности)» (см. рисунок 12).

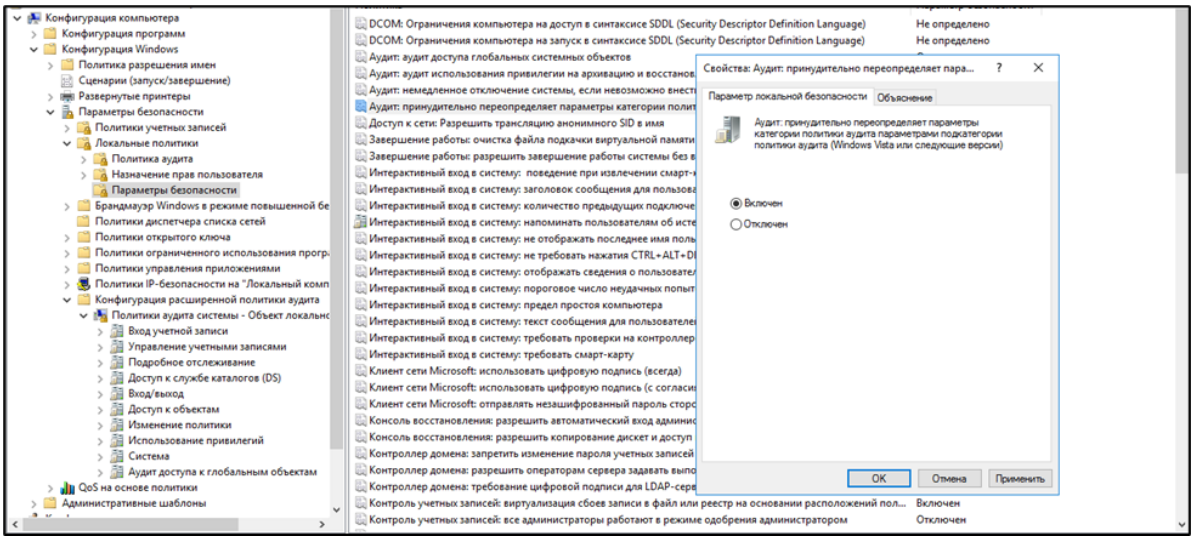

Рисунок 12 - Добавление Audit: Force audit policy subcategory settings

Для активации аудита для контроллеров домена необходимо настроить групповую политику, которая распространяется на контейнер содержащий DC (Контроллеры домена), в соответствии с таблицей 1. (см. рисунок 13).

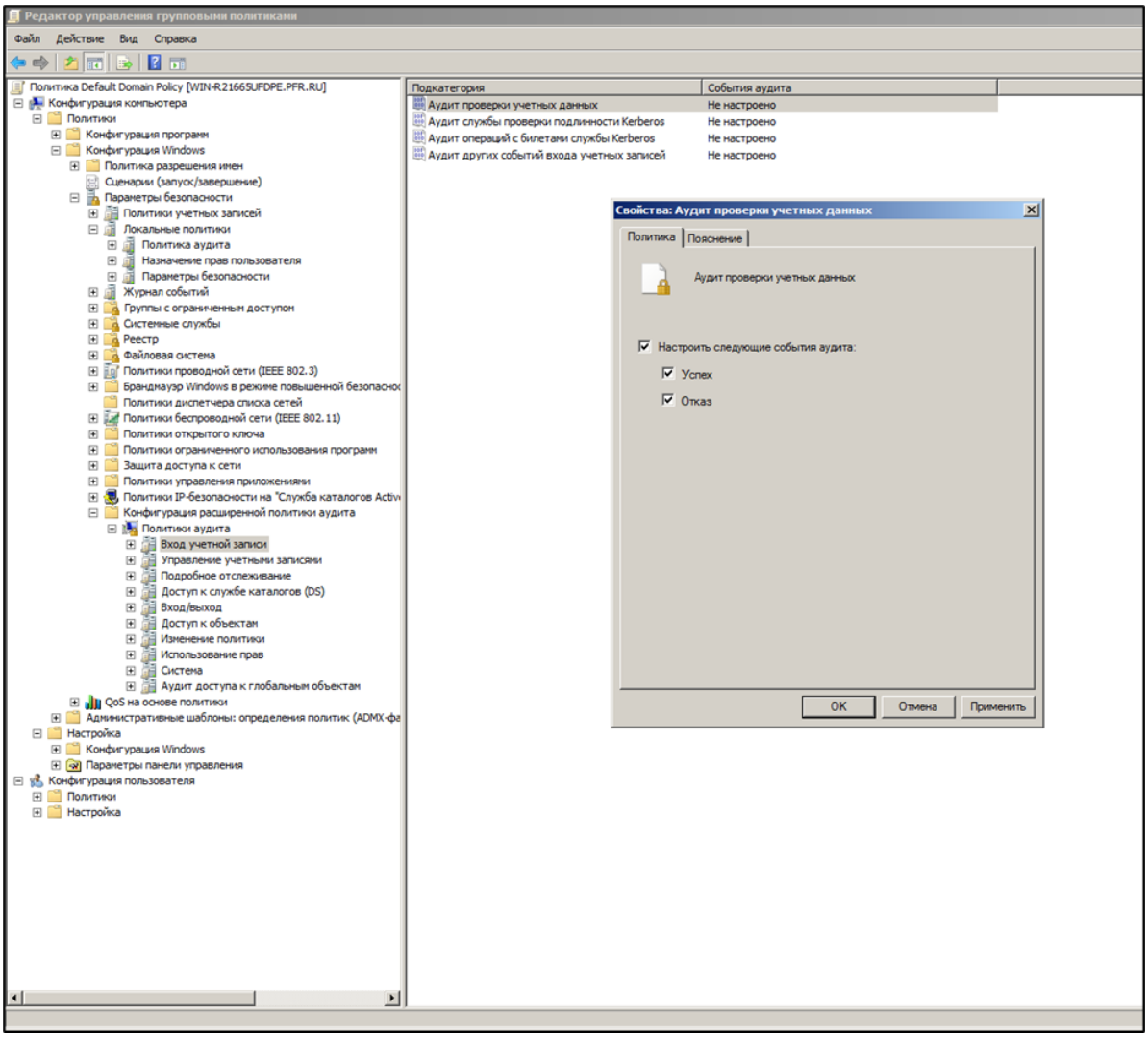

Рисунок 13 - Изменение политик аудита.

Таблица 1 -- Политики аудита ОС Windows 2008/2012

Политика аудита	**Тип событий**
Аудит проверки учетных данных (Account Logon→Audit Credential Validation)	Успех и Отказ
Аудит службы проверки подлинности Kerberos (Account Logon→Audit Kerberos Authentication Service)	Успех и Отказ
Аудит операций с билетами службы Kerberos (Account Logon→Audit Kerberos Service Ticket Operations)	Успех и Отказ
Аудит других событий входа учетных записей (Account Logon→Audit Other Account Logon Events)	Успех и Отказ
Аудит управления группами приложений(Account Management→Audit Application Group Management)	Успех и Отказ
Аудит управления учетными записями компьютеров (Account Management→Audit Computer Account Management)	Успех и Отказ
Аудит управления группами распространения (Account Management→Audit Distribution Group Management)	Успех и Отказ
Аудит других событий управления учетными записями (Account Management→Audit Other Account Management Events)	Успех и Отказ
Аудит управления группами безопасности (Account Management→Audit Security Group Management)	Успех и Отказ
Аудит управления учетными записями (Account Management→Audit User Account Management)	Успех и Отказ
Аудит активности DPAPI(Detailed Tracking→Audit DPAPI Activity)	Не фиксируются
Аудит создания процессов (Detailed Tracking→Audit Process Creation)	Успех и Отказ
Аудит завершения процессов (Detailed Tracking→Audit Process Termination)	Успех и Отказ
Аудит событий RPC (Detailed Tracking→Audit RPC Events)	Не фиксируются
Аудит подробной репликации службы каталогов (DS Access→Audit Detailed Directory Service Replication)	Не фиксируются
Аудит доступа к службе каталогов (DS Access→Audit Directory Service Access)	Успех и Отказ
Аудит изменения службы каталогов (DS Access→Audit Directory Service Changes)	Успех и Отказ

Политика аудита	Тип событий
Аудит репликации службы каталогов (DS Access→Audit Directory Service Replication)	Не фиксируются
Аудит блокировки учетных записей (Logon/Logoff→Audit Account Lockout)	Успех и Отказ
Аудит расширенного режима IPsec (Logon/Logoff→Audit IPsec Extended Mode)	Не фиксируются
Аудит основного режима IPsec (Logon/Logoff→Audit IPsec Main Mode)	Не фиксируются
Аудит быстрого режима IPsec (Logon/Logoff→Audit IPsec Quick Mode)	Не фиксируются
Аудит выхода из системы (Logon/Logoff→Audit Logoff)	Успех
Аудит входа в систему (Logon/Logoff→Audit Logon)	Успех и Отказ
Аудит сервера политики сети (Logon/Logoff→Audit Network Policy Server)	Не фиксируются
Аудит других событий входа/выхода (Logon/Logoff→Audit Other Logon/Logoff Events)	Успех и Отказ
Аудит специального входа (Logon/Logoff→Audit Special Logon)	Успех и Отказ
Аудит событий, создаваемых приложениями(Object Access→ Audit Application Generated)	Не фиксируются
Аудит сведений об общем файловом ресурсе (Object Access→ Audit Detailed File Share)	Не фиксируются
Аудит общего файлового ресурса (Object Access→ Audit File Share)	Успех и Отказ
Аудит файловой системы (Object Access→ Audit File System)	Успех и Отказ
Аудит подключения фильтрации (Object Access→ Audit Filtering Platform Connection)	Не фиксируются
Аудит отбрасывания пакетов фильтрации (Object Access→ Audit Filtering Platform Packet Drop)	Не фиксируются
Аудит работы с дискрипторами(Object Access→ Audit Handle Manipulation)	Не фиксируются
Аудит объектов ядра (Object Access→ Audit Kernel Object)	Не фиксируются

Политика аудита	Тип событий
Аудит других событий доступа к объектам(Object Access→ Audit Other Object Access Events)	Не фиксируются
Аудит реестра (Object Access → Audit Registry)	Успех и Отказ
Аудит диспетчера учетных записей безопасности (Object Access → Audit SAM)	Не фиксируются
Аудит изменения политики аудита (Policy Change→ Audit Policy Change)	Успех и отказ
Аудит изменения политики проверки подлинности (Policy Change→Audit Audit Authorization Policy Change)	Успех и Отказ
Аудит изменения политики авторизации (Policy Change→Audit Authorization Policy Change)	Успех и Отказ
Аудит изменения политики фильтрации (Policy Change→Audit Filtering Platform Policy Change)	Не фиксируются
Аудит изменения политики на уровне правил MPSSVC (Policy Change→Audit MPSSVC Rule-Level Policy Change)	Успех и Отказ
Аудит других событий изменения политики (Policy Change→Audit Other Policy Change Events)	Успех и Отказ
Аудит использования привилегий, затрагивающих конфиденциальные данные (Privilege Use→Audit Sensitive Privilege Use)	Успех и Отказ
Аудит использования привилегий, не затрагивающих конфиденциальные данные (Privilege Use→Audit Non-Sensitive Privilege Use)	Успех и Отказ
Аудит драйвера IPsec (System→Audit IPsec Driver)	Не фиксируются
Аудит других системных событий (System→Audit Other System Events)	Не фиксируются
Аудит изменения состояния безопасности (System→Audit Security State Change)	Успех и Отказ
Аудит расширения системы безопасности (System→Audit Security System Extension)	Успех и Отказ
Аудит целостности системы (System→Audit System Integrity)	Успех и Отказ

4.4. Microsoft Windows Event Forwardig (WEC) {#wec}

4.4.1. Настройка WEC вне домена

4.4.1.1. Настройка пересылки событий, инициированной сборщиком

Перед настройкой необходимо:

- разрешить между источником и сборщиком сетевое взаимодействие по портам 5985/TCP и 5986/TCP;
- на источнике добавить «Network Service» и используемого пользователя в «Local Computer Policy/Computer Configuration/Windows Settings/Security Settings/Local Policy/User Rights Assignment/Manage auditing and security log»;

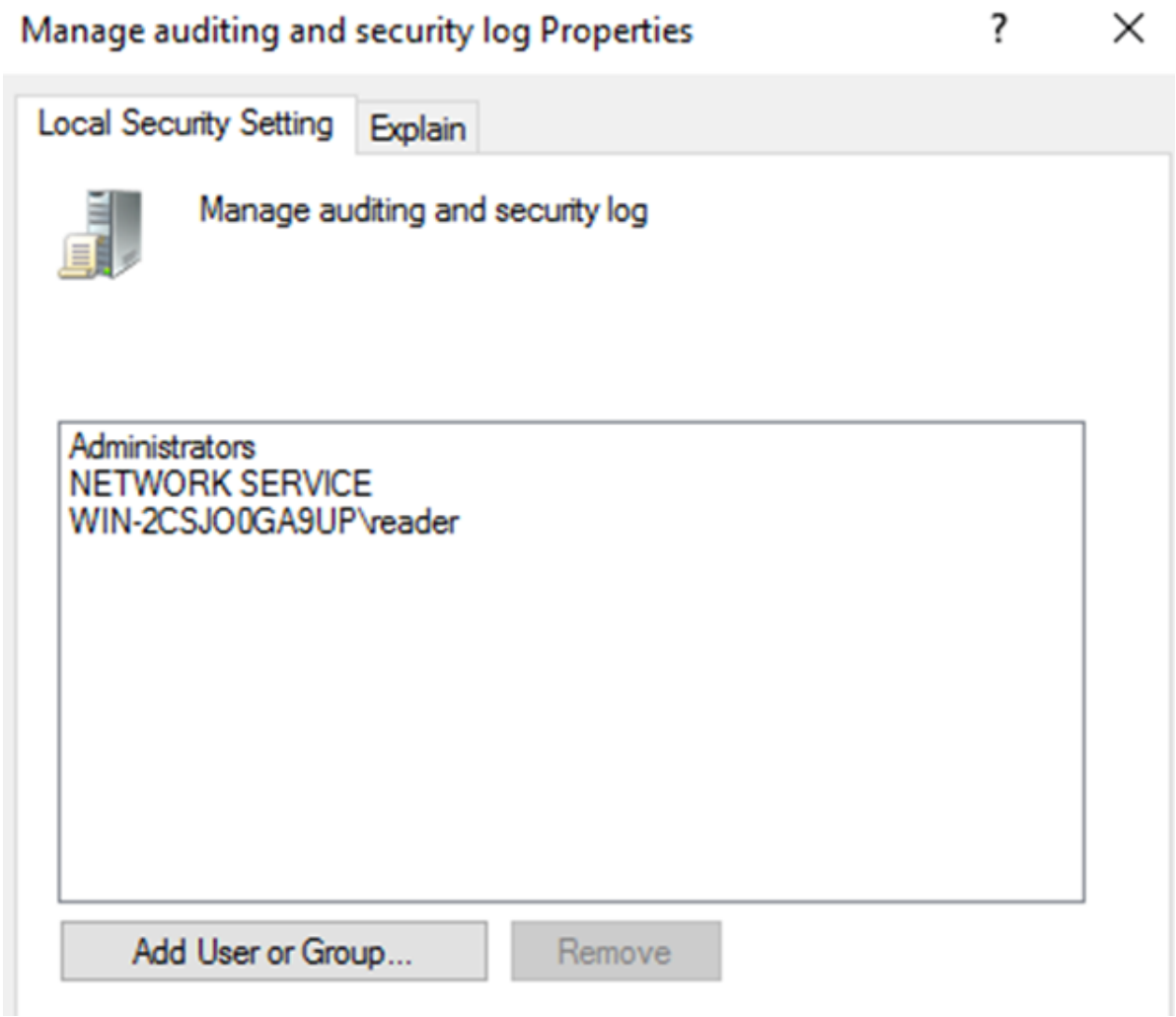

Рисунок 14

- выключить на источнике и сборщике проверку подлинности NTLM в «Local Computer Policy/Computer Configuration/Windows Settings/Security Settings/Local Policy/Security Option»(опционально);
- на источнике добавить «Network Service» и используемого пользователя в группу «Event Log Readers»;
- установить последние обновления для операционной системы и перезагрузить (исправляют известные проблемы с WinRM и .Net).

4.4.1.1.1. Настройка источника событий

1. Открыть командную строку с правами администратора системы и в ней последовательно выполнить следующие команды:

```
winrm qc -q
wecutil qc /q
winrm set winrm/config/client @{TrustedHosts="ip_адрес_сборщика"}
```

2. Создать учетную запись и добавить ее в группу «Читатели журнала событий».

4.4.1.1.2. Настройка сборщика событий

1. Открыть командную строку с правами администратора системы и в ней последовательно выполнить следующие команды:

```
winrm qc -q
wecutil qc /q
winrm set winrm/config/client @{TrustedHosts="ip_адрес_источника"}
```

2. Зайти в «Просмотр событий» и создать подписку:
    - Нажать правой кнопкой по пункту «Подписки» и выбрать «Создать подписку».

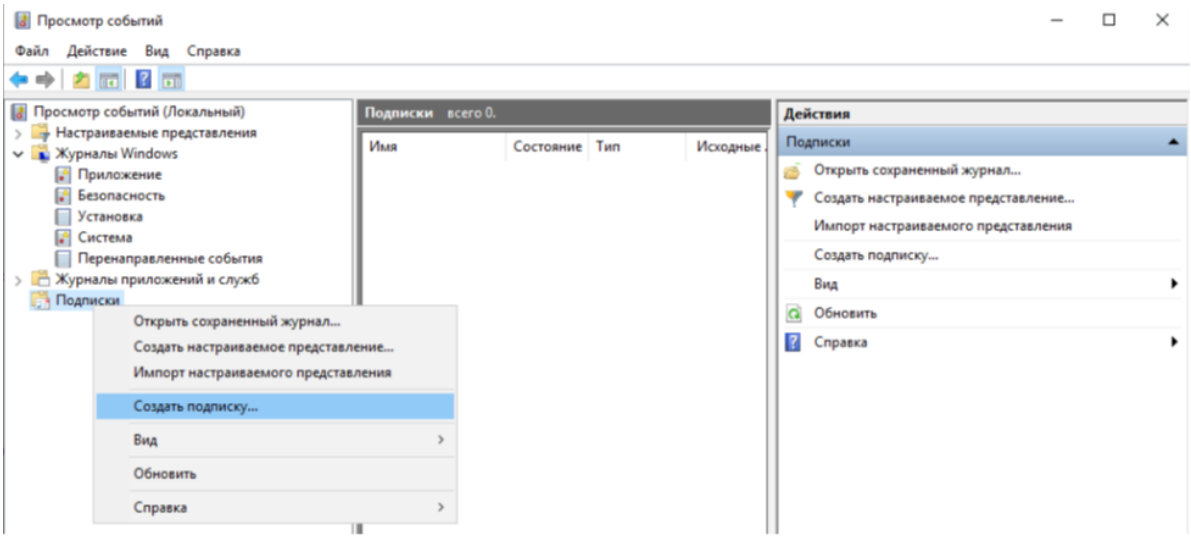

Рисунок 15

```
- В открывшемся окне ввести имя подписки, выбрать конечный журнал для получаемых событий и тип подписки «Инициировано сборщиком» и нажать «Выбрать компьютеры».
```

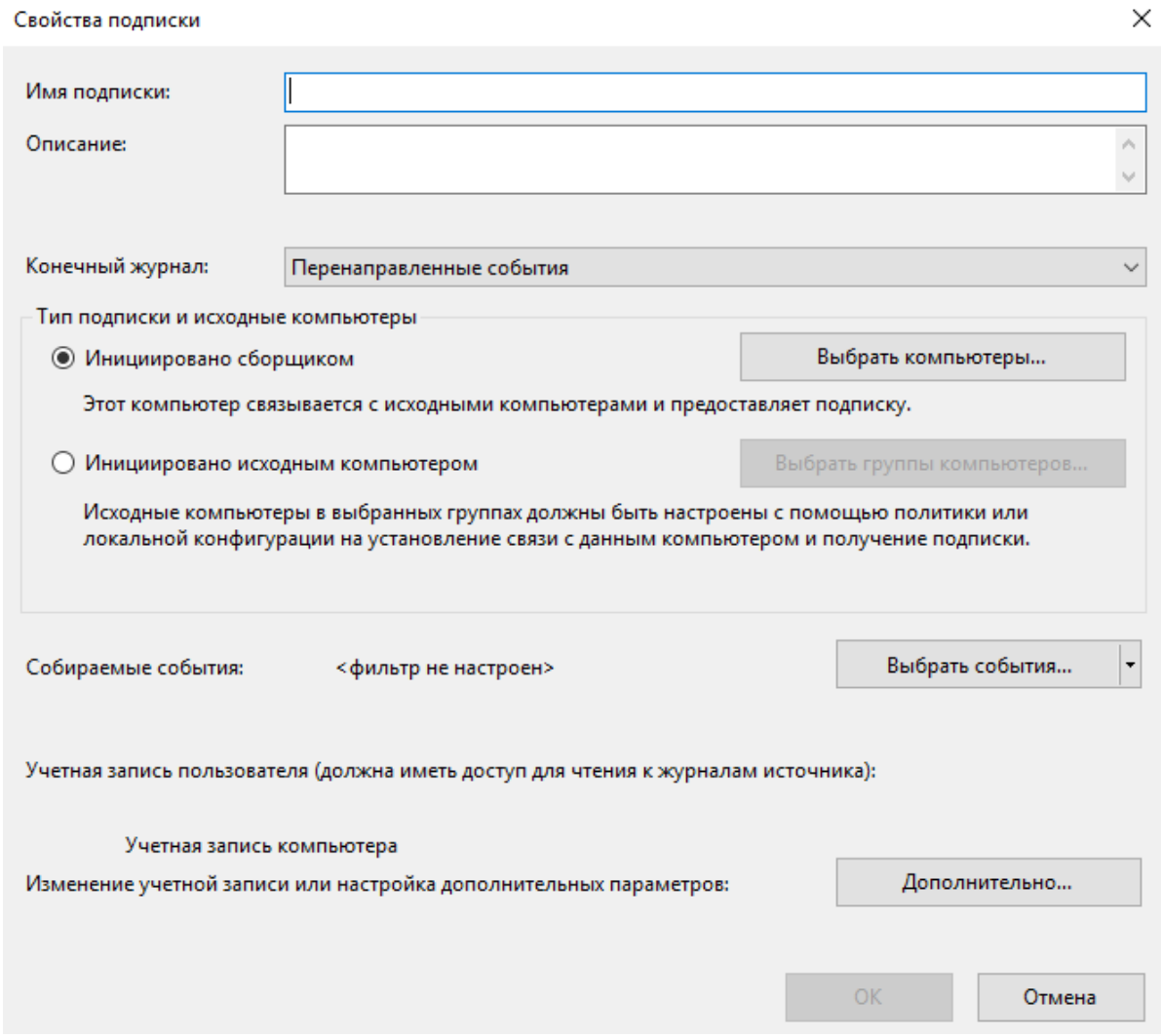

Рисунок 16

```
- Нажать «Добавить доменный компьютер».
```

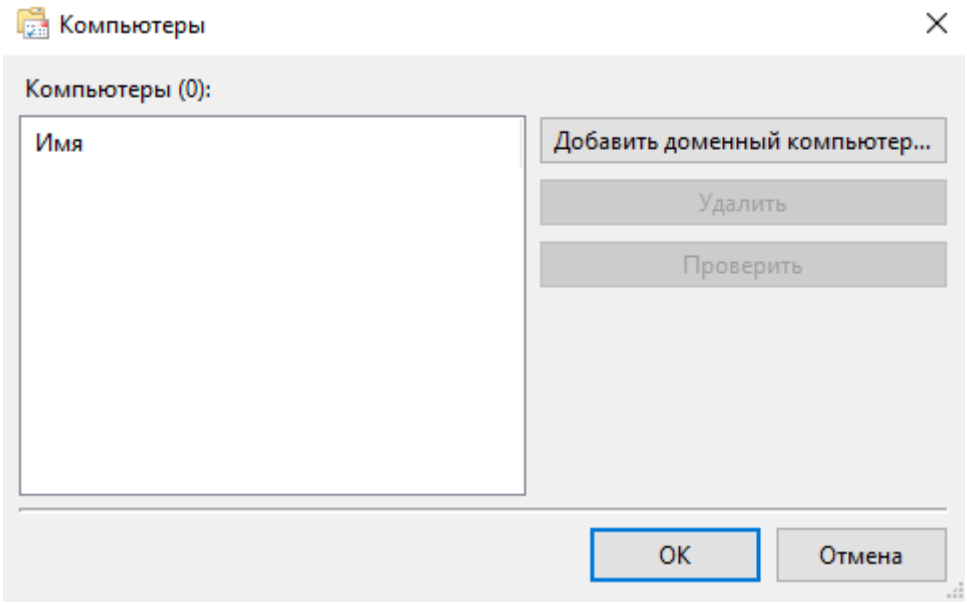

Рисунок 17

```
- Ввести IP-адрес или DNS-имя источника и нажать «OK».
```

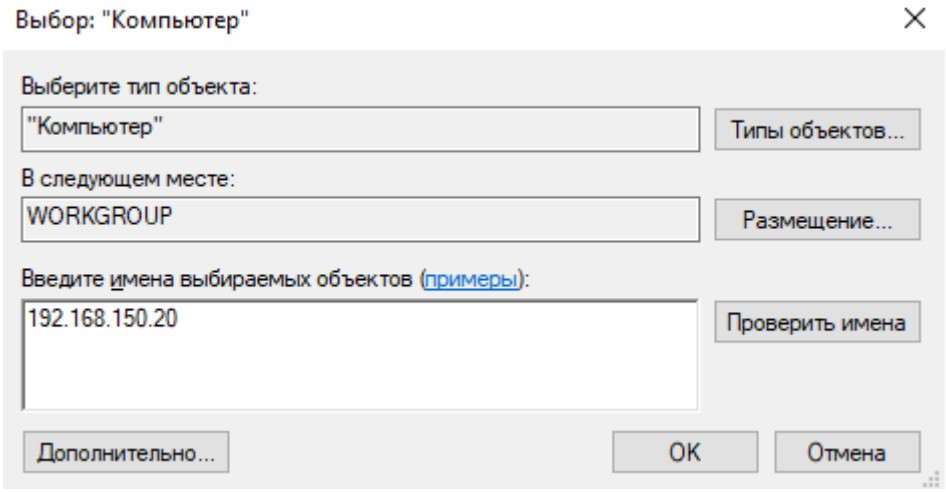

Рисунок 18

- Нажать «Выбрать события», настроить фильтр для запроса необходимых событий и нажать «ОК».

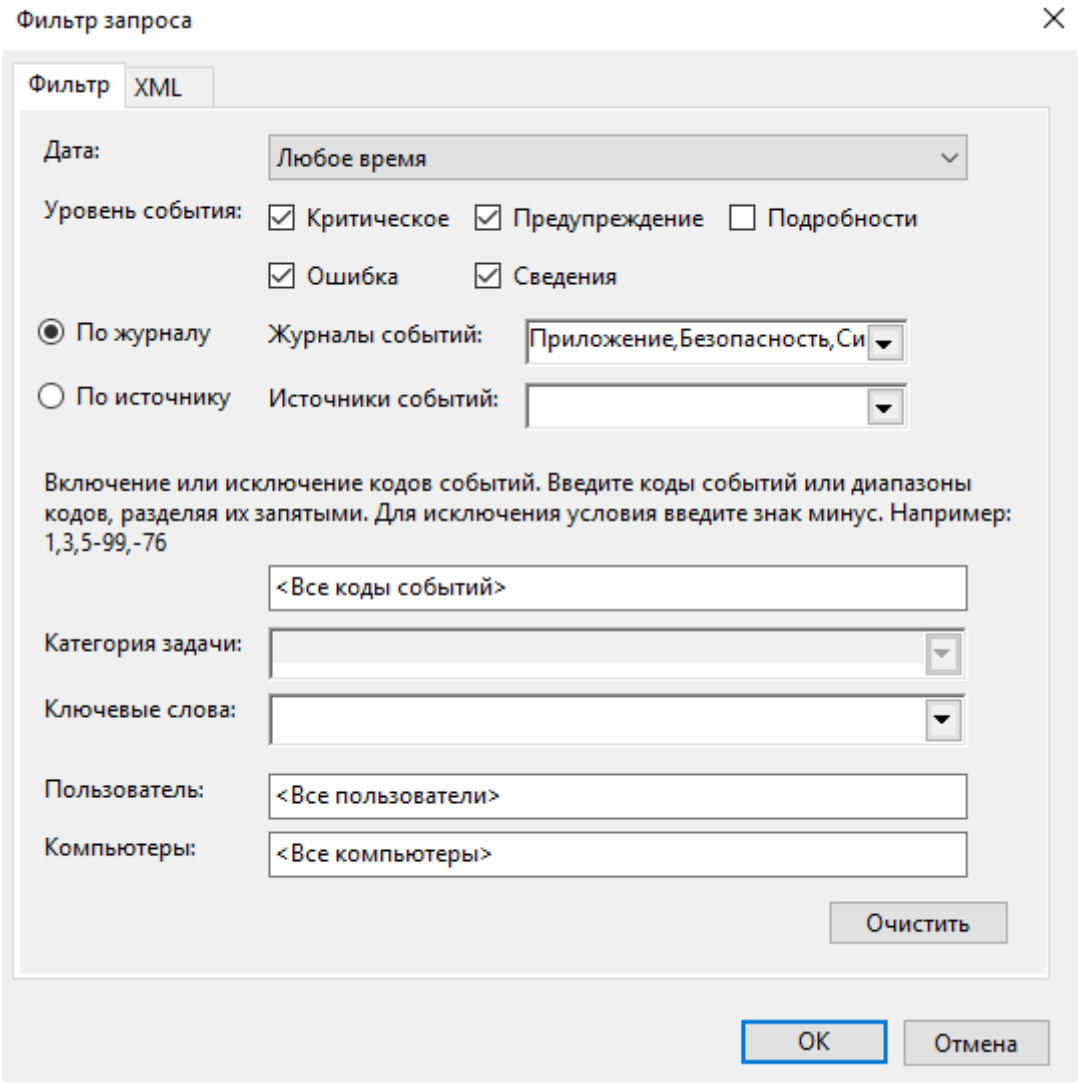

Рисунок 19

- Нажать «Дополнительно» и выбрать «Определенный пользователь», остальные параметры оставить по умолчанию.

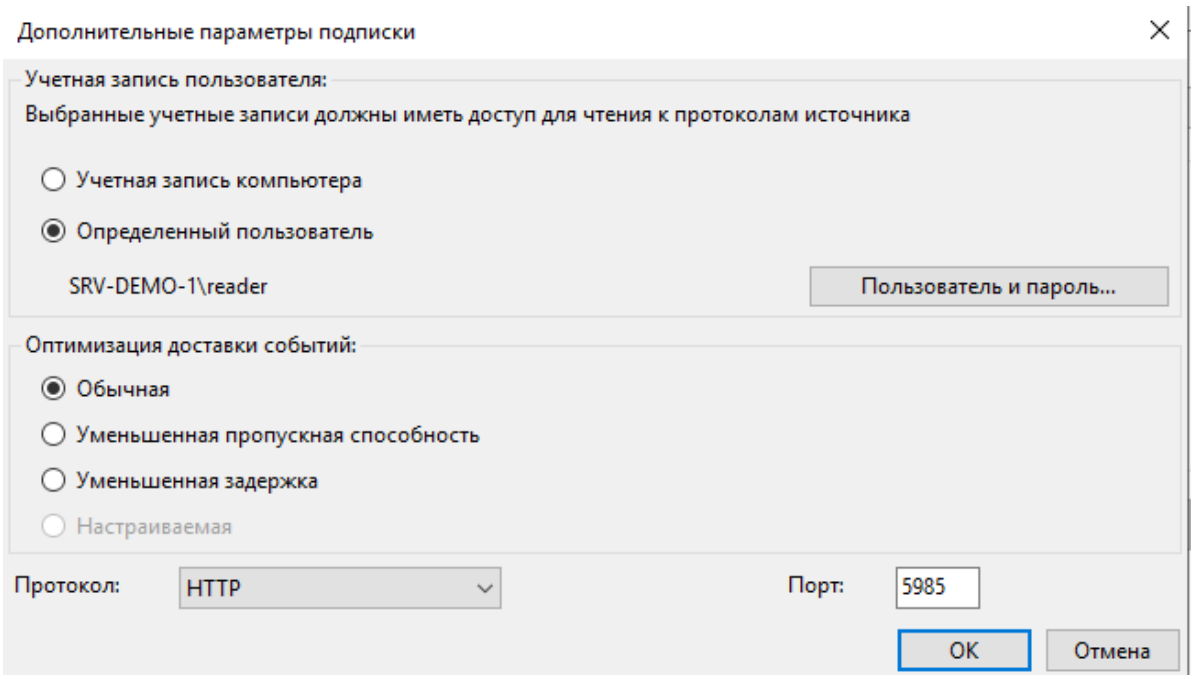

Рисунок 20

```
- Нажать «Пользователь и пароль», внести учетные данные пользователя, созданного
на сервере-источнике событий, и затем сохранить изменения.
```

3. После создания подписки проверить её статус, нажав по ней правой кнопкой и выбрав «Состояние выполнения».

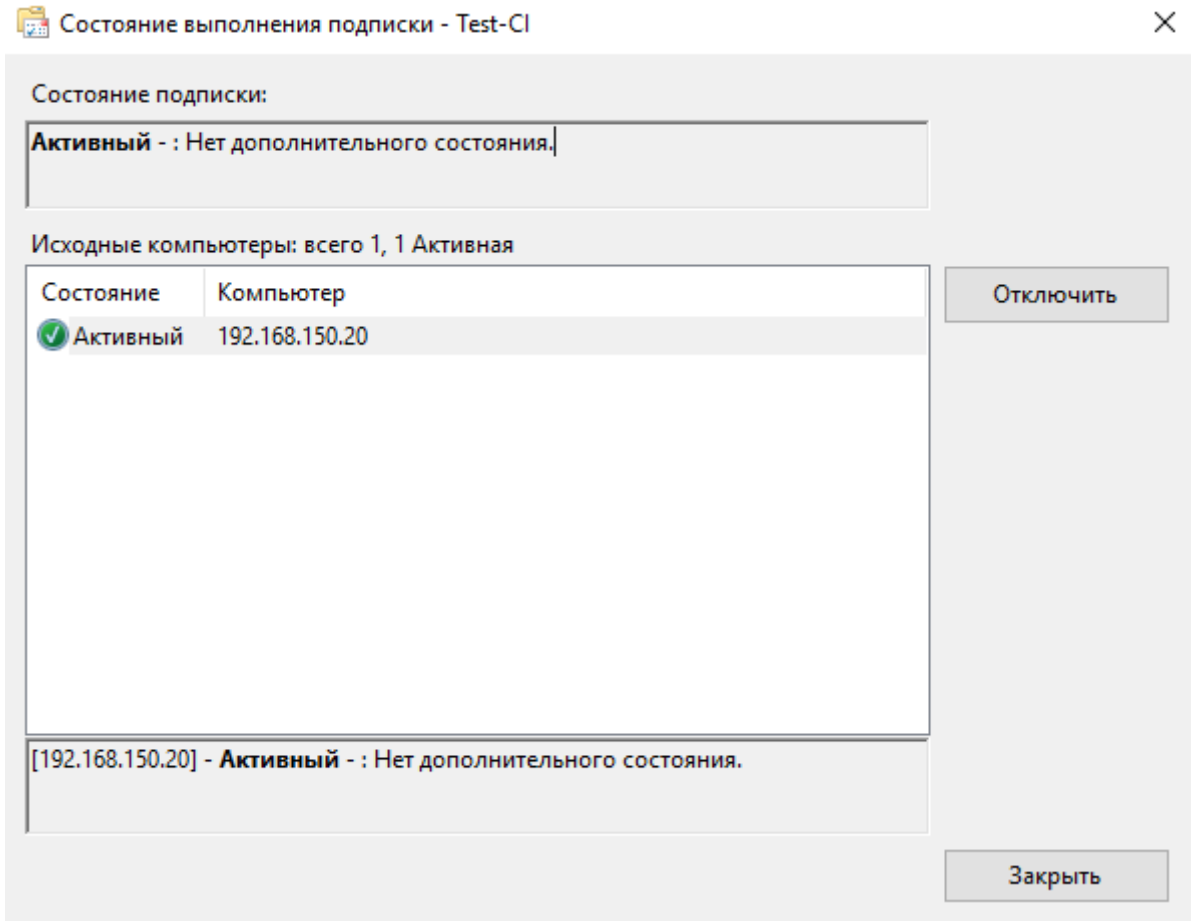

Рисунок 21

4. Проверить поступление событий в указанный в подписке журнал.

4.4.1.2. Настройка пересылки событий, инициированной источником

Перед настройкой необходимо разрешить на файерволе сборщика сетевое взаимодействие по порту 5986/tcp.
Также необходимо выпустить и установить сертификаты проверки подлинности клиента и сервера в соответствии со следующими требованиями:

- сертификат проверки подлинности сервера должен быть установлен на компьютере сборщика событий в личном хранилище локального компьютера, субъект этого сертификата должен соответствовать полному доменному имени сборщика;
- сертификат проверки подлинности клиента должен быть установлен на компьютерах источника событий в личном хранилище локального компьютера, субъект этого сертификата должен соответствовать полному доменному имени источника;
- сертификат удостоверяющего центра должен быть установлен на всех компьютерах в «Доверенные корневые центры сертификации»; если сертификат клиента выдан центром сертификации, отличным от одного из сборщиков событий, эти корневые и промежуточные сертификаты также должны быть установлены на сборщике событий;
- проверить у сертификата проверки подлинности клиента наличие разрешения на чтение для пользователя NETWORK SERVICE:

  `Консоль управления сертификатами\правой кнопкой по сертификату\Управление закрытыми ключами`
- если сертификат клиента был выдан промежуточным центром сертификации, а сборщик работает Windows 2012 или более поздней версии, необходимо настроить следующий раздел реестра:

  `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\SecurityProviders\Schannel\ClientAuthTrustMode (DWORD) = 2`

Проверить состояние отзыва сертификатов можно следующим образом:

`certutil -verify -urlfetch <путь до файла сертификата>`

4.4.1.2.1. Настройка источника событий

1. Открыть командную строку с правами администратора системы и в ней выполнить следующую команду:

   `winrm qc -q`
2. Открыть редактор локальной групповой политики (gpedit.msc) и перейти в раздел: Политика локального компьютера\Конфигурация компьютера\Административные шаблоны\Windows Компоненты\Пересылка событий.

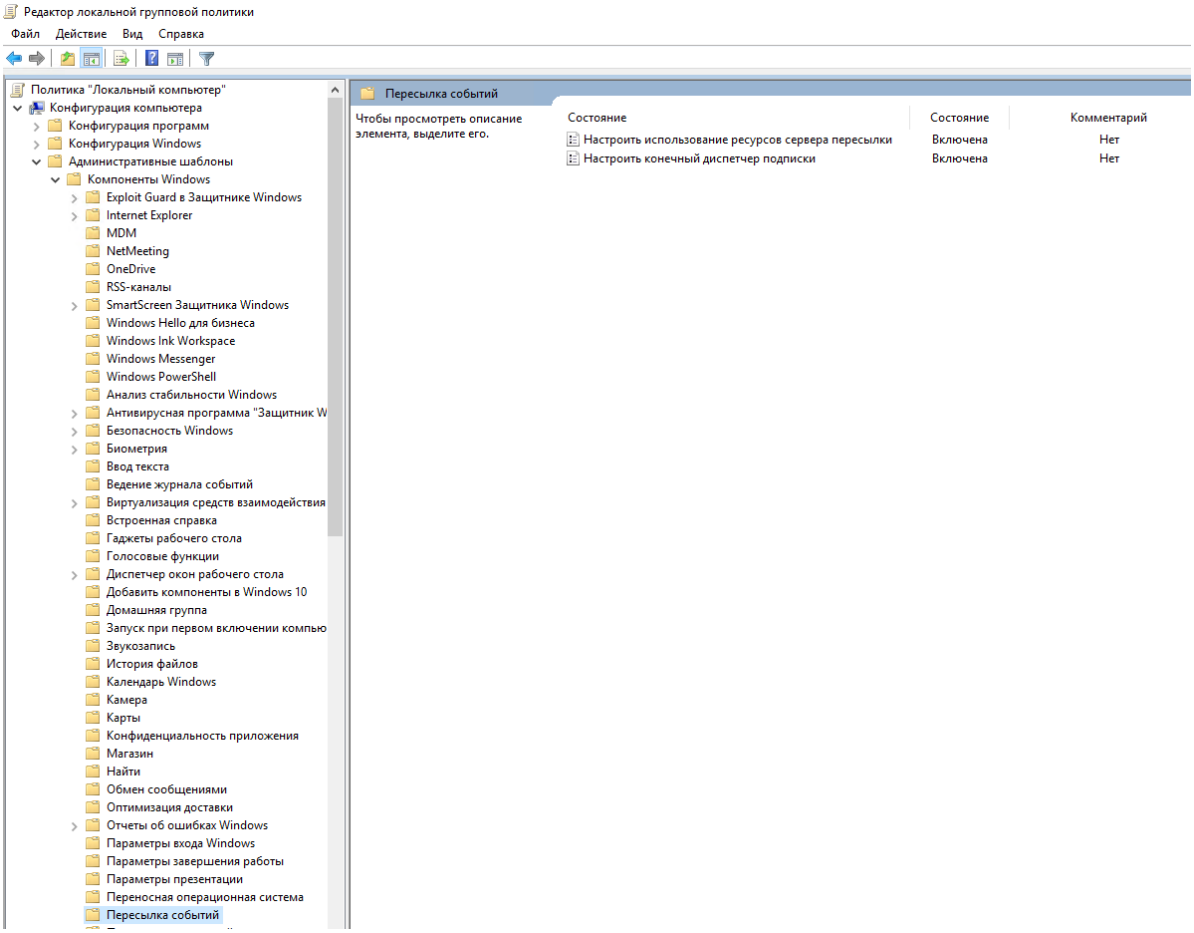

Рисунок 22

3. Открыть элемент «Настроить конечный диспетчер подписки», включить его и нажать кнопку «Показать».

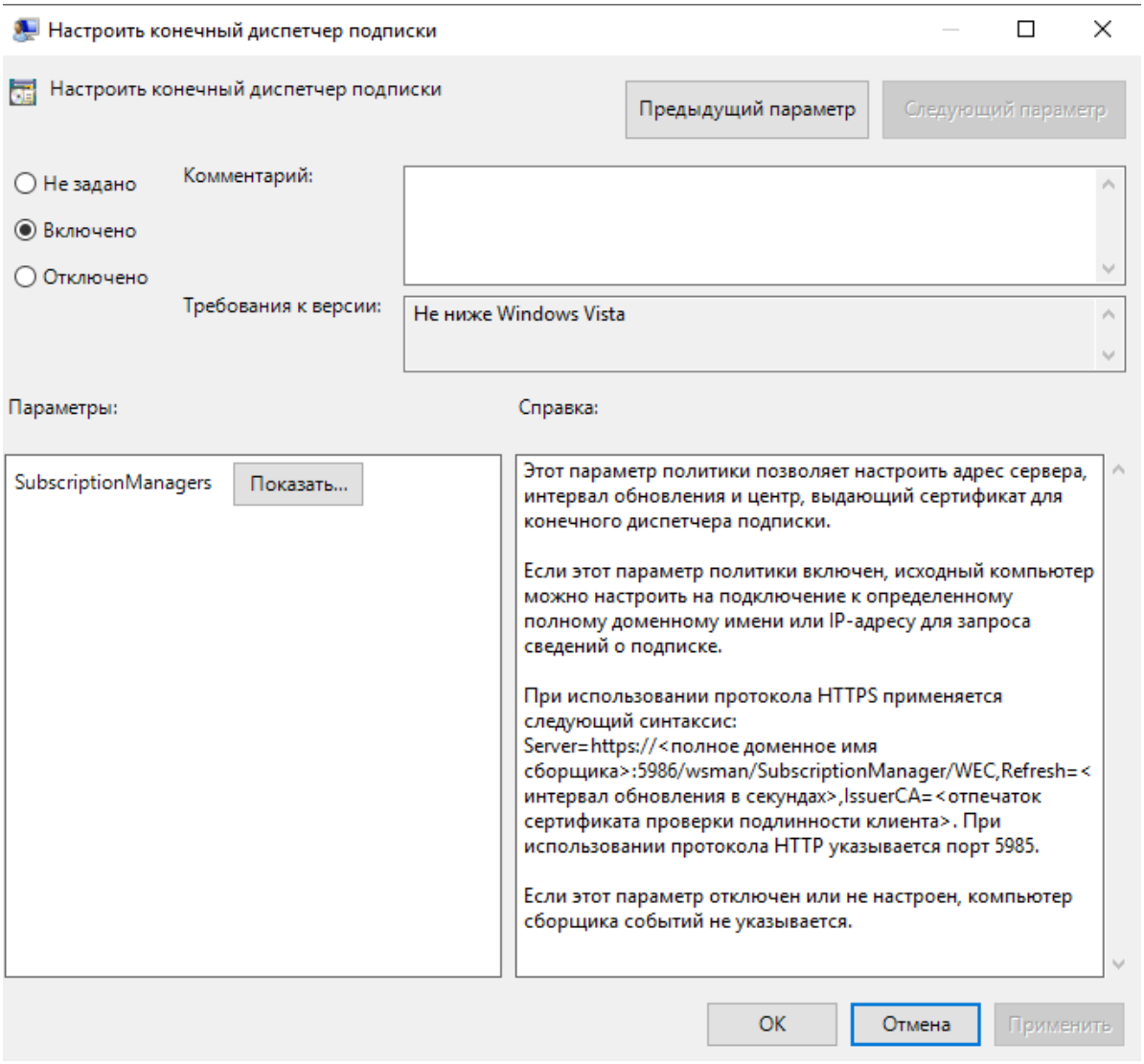

Рисунок 23

4. В открывшемся окне ввести следующий параметр и нажать «ОК»:

   `Server=https://<FQDN сборщика>:5986/wsman/SubscriptionManager/WEC,Refresh=60,IssuerCA=<Отпечаток сертификата CA>`

5. В открытой командной строке с правами администратора системы выполнить следующую команду:

   `gpupdate /force`

4.4.1.2.2. Настройка сборщика событий

1. Открыть командную строку с правами администратора системы и в ней последовательно выполнить следующие команды:

```
winrm qc -q
wecutil qc /q
winrm set winrm/config/service/auth @{Certificate="true"}
```

2. Ввести указанную ниже команду и проверить, что параметру «AllowUnencrypted» в разделах «Service» и «Client» присвоено значение «false»:

   `winrm get winrm/config`

Если присвоено значение «true», то ввести следующие команды:

```
winrm set winrm/config/service @{AllowUnencrypted="false"}
winrm set winrm/config/client @{AllowUnencrypted="false"}
```

3. Проверить настройки прослушивателя WinRM:

   `winrm e winrm/config/listener`

   Если в выводе команды «Transport=HTTP» и «Port=5985», то необходимо выполнить переключение на HTTPS и 5986.

4. Выполнить переключение прослушивателя WinRM на HTTPS, введя последовательно следующие команды:

```
winrm delete winrm/config/Listener?Address=*+Transport=HTTP
winrm create winrm/config/Listener?Address=*+Transport=HTTPS @{Hostname="
<FQDN сборщика>"; CertificateThumbprint="<Отпечаток  сертификата  проверки
подлинности  сервера>”}
```

5. Создать локального пользователя и добавить его в локальную группу администраторов.
6. Создать сопоставление сертификата, который присутствует в доверенных корневых центрах сертификации компьютера или промежуточных центрах сертификации, с созданным ранее пользователем:

```
winrm create winrm/config/service/certmapping?Issuer=<Отпечаток  сертификата
CA>+Subject=*+URI=* @{UserName=”<username>”;Password=”<password>”}
```

7. Проверить прослушиватель и сопоставление сертификатов можно следущими командами:
   - С клиента:

     ```
     winrm  get  winrm/config -r:https://<Полное_имя_сборщика>:5986 -
     a:certificate -certificate:”<Отпечаток сертификата проверки подлинности
     клиента>”
     ```

   - С сервера:

     ```
     winrm  enum  winrm/config/service/certmapping
     ```

8. Зайти в «Просмотр событий» и создать подписку:
   - Нажать правой кнопкой по пункту «Подписки» и выбрать «Создать подписку».

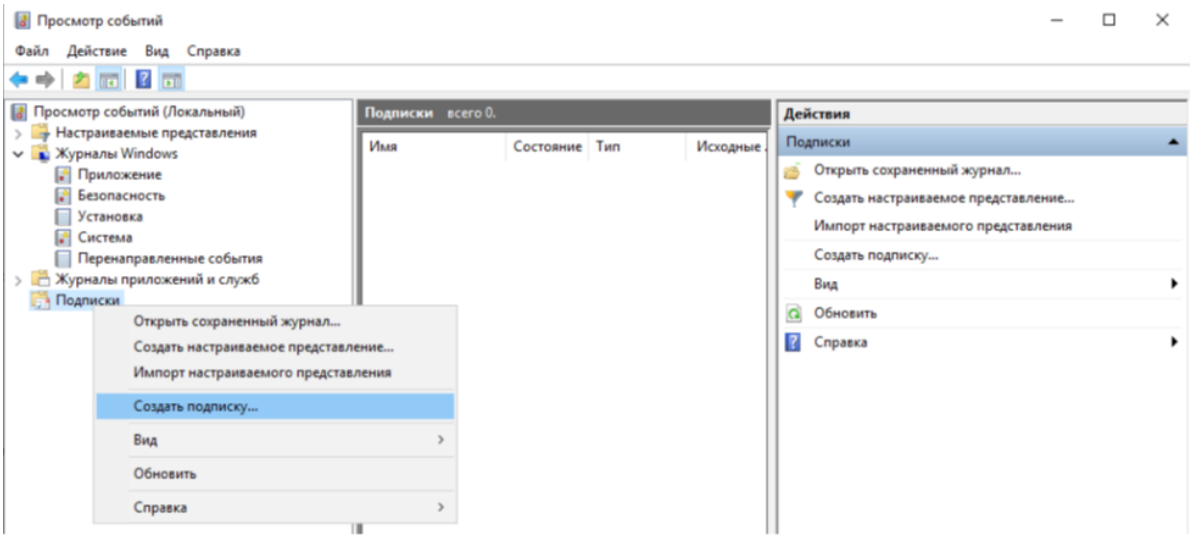

Рисунок 24

- В открывшемся окне ввести имя подписки, выбрать конечный журнал для получаемых событий и тип подписки «Инициировано исходным компьютером» и нажать «Выбрать группы компьютеров».

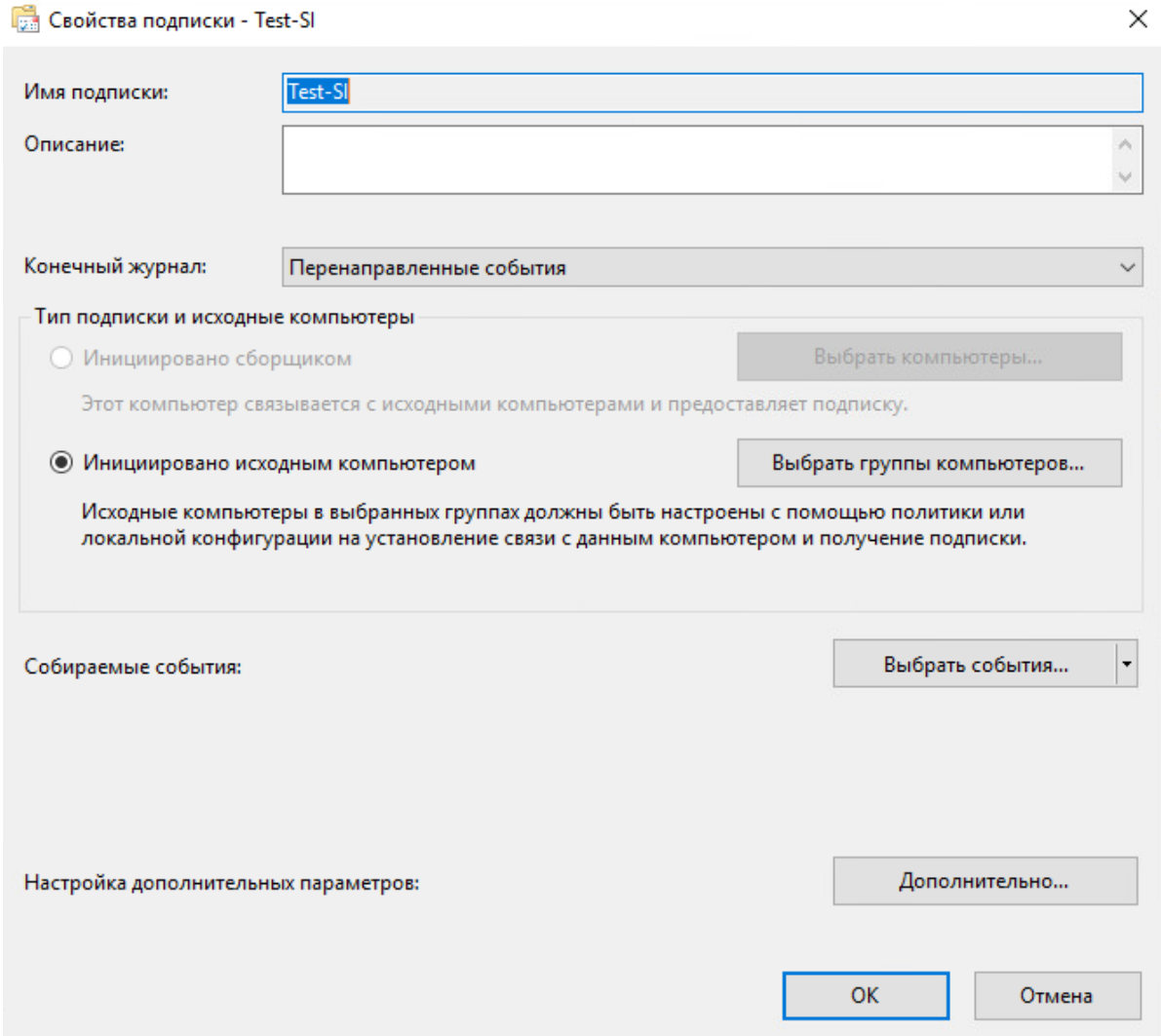

Рисунок 25

- Нажать «Добавить не доменный компьютер».

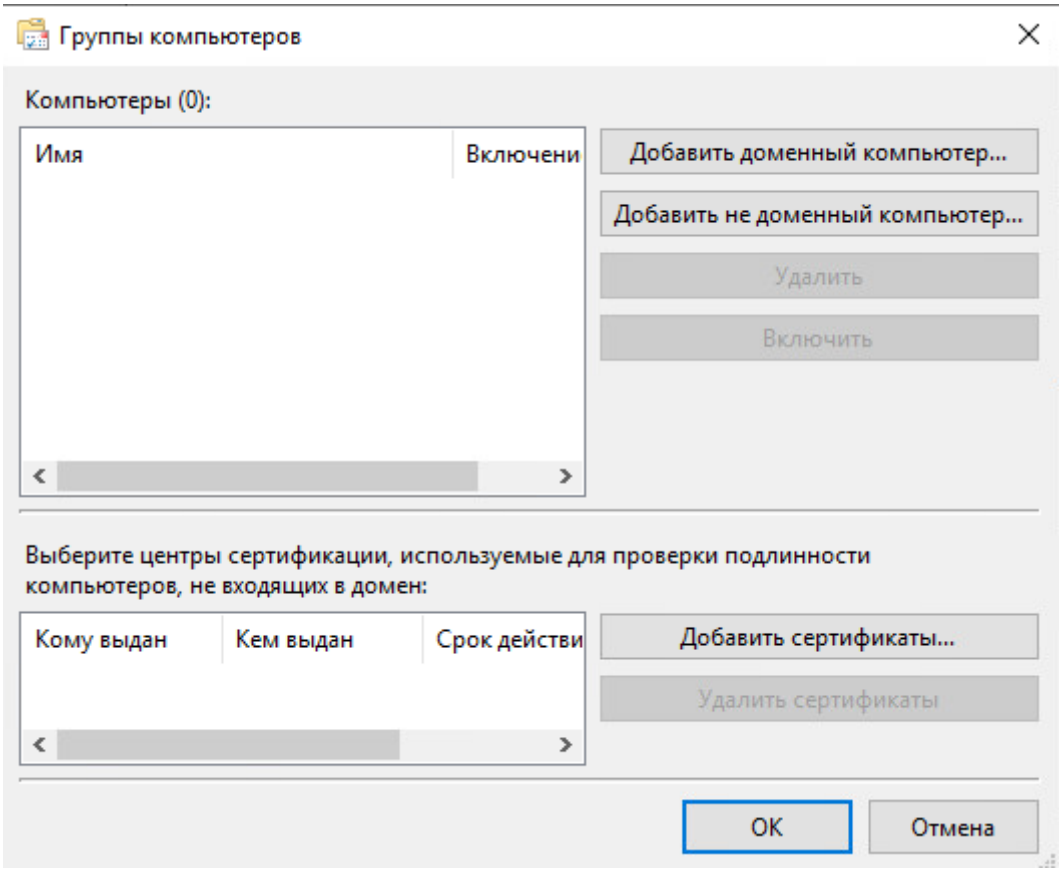

Рисунок 26

- В открывшемся окне ввести имя источника событий и нажать «ОК».

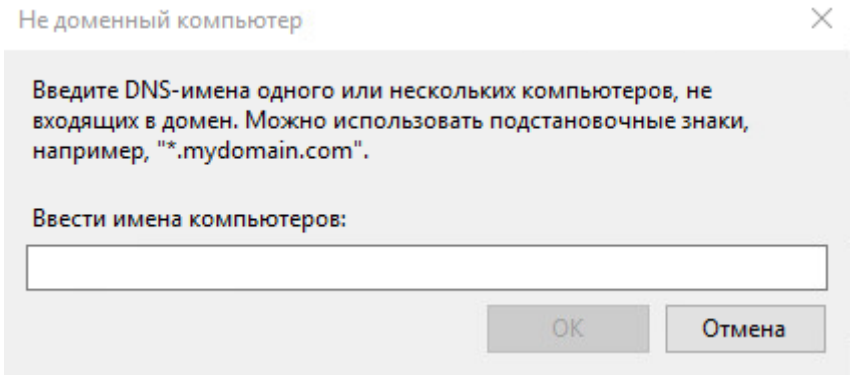

Рисунок 27

- Нажать «Добавить сертификаты».

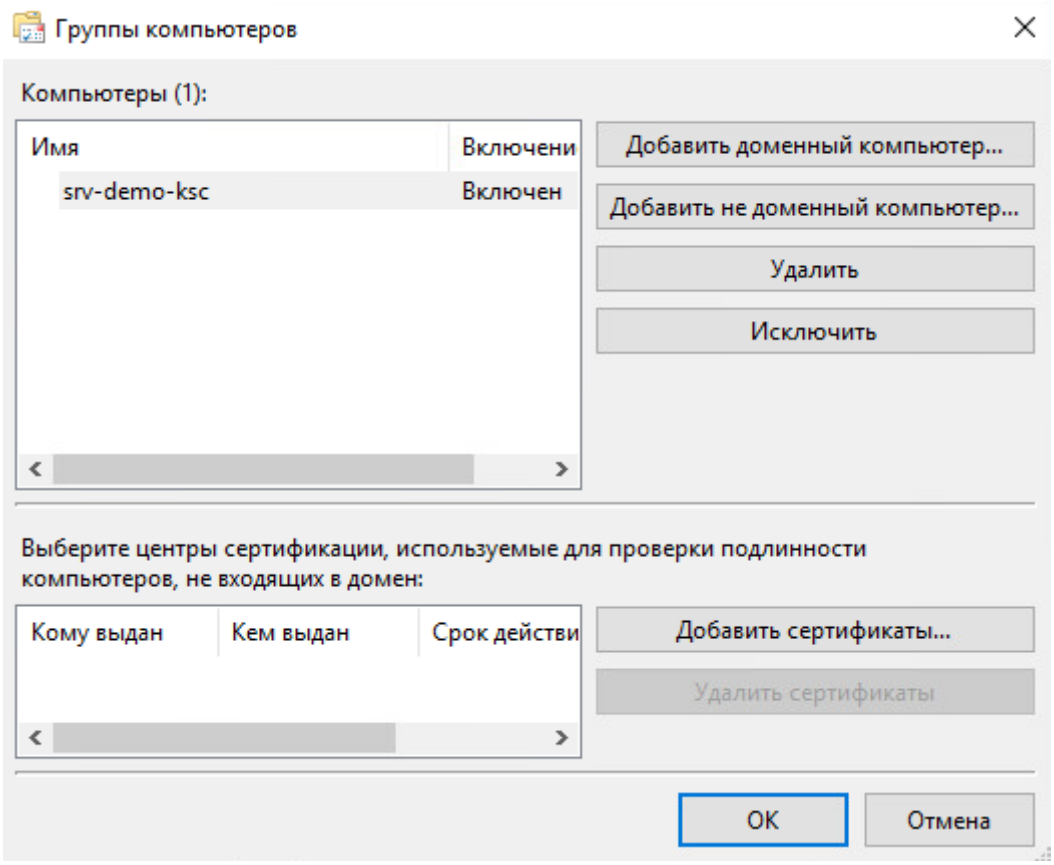

Рисунок 28

- В открывшемся окне выбрать сертификат центра сертификации, выпустившего сертификаты проверки подлинности клиента и сервера, и нажать «ОК».

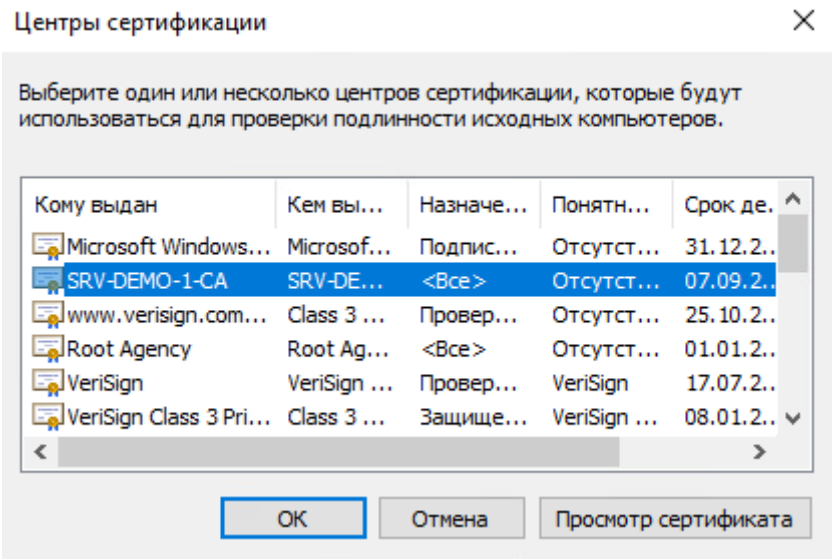

Рисунок 29

- Добавить при необходимости другие источники событий и сертификаты центров сертификации и нажать «ОК».

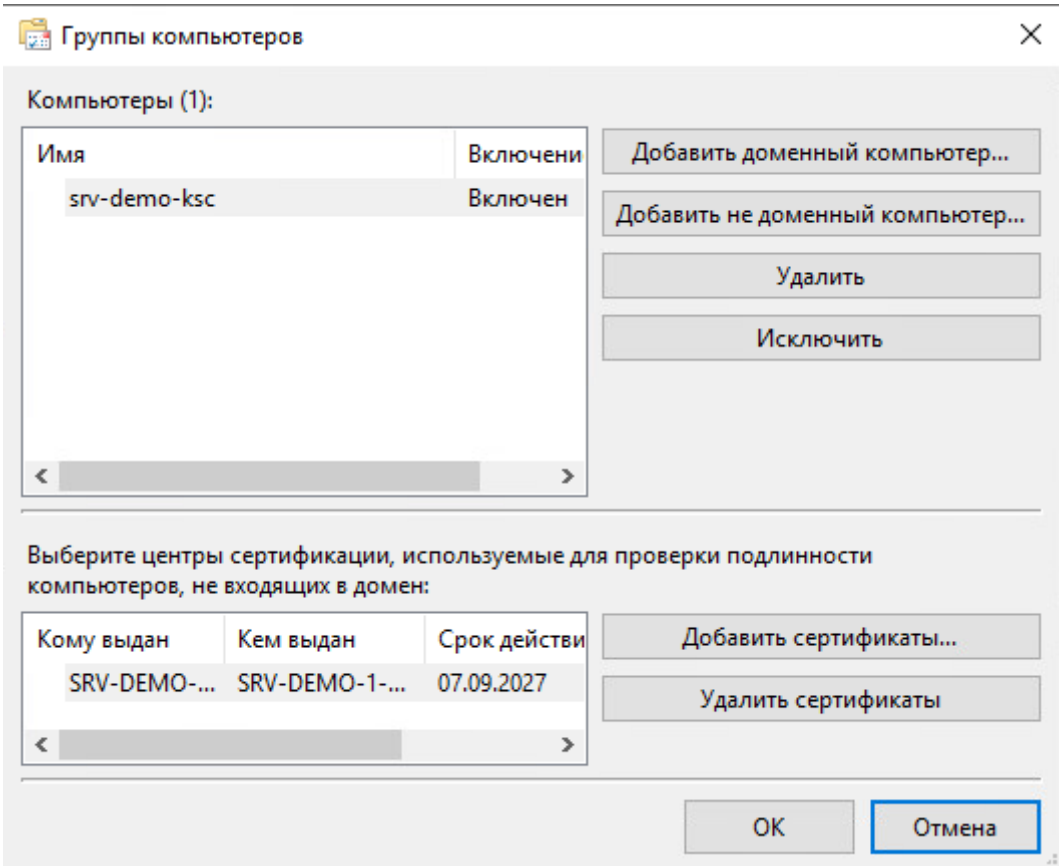

Рисунок 30

- Нажать «Выбрать события», настроить фильтр для запроса необходимых событий и нажать «ОК».

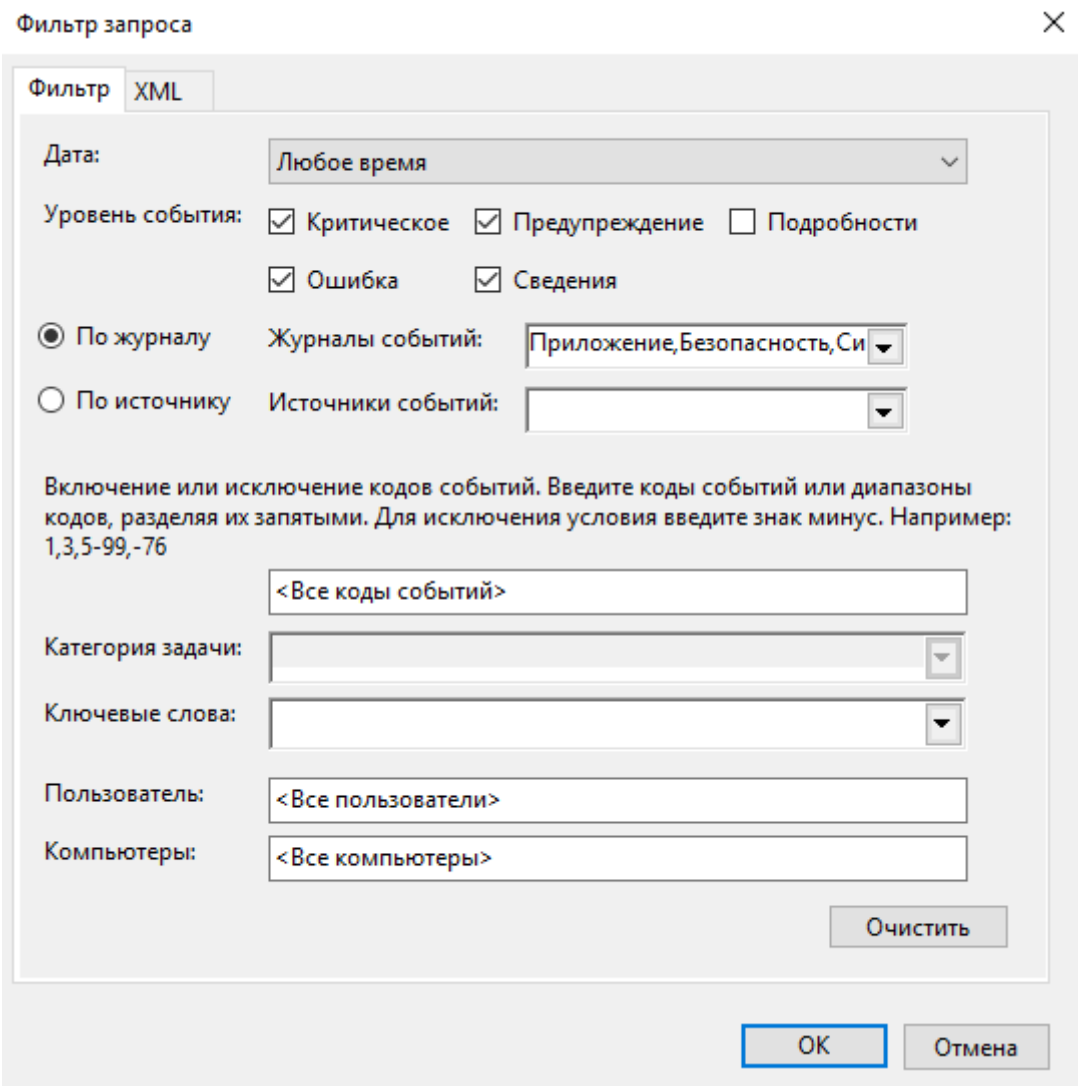

Рисунок 31

```
- Нажать «Дополнительно», выбрать протокол «HTTPS» и сохранить изменения.
```

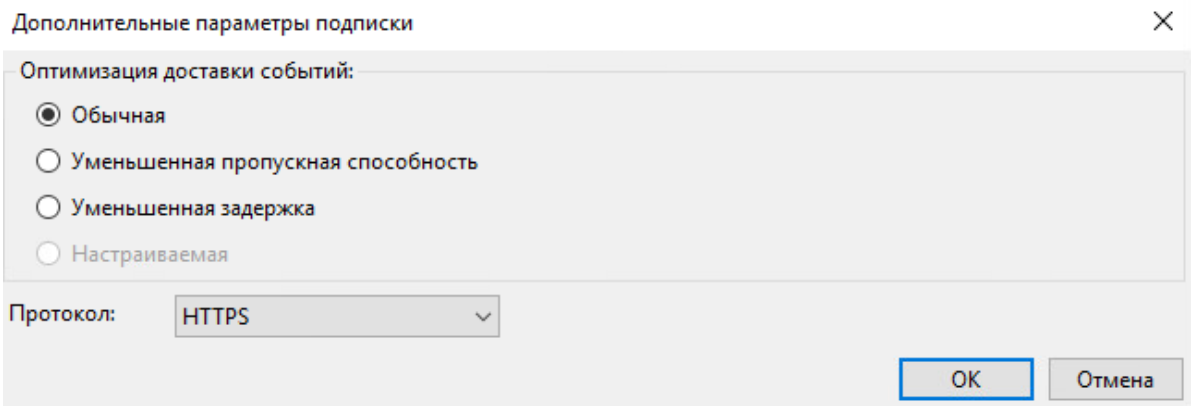

Рисунок 32

9. После создания подписки проверить её статус, нажав по ней правой кнопкой и выбрав «Состояние выполнения». Должен появиться источник событий с состоянием «Активный».

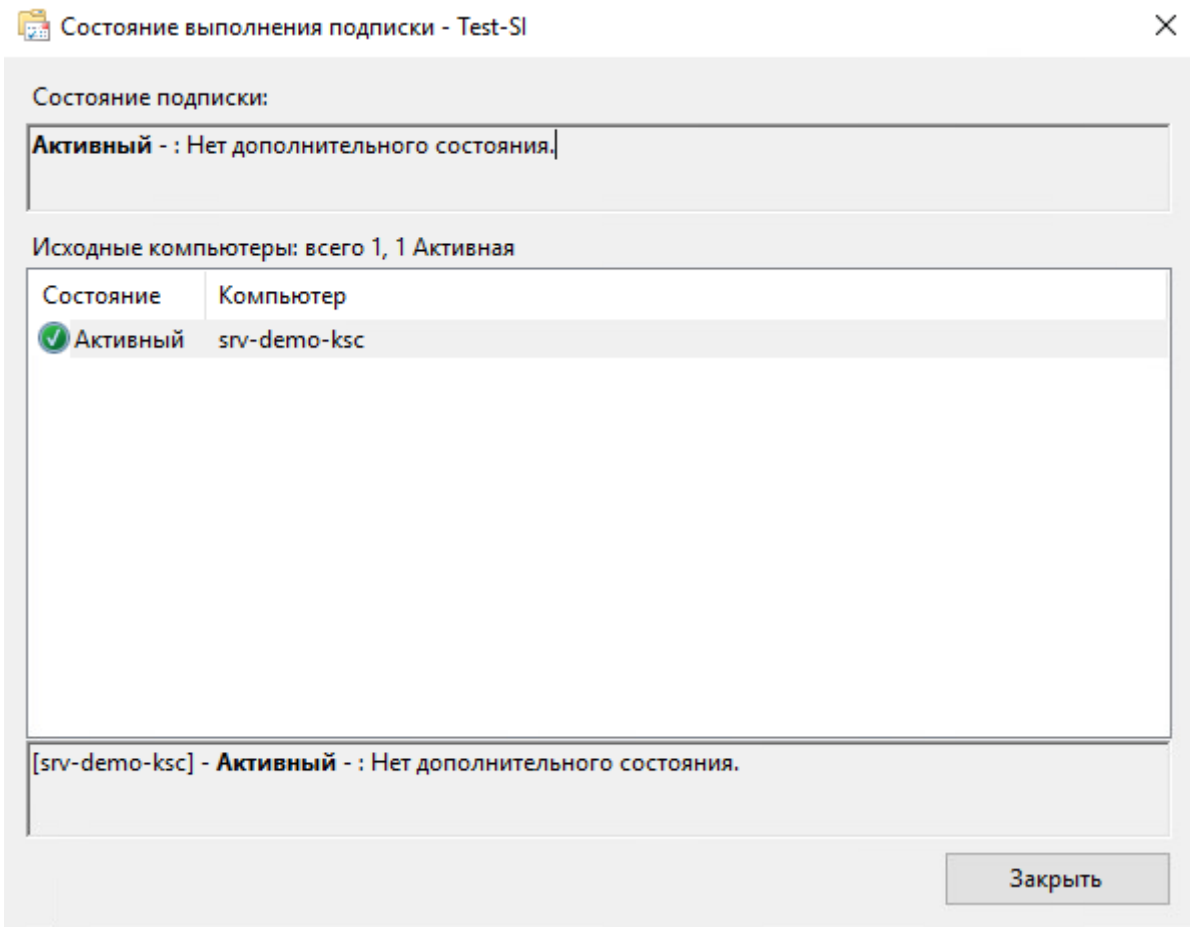

Рисунок 33

10. Проверить в конечном журнале сборщика поступление событий с источника. В случае непоступления событий просмотреть следующие журналы событий источника и сборщика на предмет наличия ошибок:

```
Microsoft-Windows-Windows Remote Management/Operational
Microsoft-Windows-Eventlog-ForwardingPlugin/Operational
Microsoft-Windows-CAPI2/Operational
Microsoft-Windows-EventCollector/Operational
```

4.4.2. Настройка пересылки событий на WEC в домене с использованием групповых политик

Данная инструкция применяется в случае, если серверы-источники и сервер WEC расположены в одном домене. Используется метод сбора, инициированный источником.

4.4.2.1. Настройка сервера WEC

Для настройки сервера WEC необходимо:

1. Открыть командную строку.
2. Запустить службу удаленного управления Windows: winrm qc –q.
3. Запустите службу сборщика событий Windows: wecutil qc /q.

Сервер-сборщик настроен.

4.4.2.2. Настройка подписки с типом «Инициировано источником»

Для настройки подписки на сервере WEC необходимо:

1. Открыть панель управления Windows.
2. Выбрать «Administrative Tools» → «Event Viewer».
3. В левой части окна выбрать «Subscriptions».
4. В главном меню выбрать «Action» → «Create Subscription...».
5. В открывшемся окне в поле «Subscription name» введите название подписки.

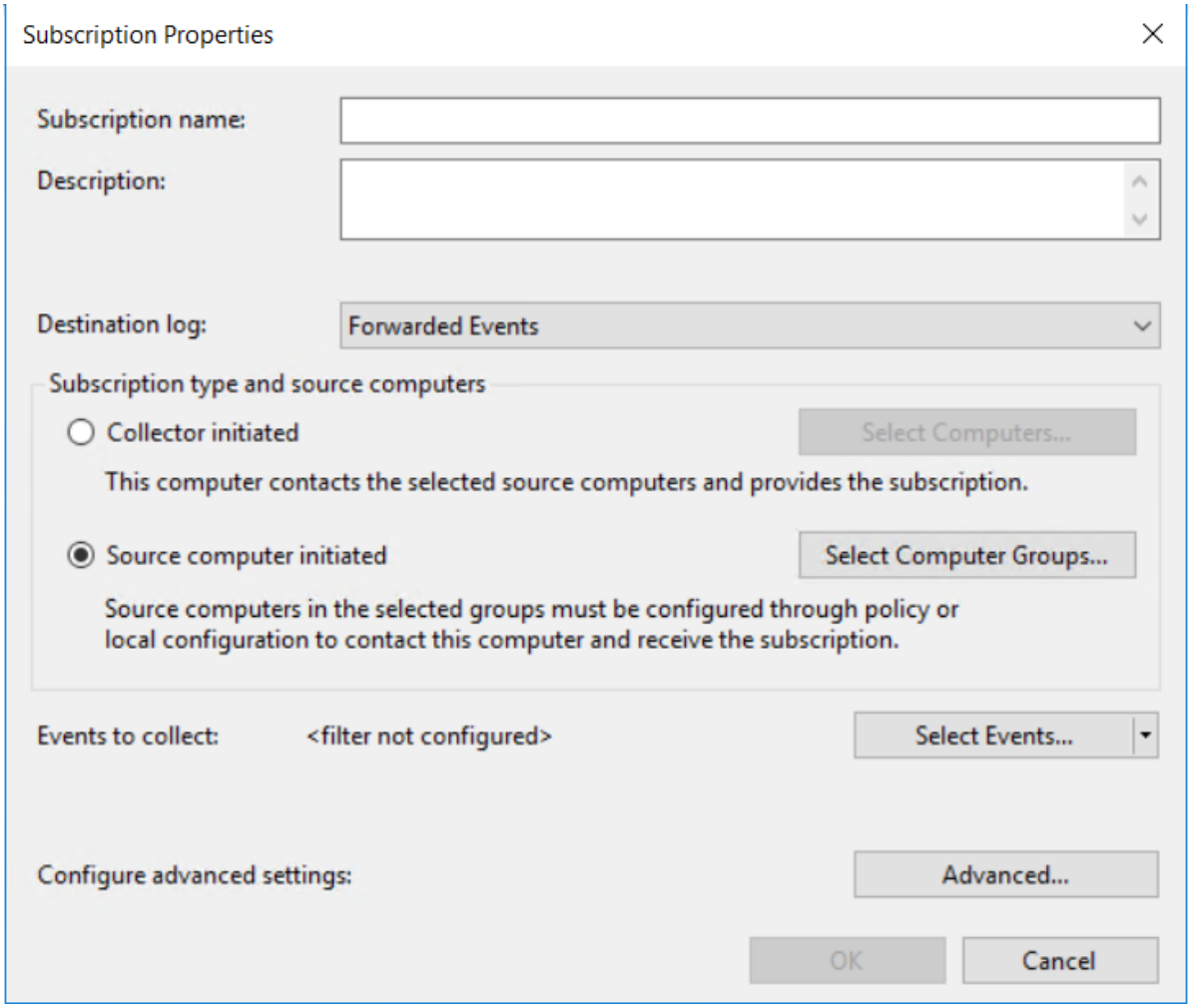

Рисунок 34

6. В раскрывающемся списке «Destination log» выбрать «Forwarded Events».
7. В блоке параметров «Subscription type and source computers» выбрать вариант «Source computer initiated».
8. Нажать кнопку «Select Computers...»
9. В открывшемся окне нажать кнопку «Add Domain Computers...».
10. В открывшемся окне в поле «Enter the object name to select» ввести имя компьютера и нажать кнопку «Check Names».
11. Нажать кнопку «OK».
12. В окне «Computer Groups» нажать кнопку «OK».
13. В окне «Subscription Properties» нажать кнопку «Select Events...».

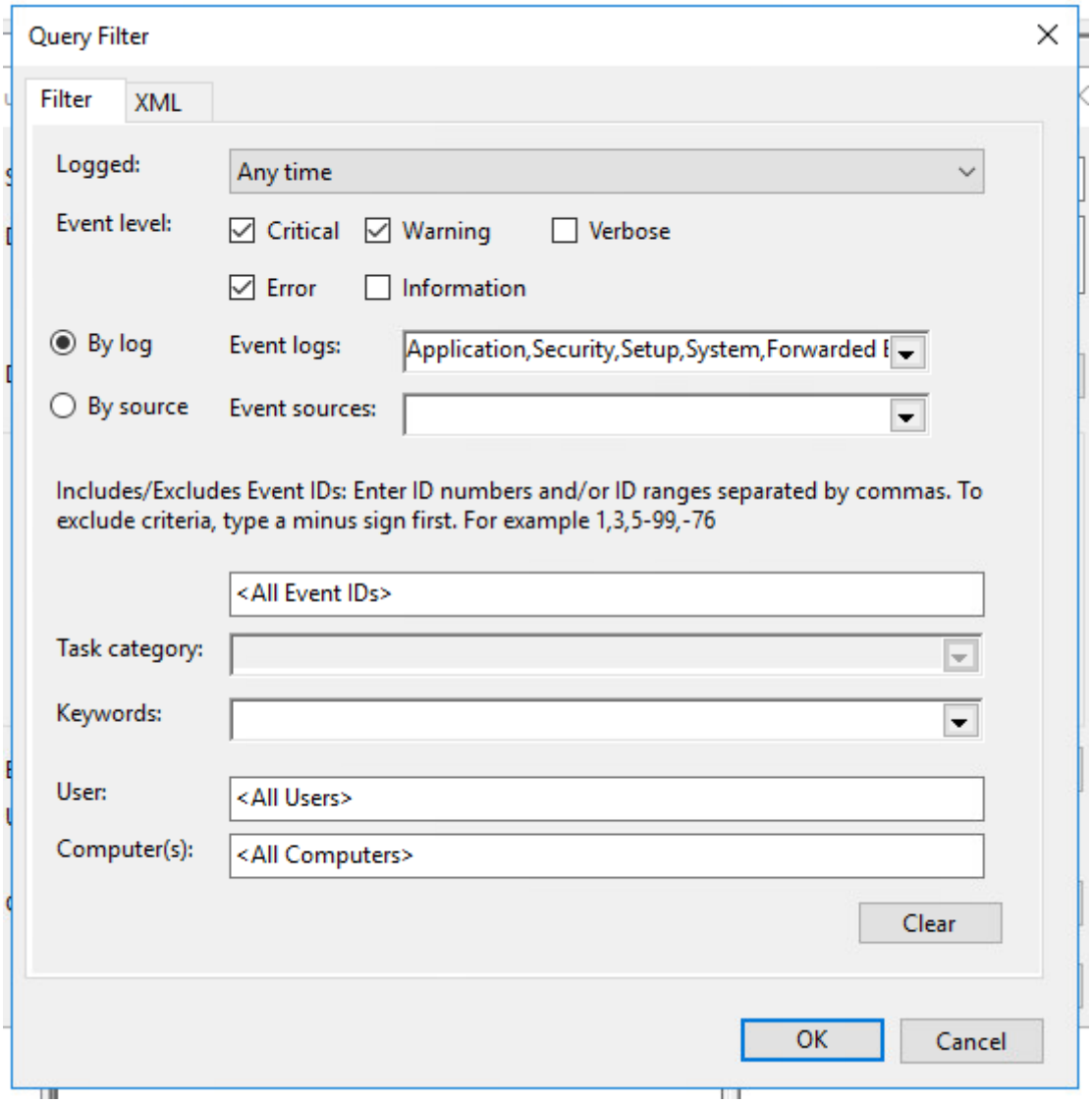

Рисунок 35

14. В открывшемся окне настроить фильтр событий и нажать кнопку «ОК».
15. В окне «Subscription Properties» нажать кнопку «Advanced...».

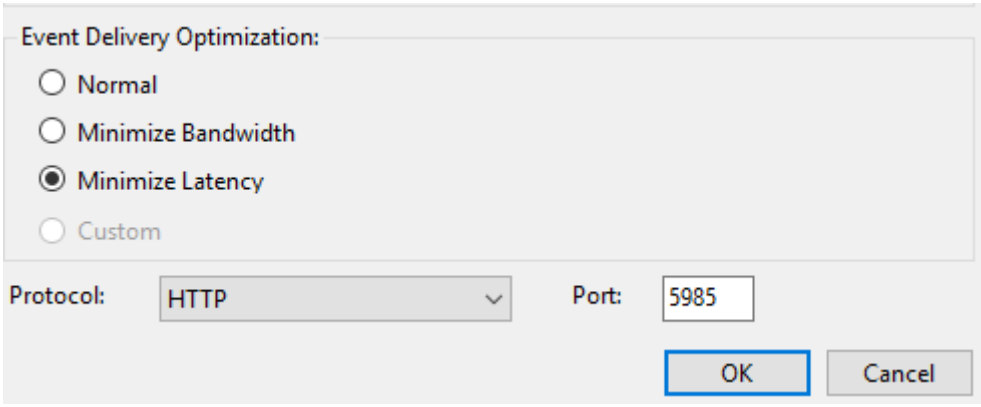

Рисунок 36

16. В открывшемся окне в блоке параметров «Event Delivery Optimization» выбрать вариант «Minimize Latency».
17. Нажать кнопку «ОК».
18. Снова нажать кнопку «ОК».

4.4.2.3. Настройка групповой политики для межсетевого экранирования

Для настройки групповой политики для межсетевого экранирования на контроллере домена необходимо:

1. Открыть панель управления Windows.
2. Выбрать «Administrative Tools» → «Group Policy Management».
3. В левой части окна выбрать групповую политику и нажать правую кнопку мыши.
4. В выпадающем списке нажать «Edit…».

   Откроется окно «Group Policy Management Editor».
5. В левой части окна выбрать узел «<Имя политики> Policy» → «Computer Configuration» → «Policies» → «Windows Settings» → «Security Settings» → «Windows Firewall with Advanced Security» → «Windows Firewall with Advanced Security» → «Inbound Rules».
6. В главном меню выбрать «Action» → «New Rule».

   Откроется мастер создания правила для нового входящего подключения.

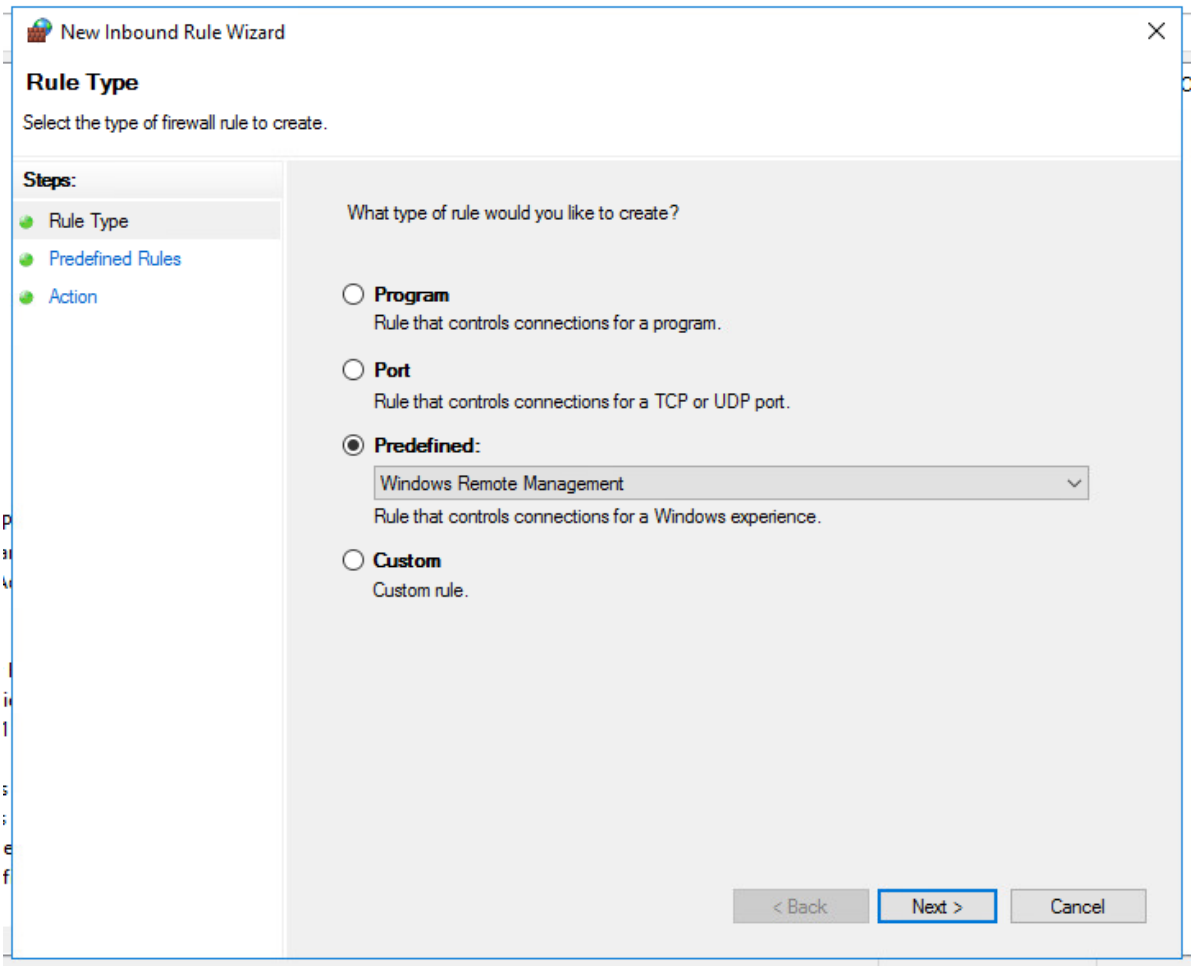

Рисунок 37

7. На первом шаге выбрать вариант «Predefined».
8. В раскрывающемся списке выбрать «Windows Remote Management» и нажать кнопку «Next».

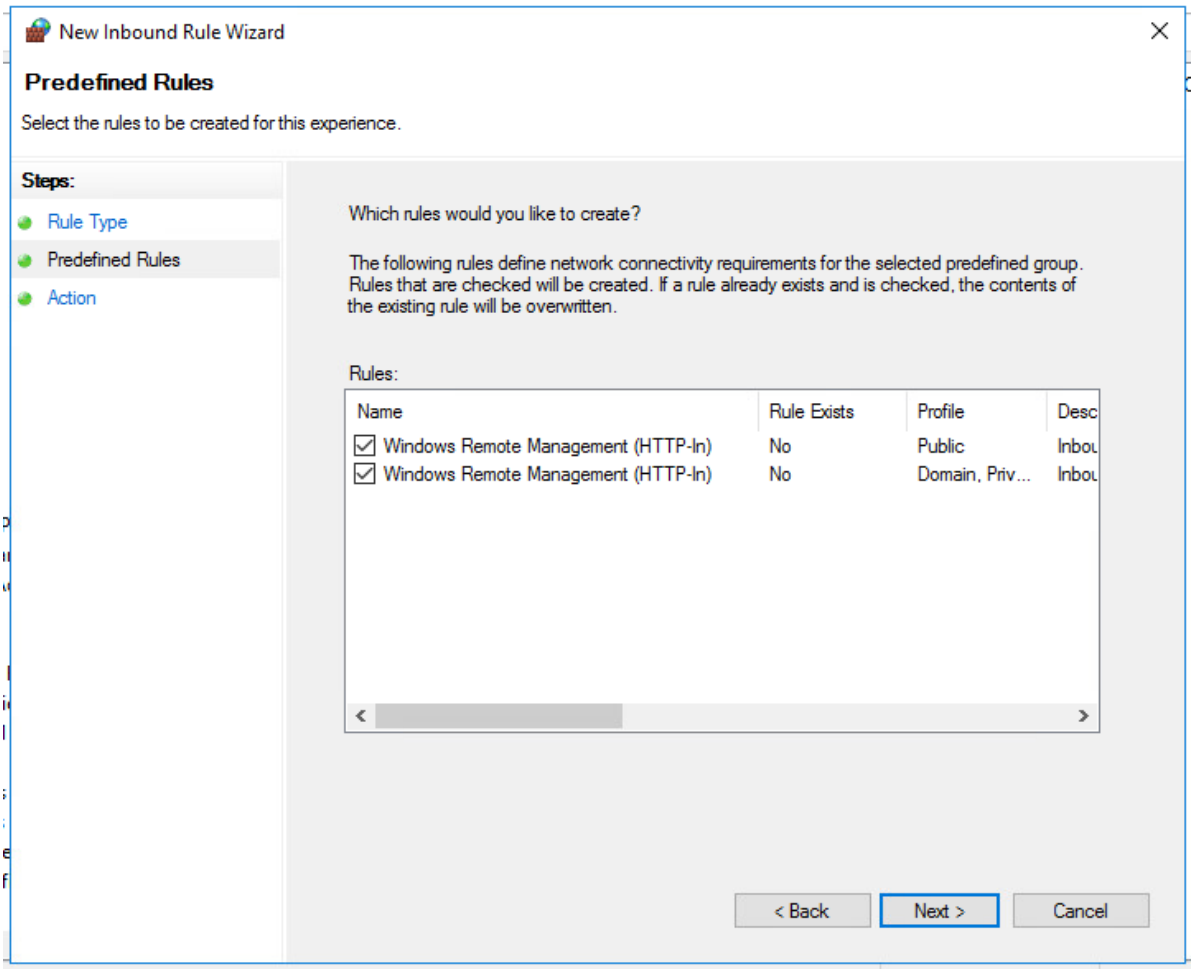

Рисунок 38

9. На следующем шаге нажать кнопку «Next».

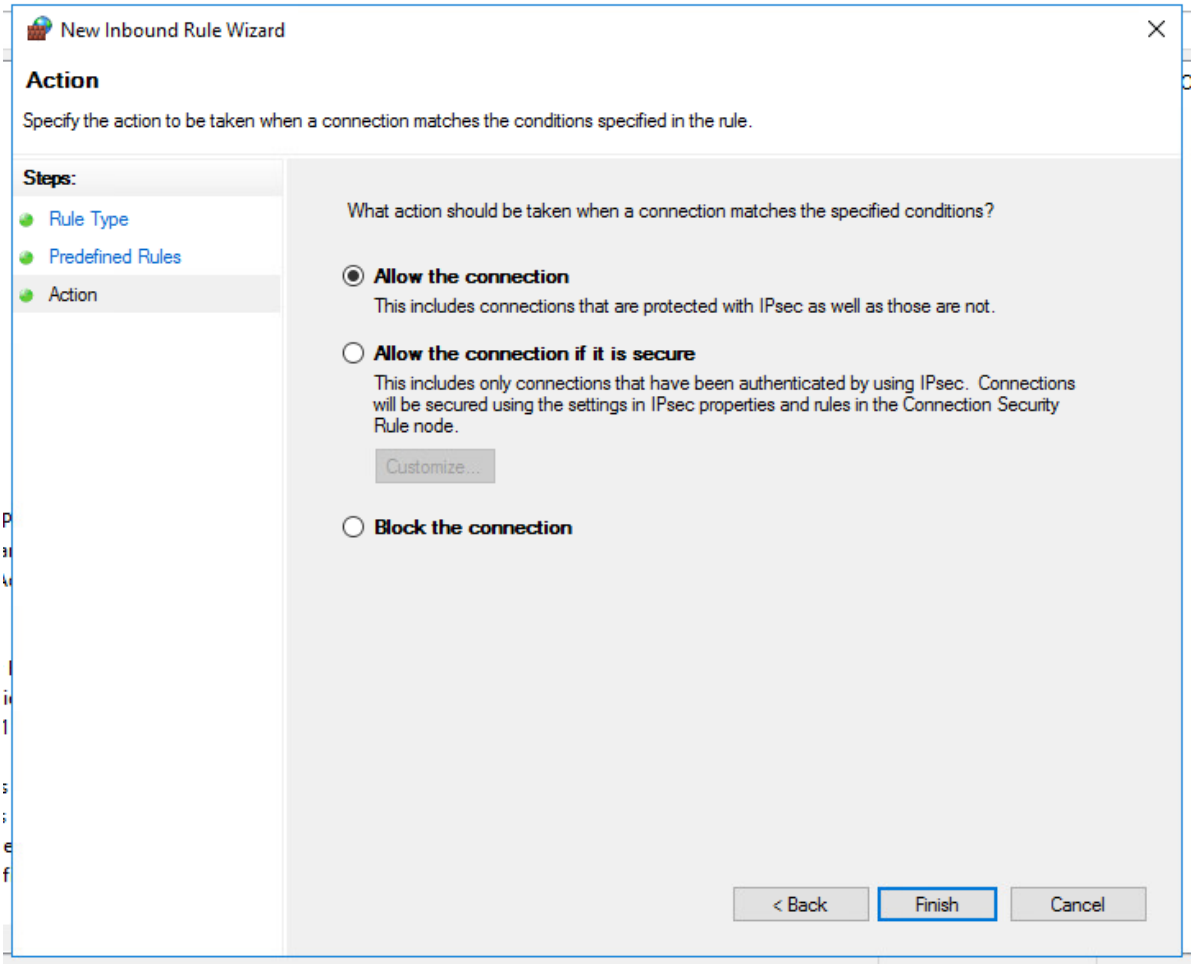

Рисунок 39

10. На следующем шаге выбрать вариант «Allow the connection» и нажать кнопку «Finish».
11. В левой части окна выбрать узел «<Имя политики> Policy» → «Computer Configuration» → «Policies» → «Windows Settings» → «Security Settings» → «System Services».
12. В правой части окна выбрать «Windows Firewall».

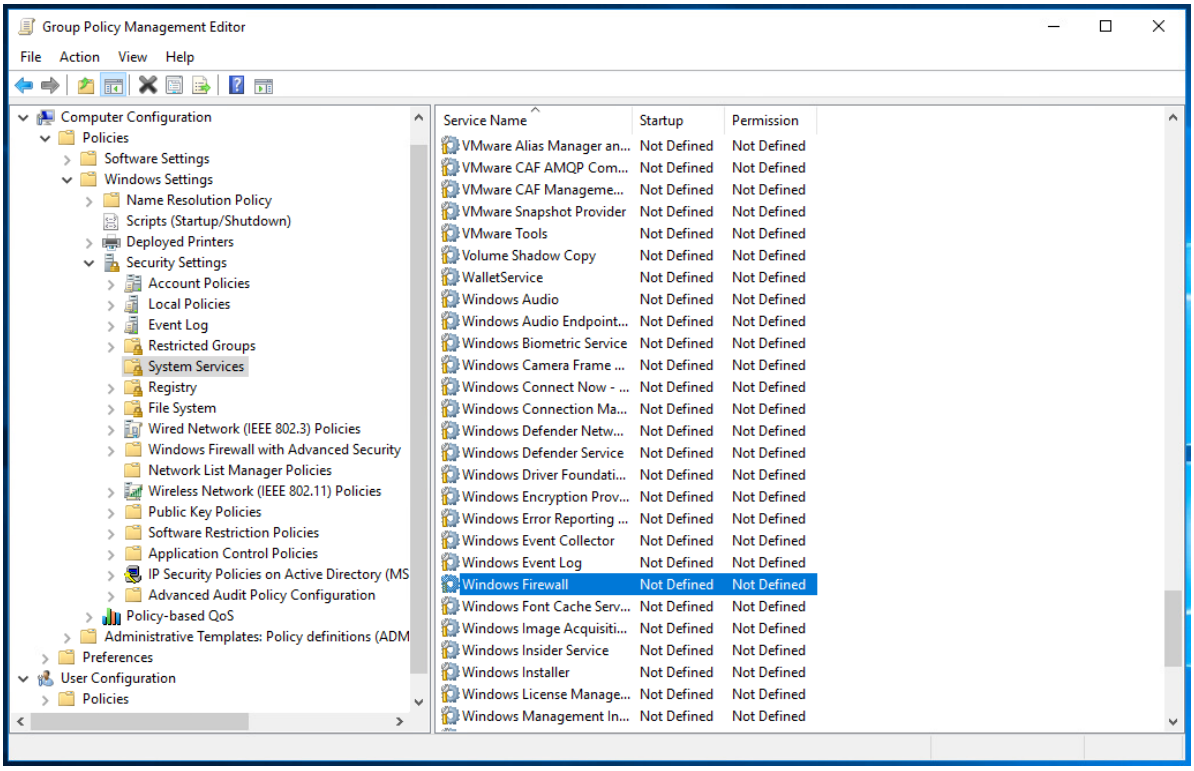

Рисунок 40

13. В главном меню выбрать «Action» → «Properties».
14. В открывшемся окне установить флажок «Define this policy setting».
15. Выбрать автоматический режим запуска службы.

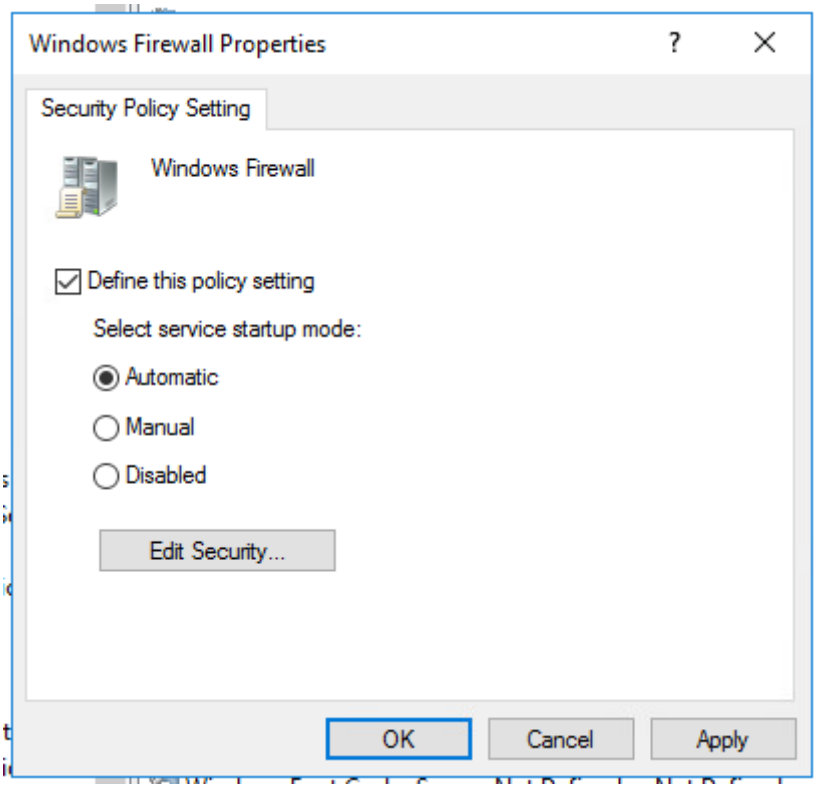

Рисунок 41

16. Нажать кнопку «OK».

Групповая политика настроена.

4.4.2.4. Настройка групповой политики для учетной записи Сервера Сборщика

Для настройки групповой политики для учетной записи «Network Service» на контроллере домена необходимо:

1. Открыть панель управления Windows.
2. Выбрать «Administrative Tools» → «Group Policy Management»
3. В левой части окна выбрать групповую политику и нажать правую кнопку мыши.
4. В выпадающем списке нажать «Edit...».

   Откроется окно «Group Policy Management Editor».
5. В левой части окна в контекстном меню узла «<Имя политики> Policy» → «Computer Configuration» → «Policies» → «Windows Settings» → «Security Settings» → «Restricted Groups» выбрать узел «Add Group».
6. В открывшемся окне в поле «Group» ввести «Event Log Readers» и нажать кнопку «OK».
7. В открывшемся окне справа от поля «Members of this groupы» нажать кнопку «Add...».

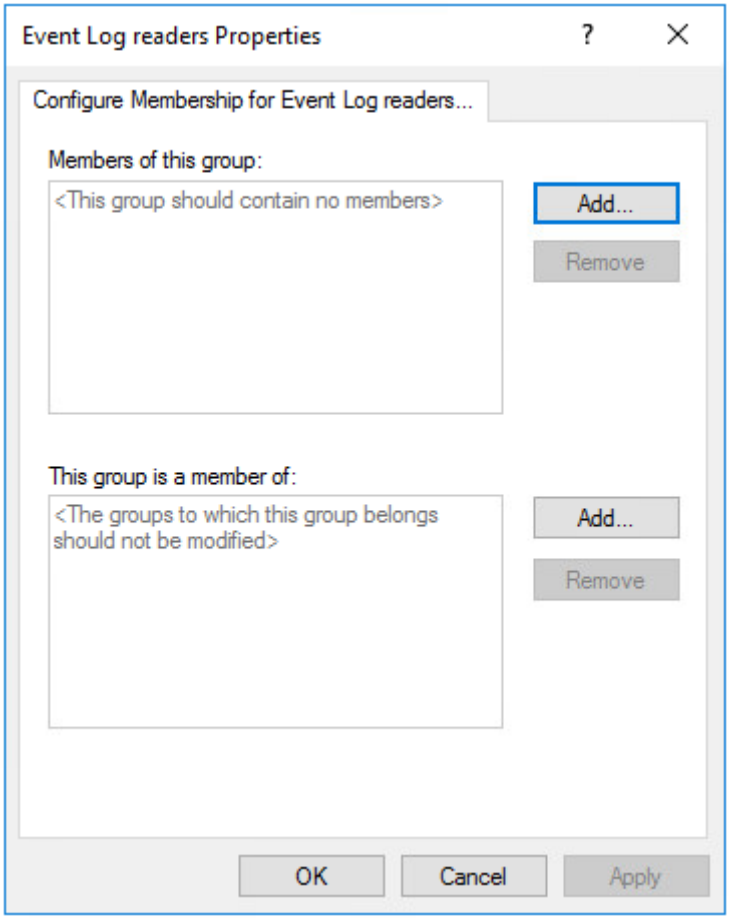

Рисунок 42

8. В открывшемся окне в поле «Members of this group» ввести «Network Service» и нажать кнопку «OK».
9. Нажать кнопку «OK».

Групповая политика настроена.

4.4.2.5. Настройка групповой политики для сервера WEC

Для настройки групповой политики для сервера WEC на контроллере домена необходимо:

1. Открыть панель управления Windows.
2. Выбрать «Administrative Tools» → «Group Policy Management»
3. В левой части окна выбрать групповую политику и нажать правую кнопку мыши.
4. В выпадающем списке нажать «Edit...».

   Откроется окно «Group Policy Management Editor».
5. В левой части окна выбрать узел «<Имя политики> Policy» → «Computer Configuration» → «Policies» → «Administrative Templates...» → «Windows Components» → «Event Forwarding».
6. Выбрать параметр «Configure target Subscription Manager».

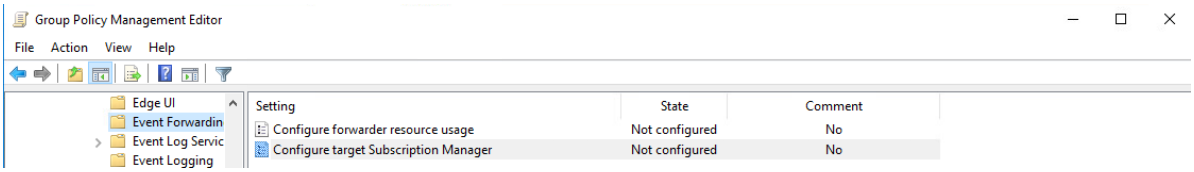

Рисунок 43

7. В главном меню выбрать «Action» → «Edit».

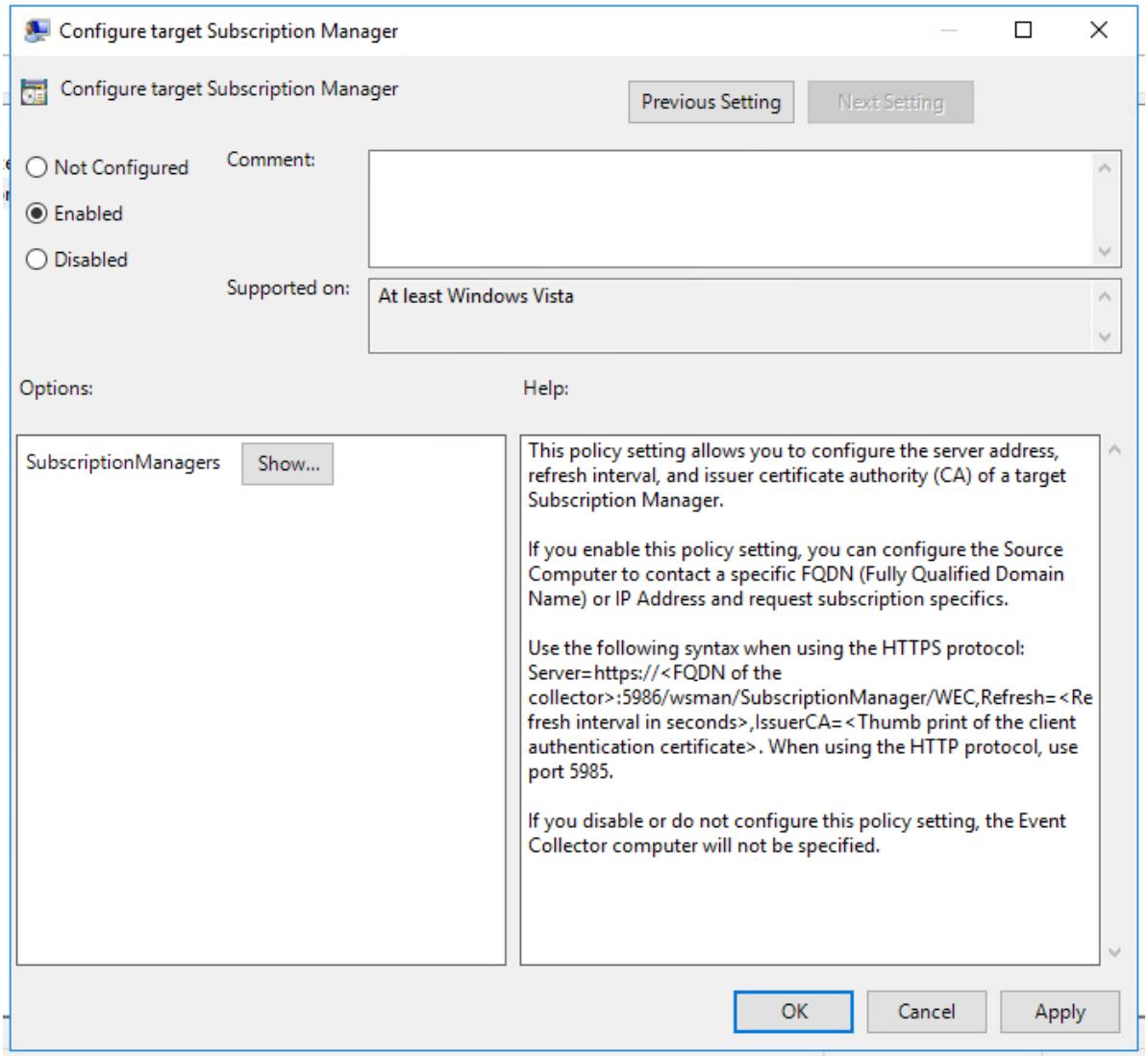

Рисунок 44

8. В открывшемся окне выбрать вариант «Enabled».
9. Нажать кнопку «Show».
10. В открывшемся окне ввести имя сервера WEC в формате FQDN, в зависимости от используемого протокола:
    - если используется протокол HTTP:

      `Server=http://<Имя сервера WEC>:5985/wsman/SubscriptionManager/WEC`
    - если используется протокол HTTPS:

      `Server=https://<Имя сервера WEC>:5986/wsman/SubscriptionManager/WEC`
11. Нажать кнопку «OK».
12. Снова нажать кнопку «OK».
13. Выбрать параметр «Configure forwarder resource usage».
14. В главном меню выбрать «Action» → «Edit».

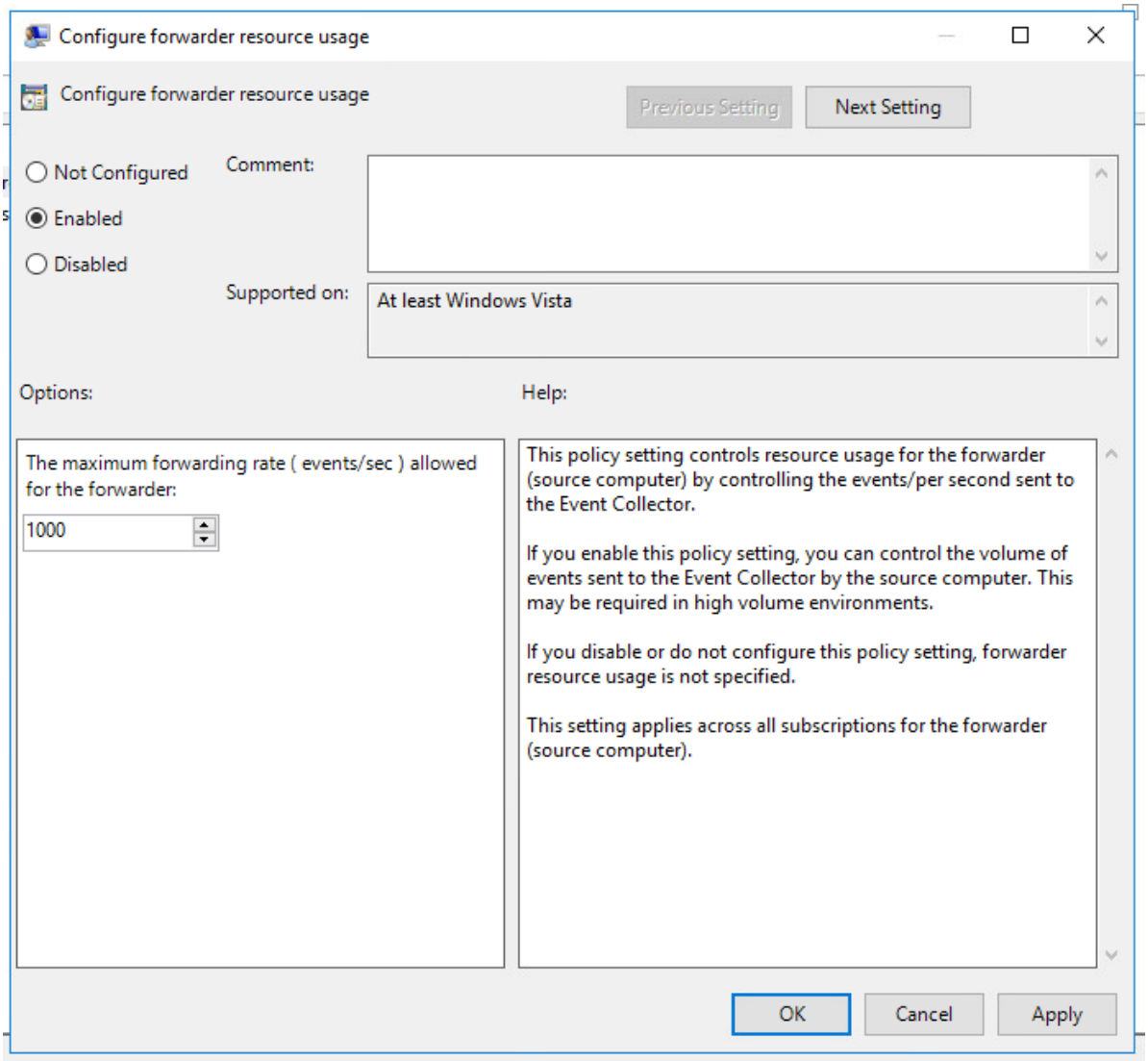

Рисунок 45

15. В открывшемся окне выбрать вариант «Включено».
16. В поле «The maximum forwarding rate (events/sec) allowed for the forwarder» ввести максимальное число событий, передаваемых за секунду.

    Среднее число событий, сохраняемое за сутки в журнале безопасности ОС (Security), можно узнать выполнив в Windows PowerShell команду:

```
(Get-WinEvent -FilterXML "<QueryList><Query><Select Path='Security'>*
[System[TimeCreated[timediff() @SystemTime )<=86400000]]]</Select></Query>
</QueryList>").count.count`
```

17. Нажать кнопку «OK».
18. В левой части окна выбрать узел «<Имя политики> Policy» → «Computer Configuration» → «Policies» → «Windows Settings» → «Security Settings» → «System Services».
19. Выбрать «Windows Remote Management (WS-Management)».

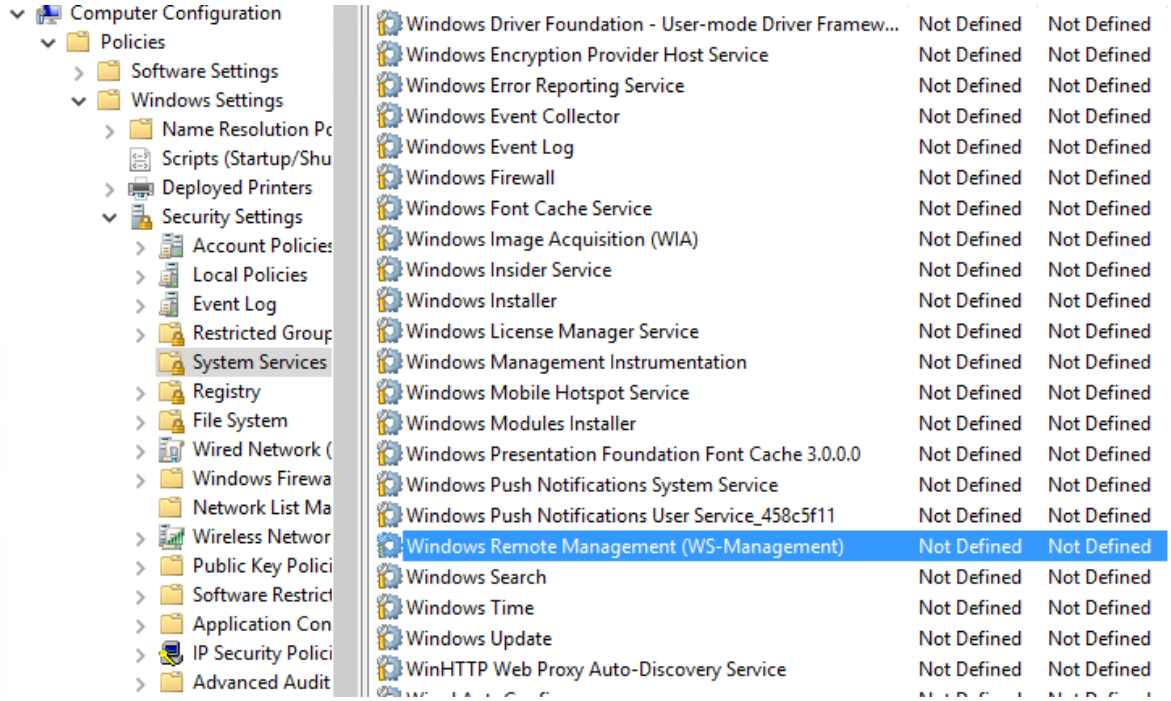

Рисунок 46

20. В главном меню выбрать «Action» → «Properties».
21. В открывшемся окне установить флажок «Define this policy setting».
22. Выбрать автоматический режим запуска службы.

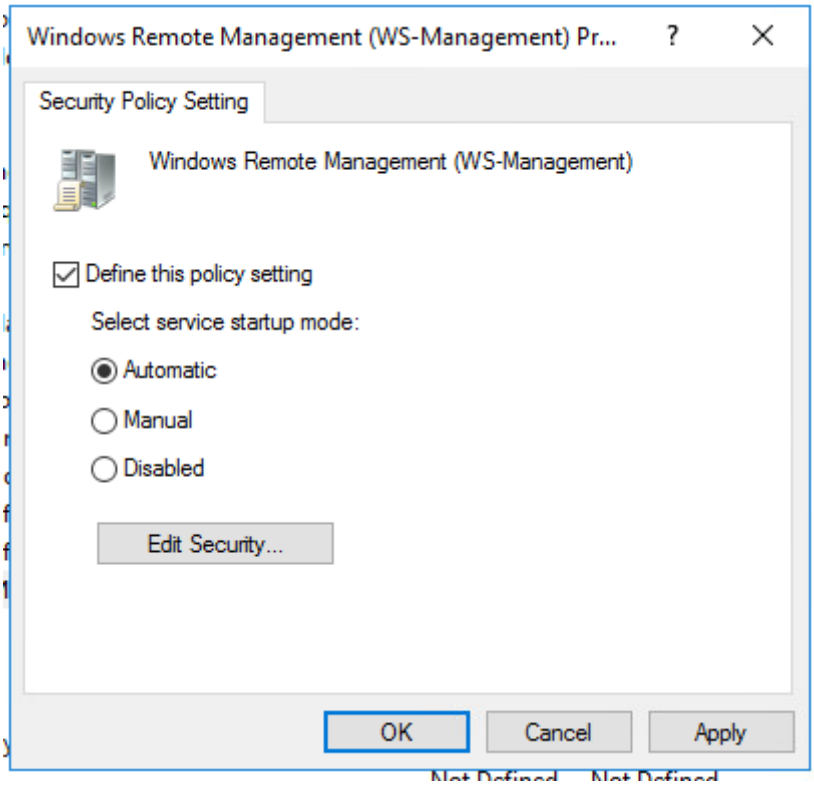

Рисунок 47

23. Нажать кнопку «ОК».

Групповая политика настроена.

4.4.2.6. Подготовка форвардеров к отправке

Для подготовки необходимо:

1. Открыть командную строку.
2. Выполнить в интерфейсе командной строки команду gpupdate /force (для обновления групповых политик на форвардерах).
3. Перезапуск службы «Служба удаленного управления Windows (WS-Management)» («Windows Remote Management (WS-Management)».

4.4.2.7. Создание групповой политики.

Для настройки групповой политики на контроллере домена необходимо:

1. Открыть панель управления Windows.
2. Выбрать «Administrative Tools» → «Group Policy Management».
3. В левой части окна выбрать объект, для которого нужно создать групповую политику.

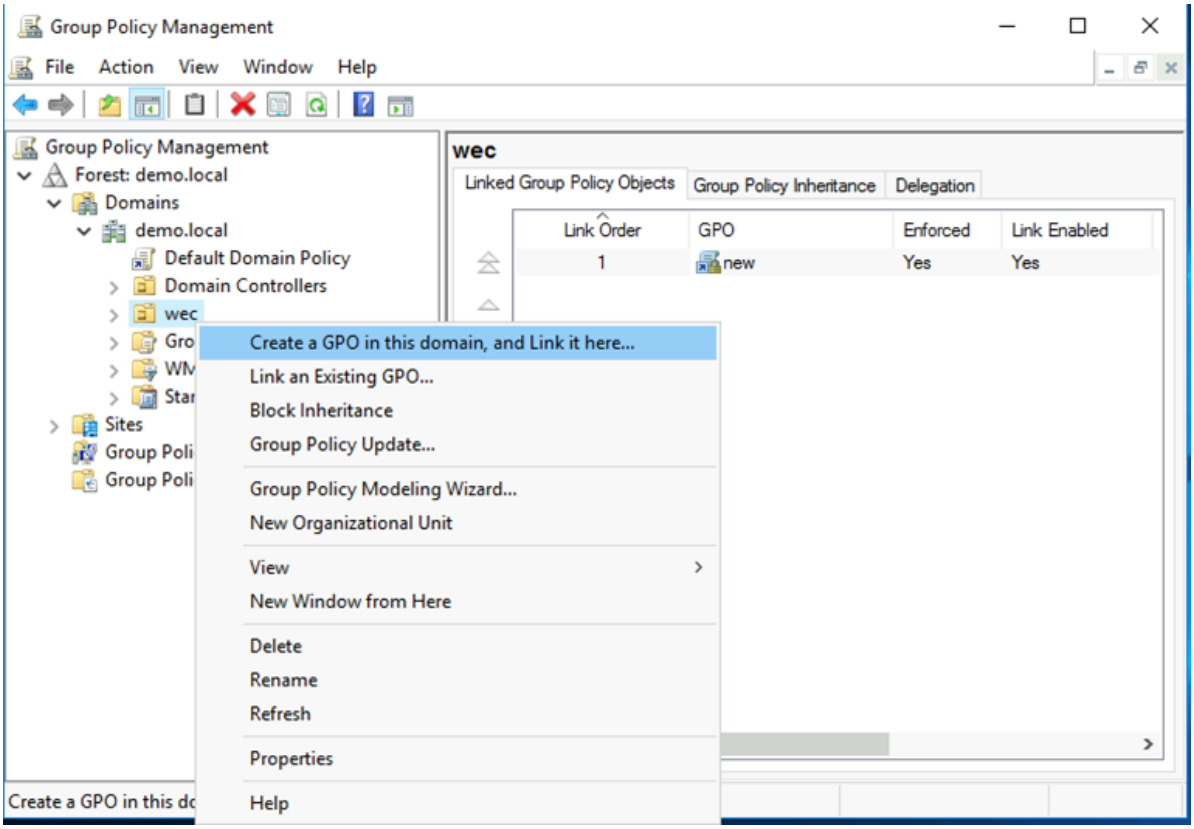

Рисунок 48

4. В главном меню выбрать «Action» → «Create a GPO in this domain, and Link it here...».
5. В открывшемся окне в поле «Имя» ввести имя групповой политики.
6. Нажать кнопку «ОК».

Групповая политика создана.

4.4.2.8. Решение возникновения возможных проблем

В случае возникновения проблем с настройкой WEC – журналы с ошибками можно найти в Event Viewer, в разделе «Applications and Services Logs» → «Microsoft» → «Windows» → «Eventlog-ForwardingPlugin» → «Operational».

Если в журнале следующая ошибка:

```
The forwarder is having a problem communicating with subscription manager at
address http://<FQDN_of_WEC_Server>:5985/wsman/SubscriptionManager/WEC. Error
code is 2150859027 and Error Message is <f:WSManFault xmlns:f="
http://schemas.microsoft.com/wbem/wsman/1/wsmanfault " Code="2150859027"
Machine="demo-server2012.demo.local"><f:Message>The WinRM client sent a request
to an HTTP server and got a response saying the requested HTTP URL was not
available. This is usually returned by a HTTP server that does not support the
WS-Management protocol. </f:Message></f:WSManFault>.
```

На WEC сервере необходимо выполнить следующие команды:

```
netsh http delete urlacl url=http://+:5985/wsman/

netsh http add urlacl url=http://+:5985/wsman/ sddl=D:(A;;GX;;;S-1-5-80-
569256582-2953403351-2909559716-1301513147-412116970)(A;;GX;;;S-1-5-80-
4059739203-877974739-1245631912-527174227-2996563517)
```

После чего перезапустить службу «Служба удаленного управления Windows (WS-Management)» («Windows Remote Management (WS-Management)» на сервере WEC и компьютере-форвардере.

4.5. IBM AIX {#aix}

Для настройки источника IBM AIX Server на отправку событий в **Платформу Радар** выполните следующие шаги:

1. Подключитесь к вашему устройству под пользователем root.
2. Откройте файл `/etc/syslog.conf`
3. Чтобы перенаправить журналы аутентификации – добавьте в файл следующую строку: `auth.info @@<IP_address-лог-коллектора>`/

   > Запись должна разделять `auth.info` и указанный IP-адрес.

   Например:

```
#####
begin
/etc/syslog.conf
mail.debug
/var/adm/maillogmail.none
/var/adm/maillogauth.notice
/var/adm/authloglpr.debug
/var/adm/lpd-errskern.debug
/var/adm/messages*.emerg;*.alert;*.crit;*.warning;*.err;*.notice;*.info
/var/adm/messagesauth.info @@IP_address-лог-коллектора >
#####
end
/etc/syslog.conf
```

4. Сохраните и закройте файл.
5. Перезапустите службу syslog командой:

   `refresh -s syslogd`
6. На машине с лог-коллектором добавьте изменения в файл конфигурации для сбора событий от источника и отправки их в **Платформу Радар**:
   - добавить Компонент сбора событий

     ```
     udp_input_ibm_aix: & udp_input_ibm_aix
         id: "udp_input_ibm_aix"
       host: "0.0.0.0"
         port: 514
         sock_buf_size: 0
       format: "json"
         log_level: "INFO"
     ```

   - добавить Компонент отправки событий

     ```
     tcp_output_ibm_aix: & tcp_output_ibm_aix
     id: "tcp_output_ibm_aix"
     target_host: "<ip адрес  Платформы Радар/или балансера>"
     port: 2641
     sock_buf_size: 0
     log_level: "INFO"
     ```

   - указать добавленные компоненты сбора и отправки в разделы collectors и senders соответственно

     ```
     collectors:
         udp_receiver:
         - <<: *udp_input_ibm_aix

     senders:
       port: 48002
         tcp:
         - <<: *tcp_output_ibm_aix
     ```

- добавить маршрут взаимодействия между компонентами сбора событий и компонентами отправки событий

```
route_1_ibm_aix: &route_1_ibm_aix
    collector_id:
    - "udp_input_ibm_aix"
    sender_id:
    - "tcp_output_ibm_aix"
```

- включить маршрут в разделе конфигурационного файла routers

```
routers:
  - <<: *route_1_ibm_aix
```

7. Перезапустите службу лог-коллектора.
8. Включить источник IBM-AIX в **Платформе Радар** и нажмите кнопку «Синхронизировать».
9. Проверьте поступающие события в **Платформе Радар** в разделе «Просмотр событий».

4.6. Unix/Linux {#linux}

4.6.1. Настройка источника

Для настройки пересылки событий необходимо настроить RSYSLOG. Для этого перейдите в настройки конфигурационного файла rsyslog командой:

`nano /etc/rsyslog.conf`

В конфигурационной файле добавьте следующую строку:

`auth,authpriv.* @<адрес лог  -коллектора>:<порт>`

После внесения изменений в конфигурационный файл rsyslog перезагрузите службу командой:

`service rsyslog restart`

> Порт необходимо выбирать в соответствии с типом операционной системы. Список поддерживаемых Linux/Unix ОС и распределение по портам представлены в таблице 2.

Таблица 2 -- Распределение поддерживаемых ОС Unix/Linux по портам

Порт	Тип ОС	ОС
2631	Unix	Solaris
2641	IBM	AIX
2651	RHEL	Linux
2661	CentOS	Linux
2671	Debian	Linux
2681	Ubuntu	Linux
2686	Astra	Linux

Порт	Тип ОС	ОС
2691	SUSE	Linux
2711	Fedora	Linux
2721	Oracle	Linux

4.6.2. Включение источника в Платформе Радар

1. Зайдите в веб-консоль **Платформы Радар**, перейдите в раздел «Источники», «Управление источниками».
2. Найдите в списке доступных источников источник начинающийся с «Linux-» и включите тот, который необходимо подключите
3. Нажмите на кнопку «Синхронизировать».

4.6.3. Настройка коллектора событий

1. В конфигурационный файл лог-коллектора (config.yaml) добавьте input компонента UDP.

   Основные параметры, которые необходимо указать:

   - `host: <ip адрес лог-коллектора>` - адрес, на котором запущен коллектор
   - `port: <порт для приема соединений>` - порт, на который будут приниматься события
2. После настройки компонента сбора событий (input) настройте компонент отправки событий (output).

   В качестве компонента отправки событий для данного источника предусмотрено использование отправки по протоколу UDP.
   Основные параметры, которые необходимо указать:

   - `target_host: <ip адрес или имя удаленного узла>` - адрес **Платформы Радар**
   - `port: <порт>` - стандартный порт для данного источника
3. Далее включите компоненты сбора (collectors) и отправки (senders).

   Основные параметры, которые нужно указать при включении компонентов сбора:

   ```
   collectors:
     udp_reciever:
       - <<: *<"id компонента сбора"> (ID компонента сбора, который указывали при объявлении компонента сбора)
   ```

   Основные параметры, которые нужно указать при включении компонентов отправки:

   ```
   senders:
     udp:
       - <<: *<"id компонента отправки"> (ID компонента отправки, который указывали при объявлении компонента отправки)
   ```

4. После этого настройте маршрутизацию событий.

   Основные параметры, которые нужно указать при настройке маршрута:

```
route_1: &route_1
  collector_id:
    - <"id компонента сбора">
  sender_id:
    - <"id компонента отправки">
```

5. Включите маршрут в разделе routers:

```
routers:
  - <<: *<название маршрута> (например - <<: *route_1)
```

5. Решения Network Security

5.1. Межсетевой экран Cisco ASA {#ciscoasa}

5.1.1. Настройка источника

1. Подключиться к консоли устройства;
2. Для включения журналирования и экспорта событий с устройства, введите команды:

`(config)# logging enable`

`(config)# logging host <имя интерфейса> <IP-адрес коллектора>`

`(config)# logging trap <уровень логирования> (указать один из уровней важности событий: alerts, critical, debugging, emergencies, errors, informational, notifications, warnings)`

`(config)# logging console <уровень логирования> (указать один из уровней важности событий: alerts, critical, debugging, emergencies, errors, informational, notifications, warnings)`

`(config)# logging asdm <уровень логирования> (указать один из уровней важности событий: alerts, critical, debugging, emergencies, errors, informational, notifications, warnings)`

`(config)# logging device-id ipaddress <id устройства>`

`(config)# logging timestamp`

5.1.2. Включение источника в Платформе Радар

Для информации! Включение источника в **Платформе Радар** представлено в разделе *Управление источниками в **Платформе Радар**, Включение/выключение поддерживаемых источников и их синхронизация*

1. Зайти в веб-консоль **Платформы Радар**, перейти в раздел «Источники» - «Управление источниками»;
2. Найти в списке доступных источников (Cisco-ASA) и включить его;
3. Кликнуть на кнопку «Синхронизировать».

5.1.3. Настройка коллектора событий

Для информации! Пример конфигурационного файла с настройкой данного источника представлен в разделе *Пример конфигурационного файла лог-коллектора*. Более подробная информация о настройках лог-коллектора представлена в разделе *Руководство по настройке лог-коллектора*

1. В конфигурационный файл лог-коллектора (config.yaml) необходимо добавить input компонента UDP.

   Пример настройки по умолчанию можно найти в документации: *Руководство по настройке лог-коллектора, Компонент UDP*

   Основные параметры, которые необходимо указать:

   `host: "<ip адрес лог-коллектора>" (адрес, на котором запущен коллектор)`

   `port: <порт для приема соединений> (порт, на который будут приниматься события, если при настройке источника оставили стандартный - 2520)`

2. После настройки компонента сбора событий (input) необходимо настроить компонент отправки событий (output).

   В качестве компонента отправки событий для данного источника предусмотрено использование отправки по протоколу TCP.

   Пример настройки по умолчанию можно найти в документации: *Руководство по настройке лог-коллектора, Компонент отправки событий по протоколу TCP*

   Основные параметры, которые необходимо указать:

   `target_host: <"ip адрес или имя удаленного узла"> (адрес Платформы Радар)`

   `port: <"порт"> (стандартный порт для данного источника 2520)`

3. Далее необходимо включить компоненты сбора (collectors) и отправки (senders).

   Пример настройки по умолчанию можно найти в документации: *Руководство по настройке лог-коллектора, Включение компонентов*

   Основные параметры, которые нужно указать при включении компонентов сбора:

   `collectors:`

   `udp_reciever:`

   `- <<: *<"id компонента сбора"> (ID компонента сбора, который указывали при объявлении компонента сбора)`

   Основные параметры, которые нужно указать при включении компонентов отправки:

   `senders:`

   `tcp:`

   `- <<: *<"id компонента отправки"> (ID компонента отправки, который указывали при объявлении компонента отправки)`

4. После чего необходимо произвести настройку маршрутизации событий.

   Пример настройки по умолчанию можно найти в документации: *Руководство по настройке лог-коллектора, Маршрутизация событий*

   Основные параметры, которые нужно указать при настройке маршрута:

   `route_1: &route_1`

`collector_id:`

`- <"id компонента сбора">`

`sender_id:`

`- <"id компонента отправки">`

5. После нужно включить маршрут в разделе routers. Пример включения маршрута:

   `routers:`

   `- <<: *<название маршрута> (например - <<: *route_1)`

5.2. Программный комплекс СКДПУ НТ {#skdpunt}

Инструкция по настройке программного комплекса «Система контроля действий поставщиков ИТ-услуг «Новые технологии» (СКДПУ НТ) для отправки событий в **Платформу Радар**:

1. Зайдите в веб-интерфейс системы СКДПУ НТ под учетной записью с правами администратора системы.
2. Выберите последовательно пункты меню: «Система» -> «Интеграция с SIEM»
3. В открывшемся окне выполните настройки:
   - выбрать «Включено» в поле «Роутинг»;
   - заполнить имя хоста или IP-адрес лог-коллектора в поле «Доменное имя или IP»;
   - заполнить порт, открытый на лог-коллекторе для приема событий от данного источника, в поле «Порт»;
   - выбрать протокол взаимодействия (TCP/UDP) в поле «Протокол»;
   - выбрать формат отправки событий в поле «Log format»;
   - выбрать формат отображения времени в отправляемом событии в поле «Формат времени»;
   - нажать «+» для добавления конфигурации, а затем «Применить» для сохранения изменений (см. рисунок 49).

Рисунок 49 - Сохранение настройки СКДПУ НТ.

5.3. McAfee Web Gateway {#mawebgateway}

Инструкция по настройке McAfee Web Gateway для отправки событий в **Платформу Радар**:

1. Зайдите в интерфейс системы под учетной записью с правами администратора системы.
2. Зайдите в меню «Policy», затем выберите вкладку «Rule Sets» и пункт меню «Log Handler» (см. рисунок 50).

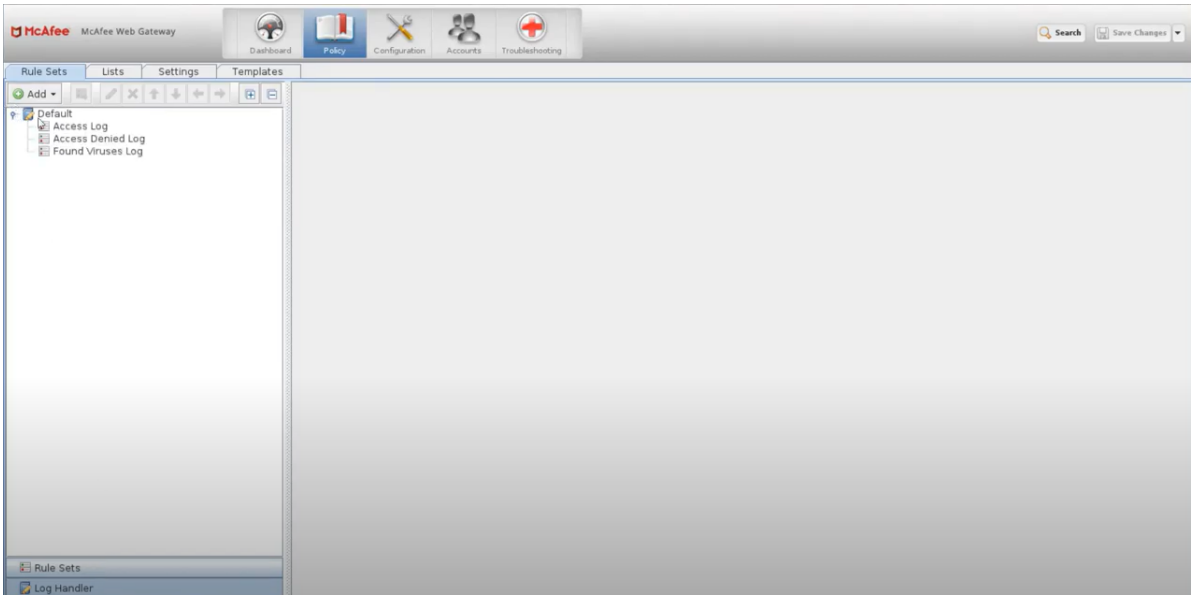

Рисунок 50 - Выбор логов.

3. Раскройте список «Default», выберите «Access Log», в правой части окна выделите правило и нажмите «Edit».
4. В секции «Events» нажмите «Add», а затем «Event».
5. Выберите «Syslog (Number, String)» и нажмите «Parameters».
6. Для параметра «1. Level (Number)» установите значение 6, что указывает на уровень логирования «Informational». Для настройки параметра «2. Message (String)» нажмите «Use Property» и выберите «User-Defined.logLine».
7. Нажмите последовательно «OK» -> «OK» -> «Finish».
8. Повторите действия п.п. 3-7 для других наборов правил.
9. Перейдите в меню «Configuration», выберите вкладку «File Editor».
10. Разверните список с именем соответствующего устройства и выберите файл rsyslog.conf.
11. Найдите в файле следующую строку:

    ```
    *.info;mail.none;authpriv.none;cron.none  /var/log/messages
    ```

    Добавьте в нее «daemon.!=info» следующим образом:

    ```
    *.info;daemon.!=info;mail.none;authpriv.none;cron.none  -/var/log/messages
    ```

    Также добавьте следующую строку для отправки событий на лог-коллектор (@ - отправка по протоколу UDP, @@ - отправка по протоколу TCP):

    ```
    daemon.info @<ip-адрес лог-коллектора>:<порт лог-коллектора>
    ```

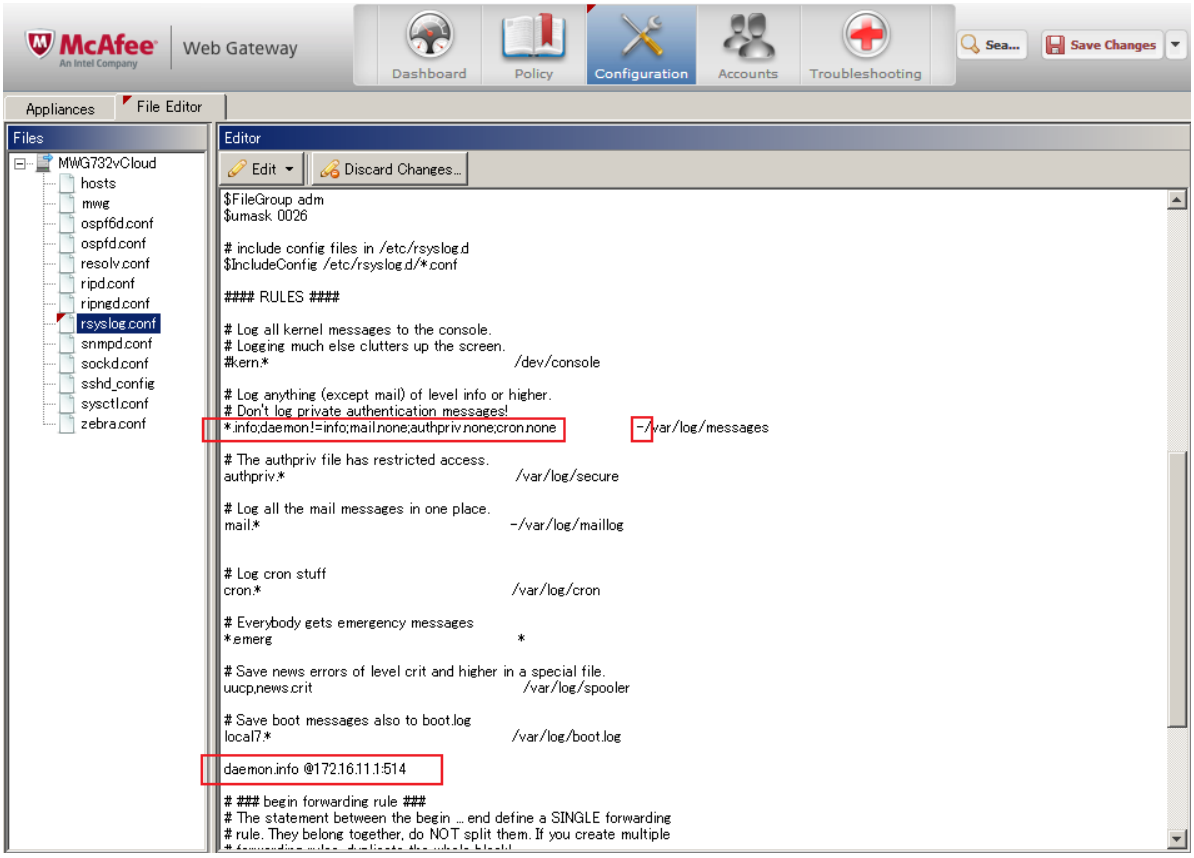

Рисунок 51 - Редактирование rsyslog.sys.

12. Нажмите кнопку «Save Changes» для сохранения изменений.

5.4. nGate Firewall {#ngate}

5.4.1. Настройка подключения источника nGate

По умолчанию логирование событий аутентификации и изменение конфигураций сохраняется в журнал ng-admin.log по пути `/var/log/ngate/ng-admin/`.

Для настройки пересылки логов с помощью `rsyslog` перейдите в директорию `/etc/rsyslog.d/` и откройте файл конфигурации `50-ng-manual-fwd.conf`. Закомментируйте содержимое и вставьте следующую конструкцию, после чего перезапустите службу `rsyslog`:

```
module(load="imfile" PollingInterval="10")
input(type="imfile"
      reopenOnTruncate="on"
      File="/var/log/ngate/ng-admin/ng-admin.log"
      Tag="ng-admin")
if $syslogtag == 'ng-admin' then @IP:PORT
& stop
```

5.4.2. Настройки конфигурации log-collectora

```
# = nGate =
udp_input_2562: &udp_input_2562
  id: "udp_input_2562"
  host: "collector_IP"
  port: 2562
```

```
  sock_buf_size: 0
  format: "json"

tcp_output_2562: &tcp_output_2562
  id: "tcp_output_2562"
  target_host: "platform_IP"
  port: 2562
  log_level: "INFO"

senders:
  port: 48002
  log_level: "INFO"
  tcp:
    - <<: *tcp_output_2562
collectors:
  log_level: "INFO"
  udp_receiver:
    - <<: *udp_input_2562
#
route_1: &route_1
  collector_id:
    - "udp_input_2562"
  sender_id:
    - "tcp_output_2562"
routers:
  - <<: *route_1
```

5.5. pfSense Firewall {#pfsense}

5.5.1. Настройка подключения источника Pfsense

Для настройки отправки событий в **Платформу Радар** от pfSense Firewall перейдите в веб-интерфейс pfSense по адресу Status > System Logs > Settings.

Прокрутите страницу вниз до Remote Logging Options.

Выполните настройку (см. рисунок 52):

1. Включить настройку отправки логов.
2. Выбрать источник.
3. Выбрать протокол.
4. Указать адрес хоста, на который будут отправляться логи. IP:PORT.
5. Выбрать, какие логи необходимо отправлять.
6. Сохранить настройки.

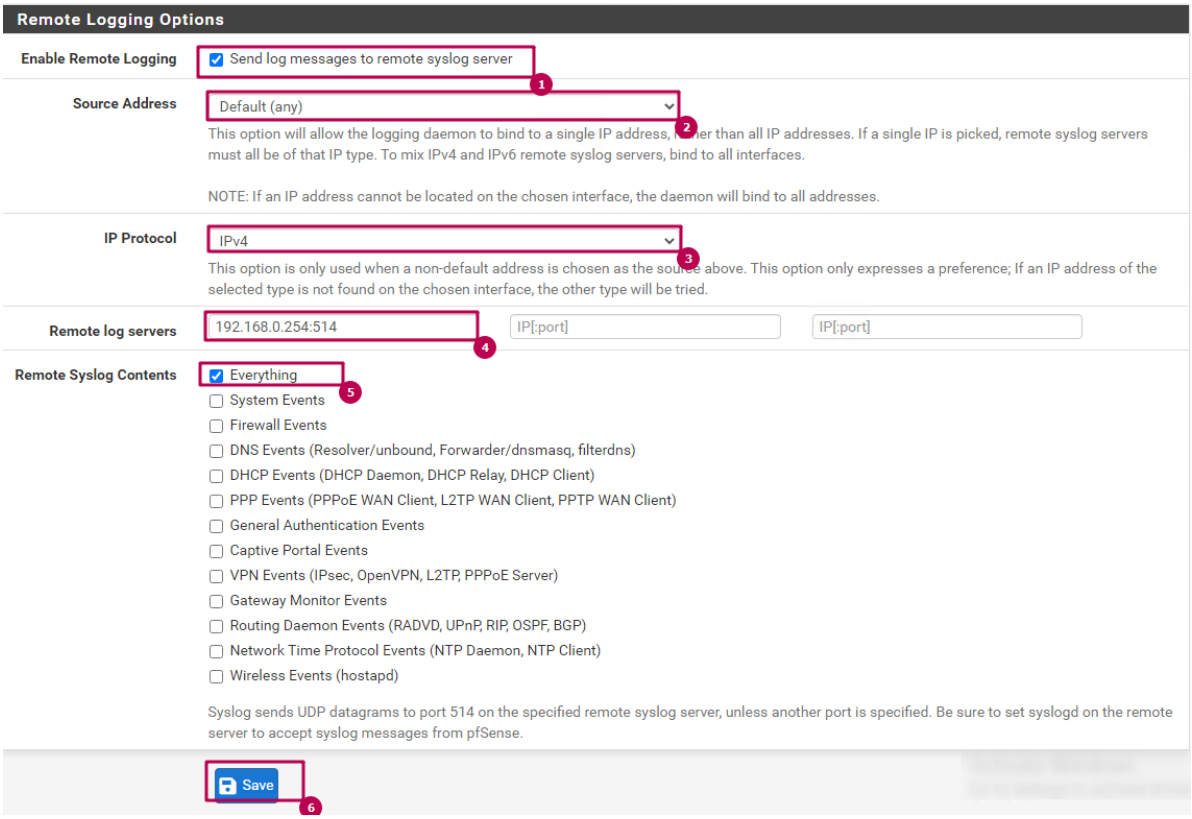

Рисунок 52 - Настройка pfSense.

5.5.2. Настройки конфигурации log-collectora

```
# = pFsense =
udp_input_515: &udp_input_515
  id: "udp_input_515"
  host: "172.30.254.166"
  port: 515
  sock_buf_size: 0
  format: "json"

tcp_output_2561: &tcp_output_2561
  id: "tcp_output_2561"
  target_host: "172.30.254.67"
  port: 2561

#=====
senders:
  port: 48002
  log_level: "INFO"
  tcp:
    - <<: *tcp_output_2561
collectors:
  log_level: "INFO"
  udp_receiver:
    - <<: *udp_input_515
#====
route_1: &route_1
  collector_id:
    - "udp_input_515"
  sender_id:
    - "tcp_output_2561"
```

```
#====
routers:
  - <<: *route_1
```

5.6. Usergate UTM Firewall {#usergate}

Подключение UserGate UTM Firewall в качестве источника событий для **Платформы Радар**

1. В Web-интерфейсе UserGate UTM перейдите в раздел «Настройки» и выберите пункт «Журналы и отчеты» (см. рисунок 53)

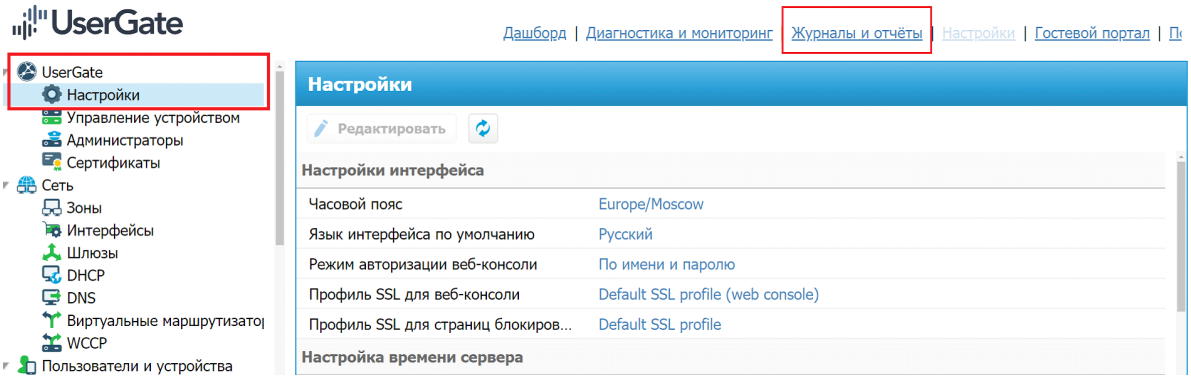

Рисунок 53 - Настройка Usergate.

2. Выберите пункт «Экспорт журналов» и нажмите кнопку «Добавить» (см. рисунок 54)

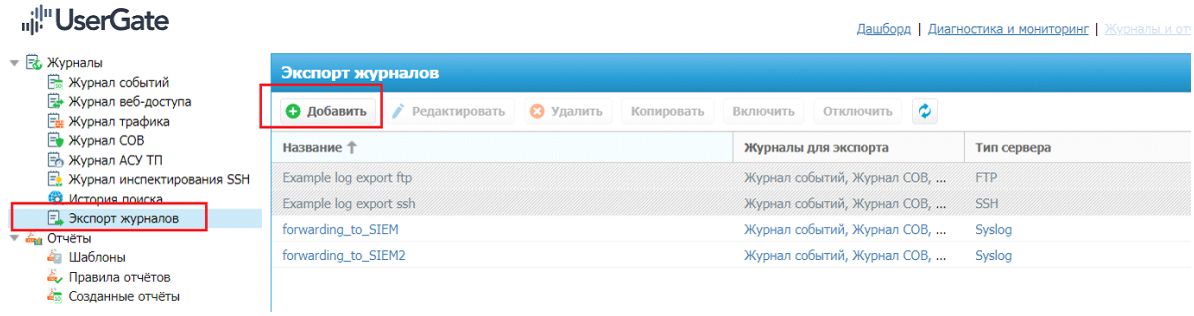

Рисунок 54 - Добавление экспорта журнала.

3. В меню «Свойства правила экспорта журналов» (см. рисунок 55) выполните нижеуказанные действия:
    - Во вкладке «Общие» (все отдельные слова в названии необходимо писать через нижнее подчеркивание «_»):
        - Установить чекбокс в стоке «Включено»;
        - Заполнить строку «Название».

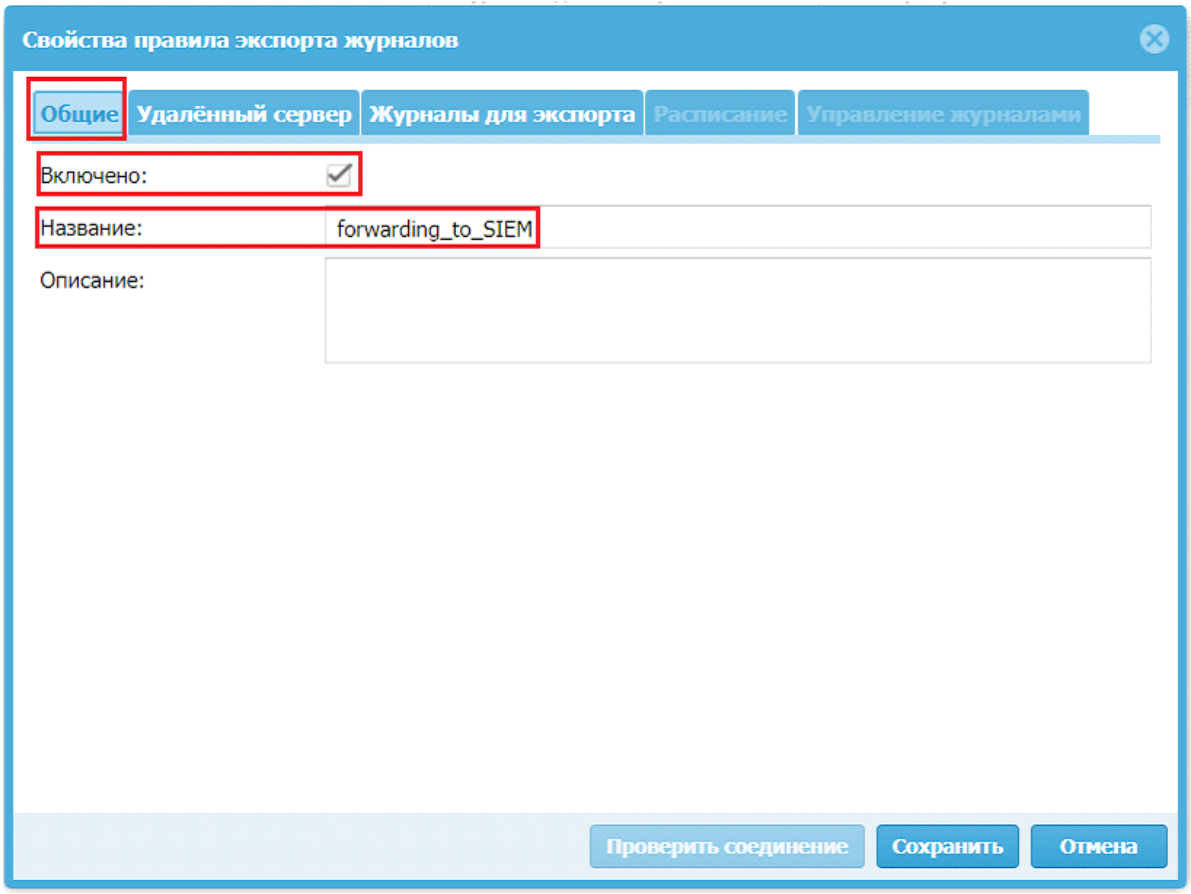

Рисунок 55 - Свойства правила экспорта журналов.

```
- Во вкладке «Удаленный сервер» (см. рисунок 56):

    - в графе «Тип сервера» установить значение «Syslog»;
    - в графе «адрес сервера» укать ip-адрес лог-коллектора;
    - указать порт для отправки событий;
    - в графе «Транспорт» установить значение «UDP»;
    - в графе «Протокол» установить значение «Syslog (RFC 5424)»;
    - в графе «Критичность» установить значение «Уведомительная»;
    - в графе «Объект» установить значение «Сообщения пользовательские»;
    - графы «Имя хоста» и «Название приложения» указать без пробелов.

По-умолчанию **Платформой Радар** для источника UserGate UTM выделен порт 2545
```

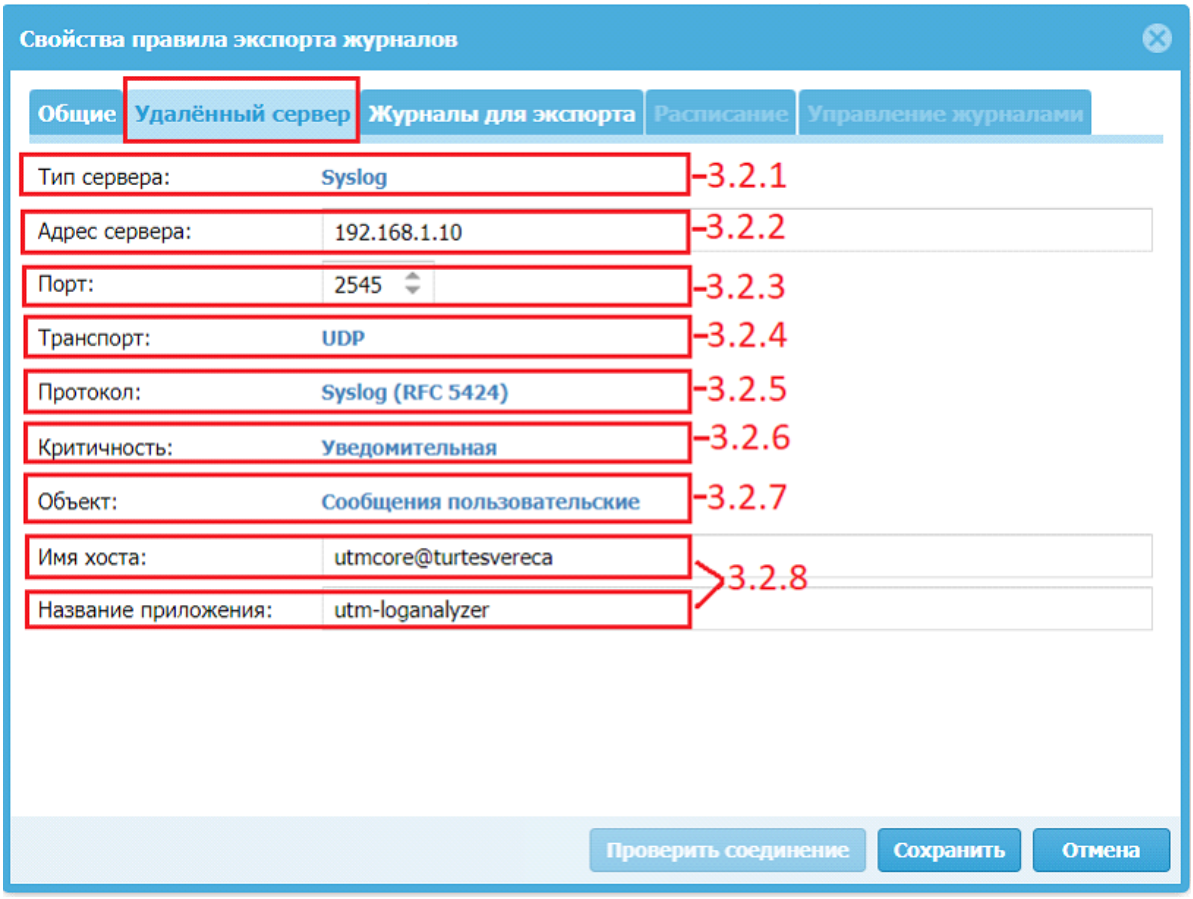

Рисунок 56 - Свойства удаленного сервера.

- Во вкладке «Журналы для экспорта» (см. рисунок 57) установите чекбоксы напротив журналов:

    - журнал событий;
    - журнал СОВ;
    - журнал трафика;
    - журнал веб-доступа;
    - выставить для всех журналов формат «JSON»;
    - нажать кнопку «Сохранить».

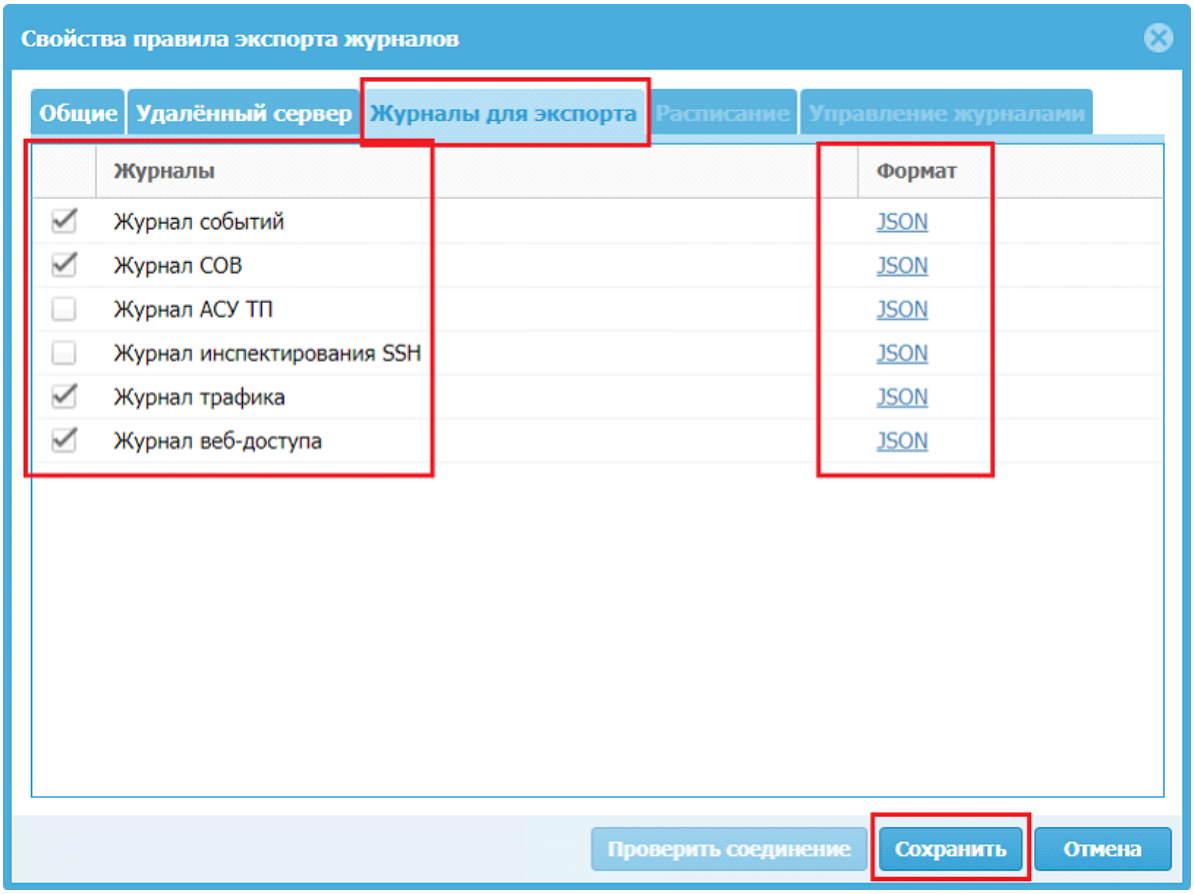

Рисунок 57 - Выбор журналов для экспорта.

5.7. Citrix ADC (Netscaler) {#netscaler}

Данное руководство описывает механизм сбора событий Citrix ADC (Netscaler) и отправки их в **Платформу Радар**. Для настройки сбора событий выполните шаги:

1. Войдите Web-интерфейс Citrix ADC (см. рисунок 58).

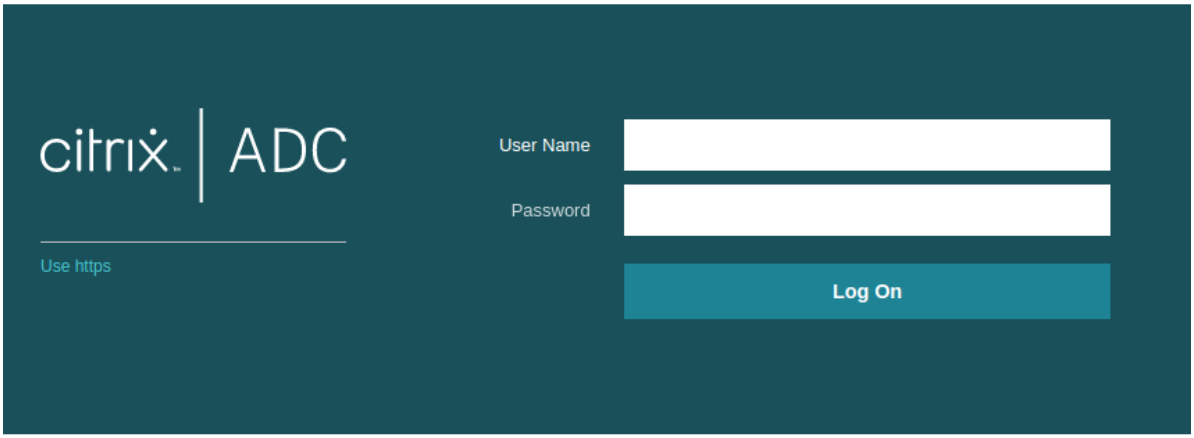

Рисунок 58 - Вход в Web-интерфейс Citrix ADC.

2. Перейдите в раздел Configuration > System > Auditing > Syslog (см. рисунок 59).

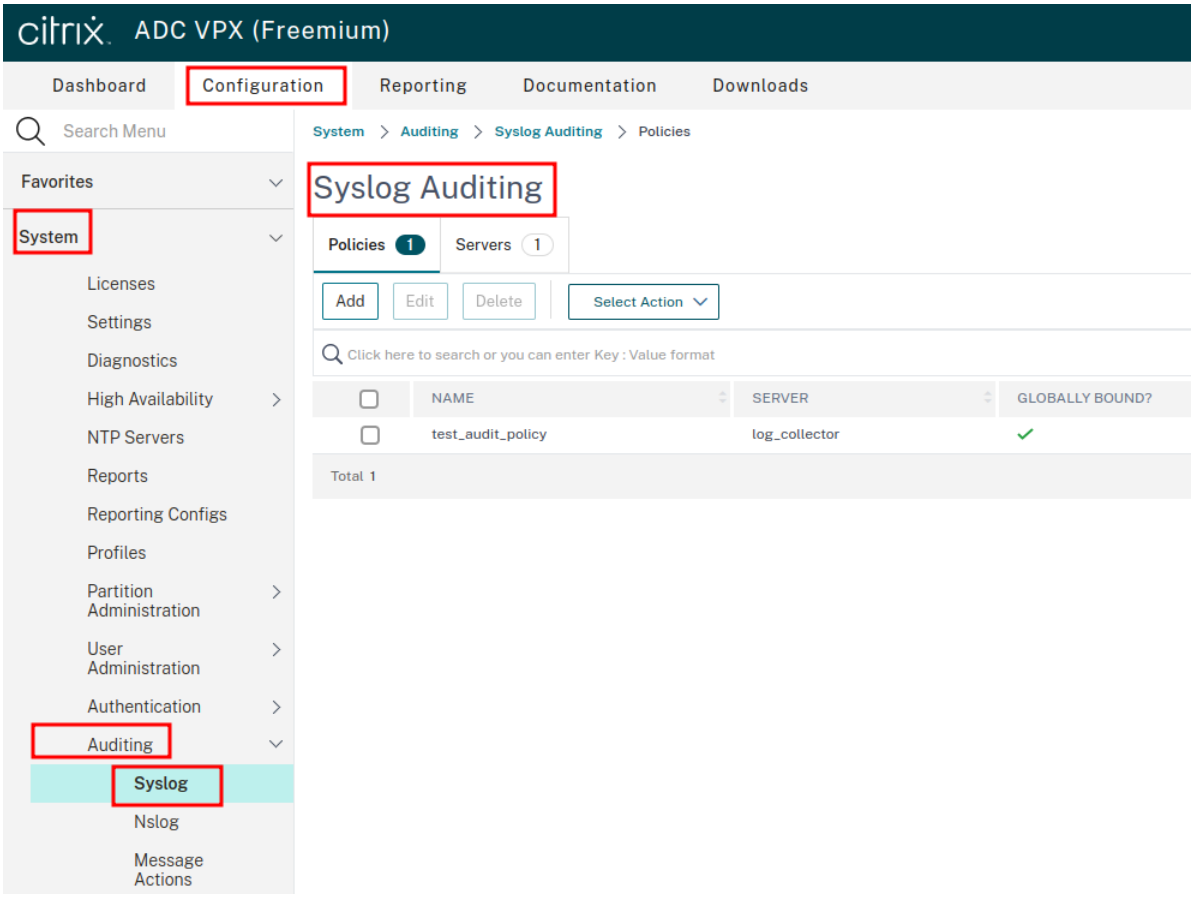

Рисунок 59 - Переход к логгированию.

3. Откройте вкладку *Servers*.
4. Нажмите кнопку *Add*.
5. На странице *Create Auditing Server* заполните необходимые поля, все (см. рисунок 60). Не забудьте указать актуальный адрес лог коллектора и выбранный порт.

Create Auditing Server

Auditing Type

SYSLOG

Name

log_collector

Server

Server Type*

Server IP

IP Address*

Port

2871

Log Levels

ALL NONE CUSTOM

Log Facility*

LOCAL0

Date Format*

DDMMYYYY

Time Zone

GMT Local

- [x] TCP Logging
- [x] ACL Logging
- [x] User Configurable Log Messages
- [x] AppFlow Logging
- [x] Large Scale NAT Logging
- [x] ALG messages Logging
- [x] Subscriber Logging
- [x] DNS
- [x] SSL Interception
- [x] URL Filtering
- [x] Content Inspection Logging

Net Profile

Add

Transport Type

TCP

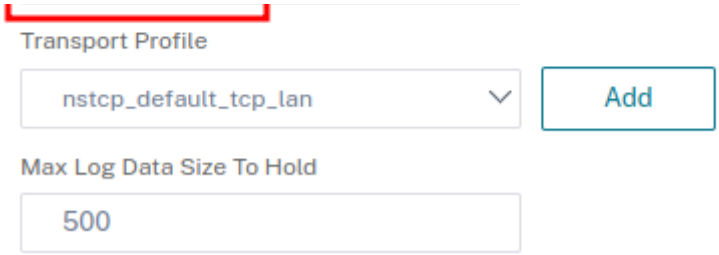

Рисунок 60 - Создание аудита.

6. Нажмите кнопку *Create*.
7. Создайте `syslog policy`. Для этого перейдите на вкладку *Policies* и нажмите кнопку *Add*.
8. На странице *Create Auditing Syslog Policy* заполните поля (см. рисунок 61).

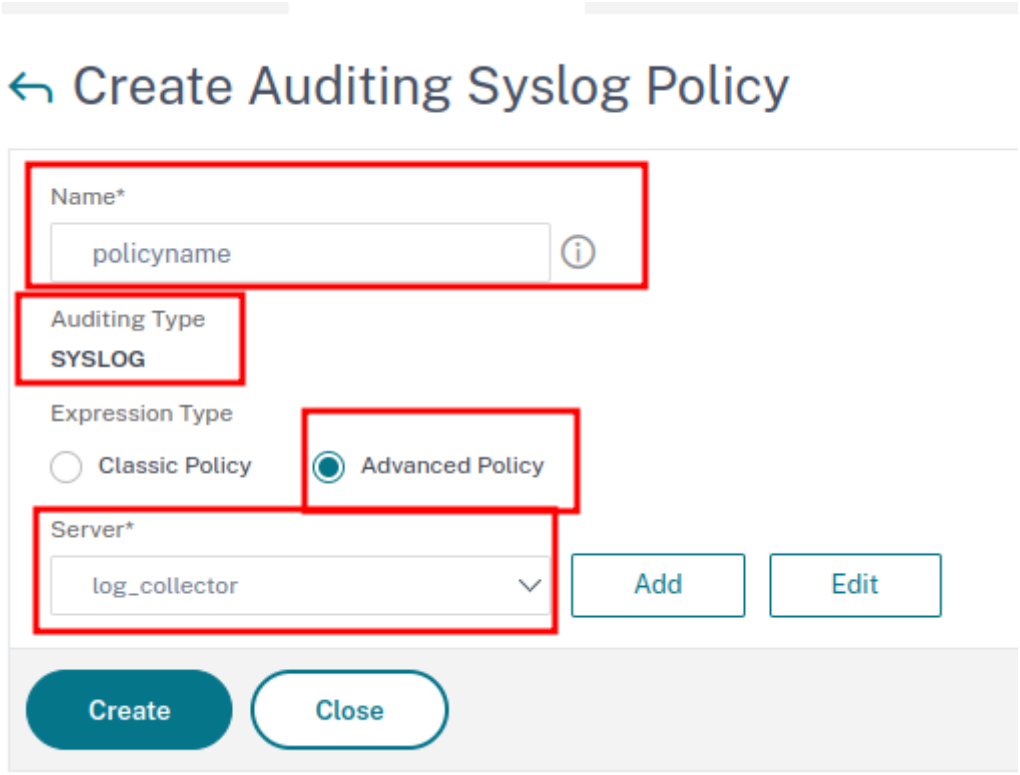

Рисунок 61 - Заполнение полей аудита.

```
Введите название политики и выберите syslog сервер, который был добавлен ранее
(п.п. 3-6)
```

9. Нажмите кнопку *Create*.

Настройка источника на этом закончена. Более детальную информацию о параметрах, а также о способе настройки источника с помощью командной строки, можно прочитать в документации на сайте вендора.

Мы рекомендуем настраивать источник через web-интерфейс и использовать указанные параметры конфигурации. При конфигурировании через командную сроку используйте точно такие же параметры. Изменение любого из них может повлиять на корректность работы правил разбора в **Платформе Радар**.

Пример конфигурационного файла лог коллектора:

```
cluster:
```

```
  url: "https://адрес_сервера"
  api_key: "api_key"
controller:
  port: 48000
metric_server:
  port: 48005
secret_file: "/opt/pangeoradar/configs/logcollector/secret"
secret_storage: "/opt/pangeoradar/configs/logcollector/secret_storage"
api_server:
  address: "server ip or address"
  port: 8001
  read_timeout: 60
  write_timeout: 60
  wait: 5
  enable_tls: true
  cert_file: "/opt/pangeoradar/certs/agent.crt"
  key_file: "/opt/pangeoradar/certs/agent.key"
  cert_key_pass: ""
  require_client_cert: false
  ca_file: "/opt/pangeoradar/certs/pgr.crt"
  log_level: "INFO"

journal:
  port: 48004
  log_level: "INFO"
  log_path: "/var/log/logcollector/journal.log"
  rotation_size: 30
  max_backups: 7
  max_age: 7

tcp_input_citrix_adc: &tcp_input_citrix_adc
  id: "tcp_input_citrix_adc"
  host: "172.30.250.32"
  port: 2871 # Здесь можно указать любой незанятый порт, не забудьте указать его
же в конфигурации источника
  enable_tls: false
  compression_enabled: false
  connections_limit: 10
  format: "json"
  log_level: "INFO"

tcp_output_citrix_adc: &tcp_output_citrix_adc
  id: "tcp_output_citrix_adc"
  target_host: "172.30.254.68"
  port: 2870

senders:
  port: 48001
  tcp:
    - <<: *tcp_output_citrix_adc

collectors:
  log_level: "INFO"
  tcp_receiver:
    - <<: *tcp_input_citrix_adc
```

```
route_citrix_adc: &route_citrix_adc
  collector_id:
    - "tcp_input_citrix_adc"
  sender_id:
    - "tcp_output_citrix_adc"

routers:
  - <<: *route_citrix_adc
```

При необходимости откройте нужные порты на межсетевом экране (порты указаны в файле конфигурации).
Перезапустите службу лог коллектора.
Проверьте наличие событий в интерфейсе **Платформы Радар**.

5.8. Checkpoint NGFW {#checkpoint}

Настройка сбора событий Checkpoint через log-export.
Для настройки отправки событий с Checkpoint firewall по syslog выполните следующие шаги:

1. Подключитесь по ssh к инстансу Checkpoint.
2. Переключитесь в режим expert:

   `> expert`
3. Выполните команду для создания конфигурации отправки:

   `# cp_log_export add name <имя конфигурации> target-server <ip-адрес лог-коллектора> target-port 2511 protocol tcp format <формат событий syslog>`
4. Запустите конфигурацию командой:

   `# cp_log_export restart name <имя конфигурации>r`

   Если в конфигурации была допущена ошибка, то для ее изменения выполните команду:

   `# cp_log_export set name <имя конфигурации> [параметры значения]`
5. На машине с лог-коллектором добавьте изменения в файл конфигурации для сбора событий от источника и отправки их в **Платформу Радар**:
   - добавить Компонент сбора событий:

     ```
     tcp_input_checkpoint: & tcp_input_checkpoint
         id: "tcp_input_checkpoint"
       host: "0.0.0.0"
         port: 2511
         sock_buf_size: 0
       format: "json"
         log_level: "INFO"
     ```

   - добавить Компонент отправки событий:

```
tcp_output_checkpoint: & tcp_output_checkpoint
id: "tcp_output_checkpoint "
  target_host: "<ip адрес  Платформы Радар/или балансера>"
port: 2511
  sock_buf_size: 0
  log_level: "INFO"
```

- указать добавленные компоненты сбора и отправки в разделы collectors и senders соответственно:

```
collectors:
    tcp_receiver:
    - <<: *tcp_input_checkpoint

senders:
  port: 48002
    tcp:
    - <<: *tcp_output_checkpoint
```

- добавить маршрут взаимодействия между компонентами сбора событий и компонентами отправки событий:

```
route_1_checkpoint: &route_1_checkpoint
    collector_id:
    - "udp_input_checkpoint "
    sender_id:
    - "tcp_output_checkpoint "
```

- включить маршрут в разделе конфигурационного файла routers:

```
routers:
  - <<: *route_1_checkpoint
```

Перезапустите службу лог-коллектора.
Включите источник Checkpoint в **Платформе Радар** и нажмите кнопку «Синхронизировать».
Проверьте поступающие события в **Платформе Радар** в разделе «Просмотр событий».

5.9. Cisco snort {#snort}

5.9.1. Настройка rsyslog на сервере snort.

Логирование snort выполняется в системный журнал syslog.

Для отправки логов в **Платформу Радар** выполните шаги:

1. Создайте шаблон для rsyslog'a по пути `/etc/rsyslog.d/`. Например `snort.conf`

   `sudo nano /etc/rsyslog.d/snort.conf`

   Содержимое файла представлено ниже:

```
If ($programname contains 'snort' and ($msg contains 'start' or $msg
contains 'Start' or $msg contains 'Stop' or $msg contains 'stop' or $msg
contains 'ERROR' or $msg contains 'fail' or $msg contains 'Fail')) or ($msg
contains 'snort' and $msg contains 'exit') then @@x.x.x.x:515
If $msg contains 'Classification' and $programname contains 'snort' then
@@x.x.x.x:515
```

Где вместо x.x.x.x необходимо указать ip-адрес лог-коллектора и порт после двоеточия.

Первая строчка конфигурации позволяет отправлять в **Платформу Радар** системные логи, исключая не информативные.
Вторая строчка включает пересылку алертов в **Платформу Радар**.

2. Перезапустить службу rsyslog.

   `systemctl restart rsyslog`

5.9.2. Настройки конфигурации log-collectora

```
tcp_input_515: &tcp_input_515
  id: "tcp_input4"
  host: "0.0.0.0"
  port: 515
  sock_buf_size: 0
  format: "json"

tcp_output_2517: &tcp_output_2517
  id: "tcp_output_4"
  target_host: "x.x.x.x"
  port: 2517

senders:
  port: 48002
  log_level: "INFO"
  tcp:
    - <<: *tcp_output_2517

collectors:
  tcp_receiver:
    - <<: *tcp_input_515

route_2517: &route_2517
  collector_id:
    - "tcp_input_515"
  sender_id:
    - "tcp_output_2517"

routers:
  - <<: *route_2517
```

Вместо x.x.x.x необходимо также указать ip-адрес лог-коллектора и выбранный ранее порт для tcp_input.

6. Системы антивирусной защиты

6.1. О событиях в Kaspersky Security Center {#kaspersky}

В Kaspersky Security Center существуют следующие типы уведомлений:

- *Общие события*. Эти события возникают во всех управляемых программах "Лаборатории Касперского". Например, общее событие Вирусная атака. Общие события имеют строго определенные синтаксис и семантику. Общие события используются, например, в отчетах и панели мониторинга.
- *Специфические события управляемых программ "Лаборатории Касперского"*. Каждая управляемая программа "Лаборатории Касперского" имеет собственный набор событий.

Каждое событие имеет собственный уровень важности. В зависимости от условий возникновения, событию могут быть присвоены различные уровни важности.

Существует четыре уровня важности событий:

- *Критическое событие* – событие, указывающее на возникновение критической проблемы, которая может привести к потере данных, сбою в работе или критической ошибке.
- *Отказ функционирования* – событие, указывающее на возникновение серьезной проблемы, ошибки или сбоя, произошедшего во время работы программы или выполнения процедуры.
- *Предупреждение* – событие, не обязательно являющееся серьезным, однако указывающее на возможное возникновение проблемы в будущем. Чаще всего события относятся к Предупреждениям, если после их возникновения работа программы может быть восстановлена без потери данных или функциональных возможностей.
- *Информационное сообщение* – событие, возникающее с целью информирования об успешном выполнении операции, корректной работе программы или завершении процедуры.

Для каждого события задано время хранения, которое можно посмотреть или изменить в Kaspersky Security Center. Некоторые события не сохраняются в базе данных Сервера администрирования по умолчанию, поскольку для них установленное время хранения равно нулю. Во внешние системы можно экспортировать только те события, которые хранятся в базе данных Сервера администрирования не менее одного дня.

6.2. Kaspersky Security Center через Microsoft SQL Server

6.2.1. Настройка источника

1. Создание учетной записи для сбора событий.

   Для сбора событий с базы данных Kaspersky Security Center необходимо создать учетную запись с членством в роли db_datareader для базы KAV.

   Процесс создания учетной записи приведен в разделе *Создание учетной записи Microsoft SQL Server.*

2. При использовании межсетевого экрана на узле необходимо сделать правило для входящих соединений.

6.2.2. Включение источника на Платформе Радар

> Включение источника в **Платформе Радар** представлено в разделе *Управление источниками в Платформе, Включение/выключение поддерживаемых источников и их синхронизация*

1. Зайти в веб-консоль **Платформы Радар**, перейти в раздел «Источники», «Управление источниками»;
2. Найти в списке доступных источников «Kaspersky-SecurityCenter-db» и включить его;
3. Кликнуть на кнопку «Синхронизировать».

6.2.3. Настройка коллектора событий

> Пример конфигурационного файла с настройкой данного источника представлен в разделе *Пример конфигурационного файла лог-коллектора*. Более подробная информация о настройках лог-коллектора представлена в разделе *Руководство по настройке лог-коллектора*

1. В конфигурационный файл лог-коллектора (config.yaml) необходимо добавить input компонента ODBC.

   Пример настройки по умолчанию можно найти в документации: *Руководство по настройке лог-коллектора, Компонент ODBC*

   Основные параметры, которые необходимо указать:

   `connection_string: "Driver={ODBC Driver 17 for SQL Server};Server=<ip-адрес>;Port=1433;Database=KAV;UID=<username>;PWD=<password>;"`

   Строка с sql запросом к базе представлена в разделе SQL запрос для KSC.

2. После настройки компонента сбора событий (input) - необходимо настроить компонент отправки событий (output).

   В качестве компонента отправки событий для данного источника предусмотрено использование отправки по протоколу TCP.

   Пример настройки по умолчанию можно найти в документации: *Руководство по настройке лог-коллектора, Компонент отправки событий по протоколу TCP*

   Основные параметры, которые необходимо указать:

   `target_host: <"ip адрес или имя удаленного узла"> (адрес **Платформы Радар**)`

   `port: <"порт"> (стандартный порт для данного источника 2604)`

3. Далее необходимо включить компоненты сбора (collectors) и отправки (senders).

   Пример настройки по умолчанию можно найти в документации: *Руководство по настройке лог-коллектора, Включение компонентов*

   Основные параметры, которые нужно указать при включении компонентов сбора:

   `collectors:`

   `odbc:`

   `- <<: *<"id компонента сбора"> (ID компонента сбора, который указывали при объявлении компонента сбора)`

   Основные параметры, которые нужно указать при включении компонентов отправки:

   `senders:`

`tcp:`

`- <<: *<"id компонента отправки"> (ID компонента отправки, который указывали при объявлении компонента отправки)`

4. После чего необходимо произвести настройку маршрутизации событий.

   Пример настройки по умолчанию можно найти в документации: *Руководство по настройке лог-коллектора, Маршрутизация событий*

   Основные параметры, которые нужно указать при настройке маршрута:

   `route_1: &route_1`

   `collector_id:`

   `- <"id компонента сбора">`

   `sender_id:`

   `- <"id компонента отправки">`

5. После нужно включить маршрут в разделе routers. Пример включения маршрута:

   `routers:`

   `- <<: *<название маршрута> (например - <<: *route_1)`

6.2.4. Создание учетной записи Microsoft SQL Server {#create_account}

Создание имени входа на сервер

Настройку сервера необходимо выполнять от имени учетной записи, имеющей права локального администратора ОС Windows. Для создания данной учётной записи необходимо выполнить следующие действия:

1. В меню Пуск открыть среду разработки MS SQL Management Studio (Диспетчер конфигурации SQL Server).
2. В окне Connect to Server (Соединение с сервером) подключится к экземпляру необходимой базы данных (БД) с правами администратора sa (см. рисунок 62).

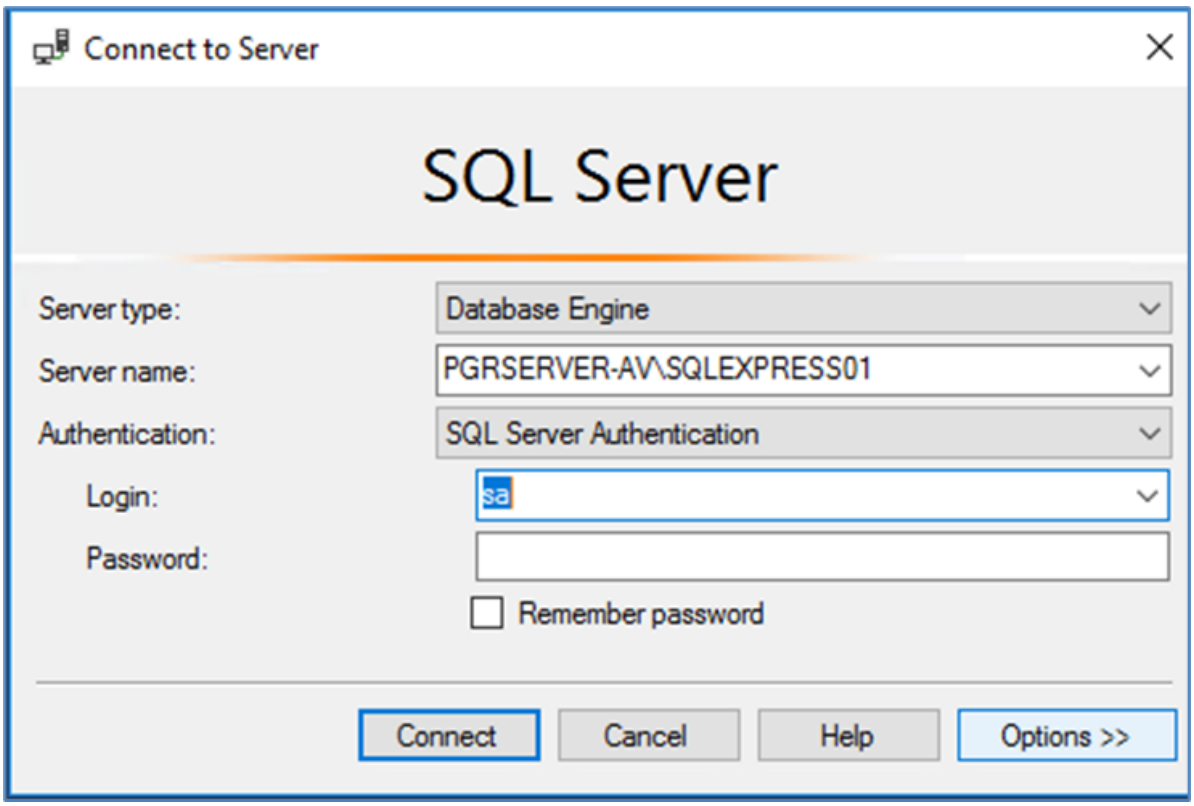

Рисунок 62 - Подключение к экземпляру БД

3. Подключится к экземпляру БД. Для предоставления доступа к экземпляру БД выполнить следующие действия:
    - В окне Object Explorer (Обозреватель объектов) выбранного экземпляра БД (SQLEXPRESS) открыть контекстное меню раздела Logins (Имена для входа):

      Security → Logins (Безопасность → Имена для входа)
    - В контекстном меню выбрать команду New Login (Создать имя для входа - см. рисунок 63).

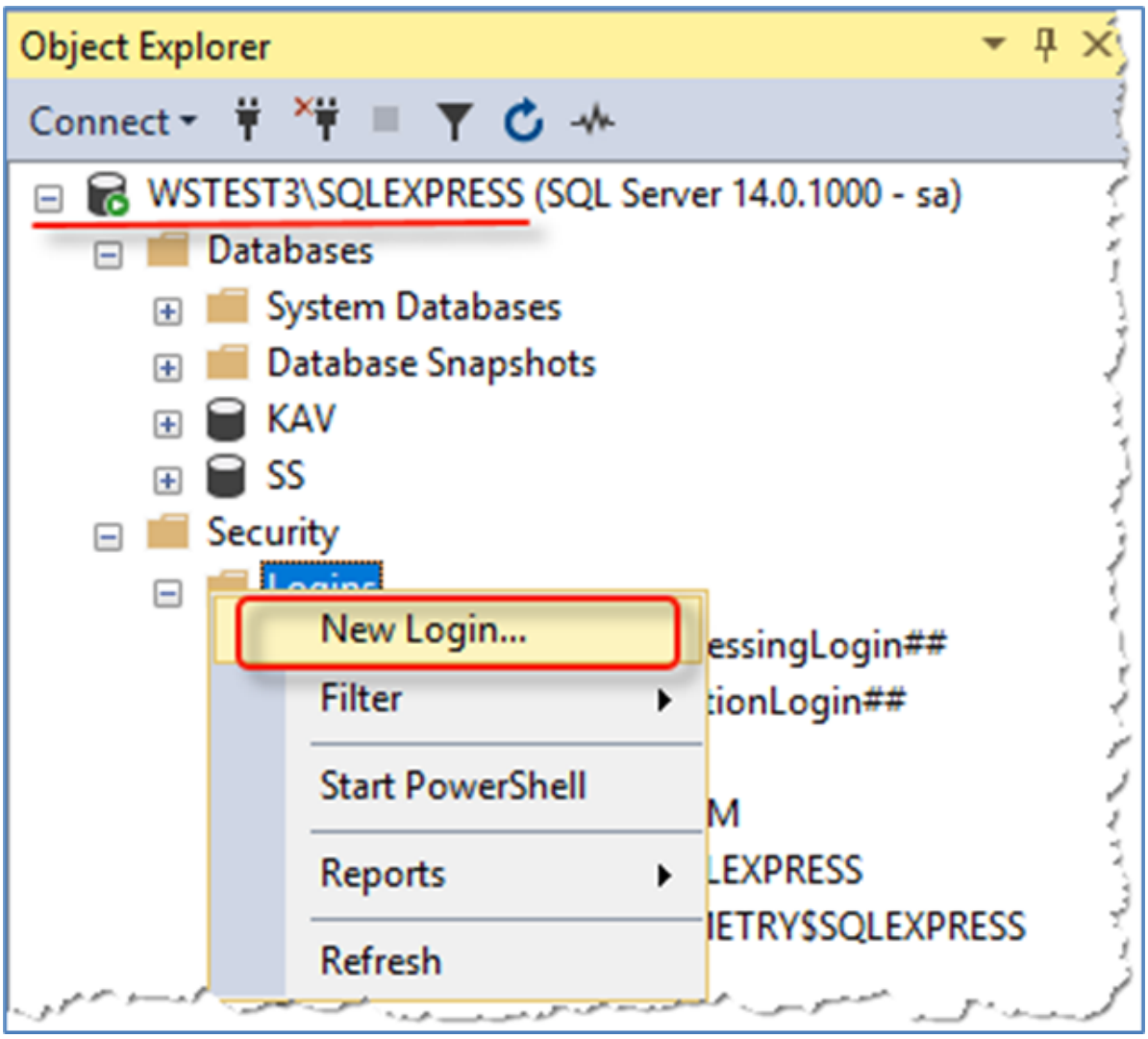

Рисунок 63 - Дерево каталогов экземпляра БД

- В открывшемся окне Login--New (Создание имени для входа) в разделе General (Общие) выполнить следующие настройки (см. рисунок 64):

- Ввести имя пользователя (*radaruser*) в поле Login Name (Имя для входа).

- Установить пароль в полях Password и Confirm Password (Пароль, Подтверждение пароля).

- При необходимости выставить настройки в пунктах:

    - Enforce password policy (Требовать использование политики паролей);

    - Enforce password expiration (Задать срок окончания действия пароля).

-   Выбрать режим SQL Server authentication (Проверка подлинности SQL Server).

-   Выбрать *KAV* в качестве БД по умолчанию в раскрывающемся списке Default Database (База данных по умолчанию).

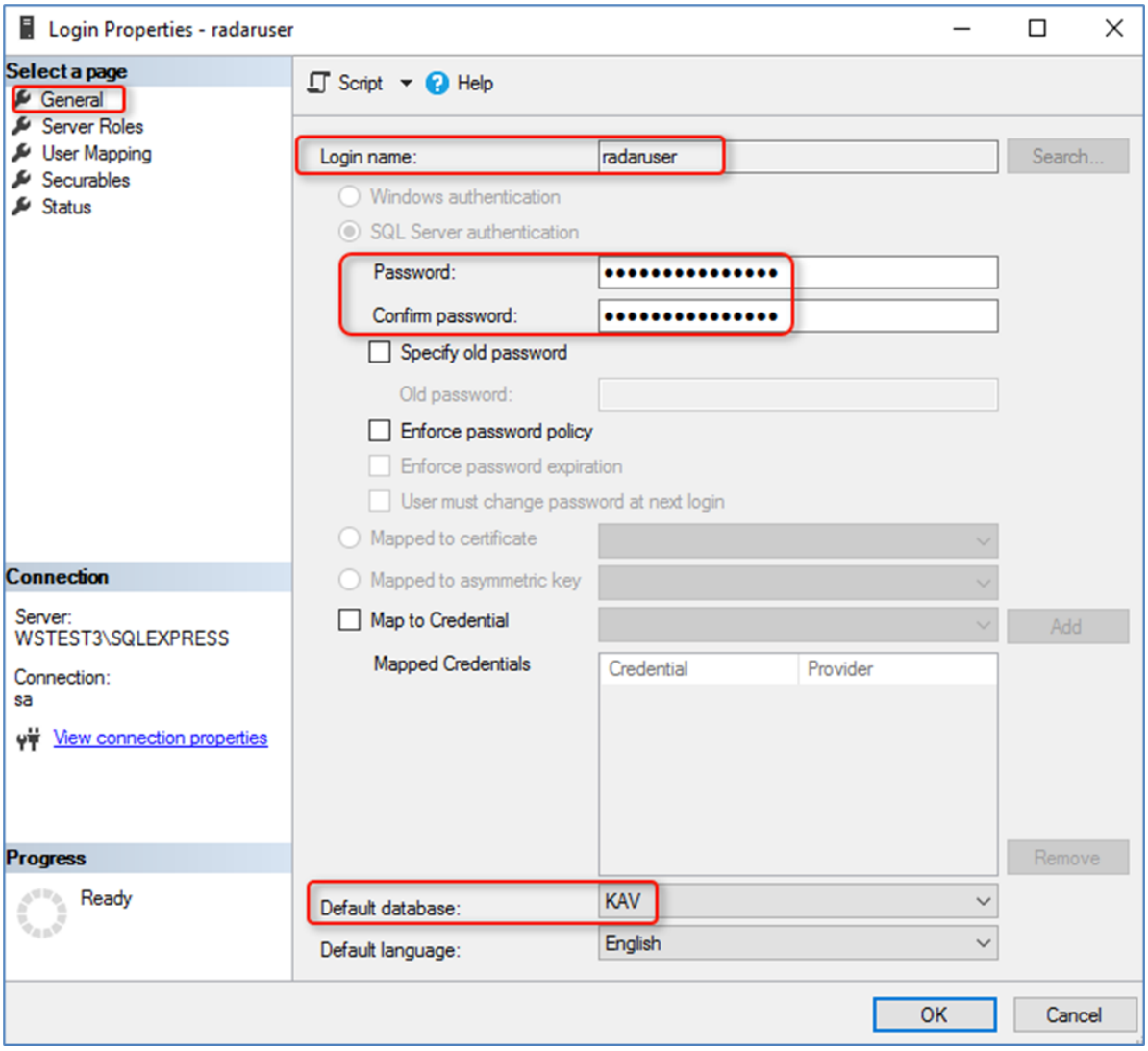

Рисунок 64 - Создание нового пользователя экземпляра БД

4. В разделе Server Roles (Роли сервера) проверить что пользователю предоставлена роль *public* (см. рисунок 65).

   Если она не предоставлена, то предоставить пользователю роль *public*.

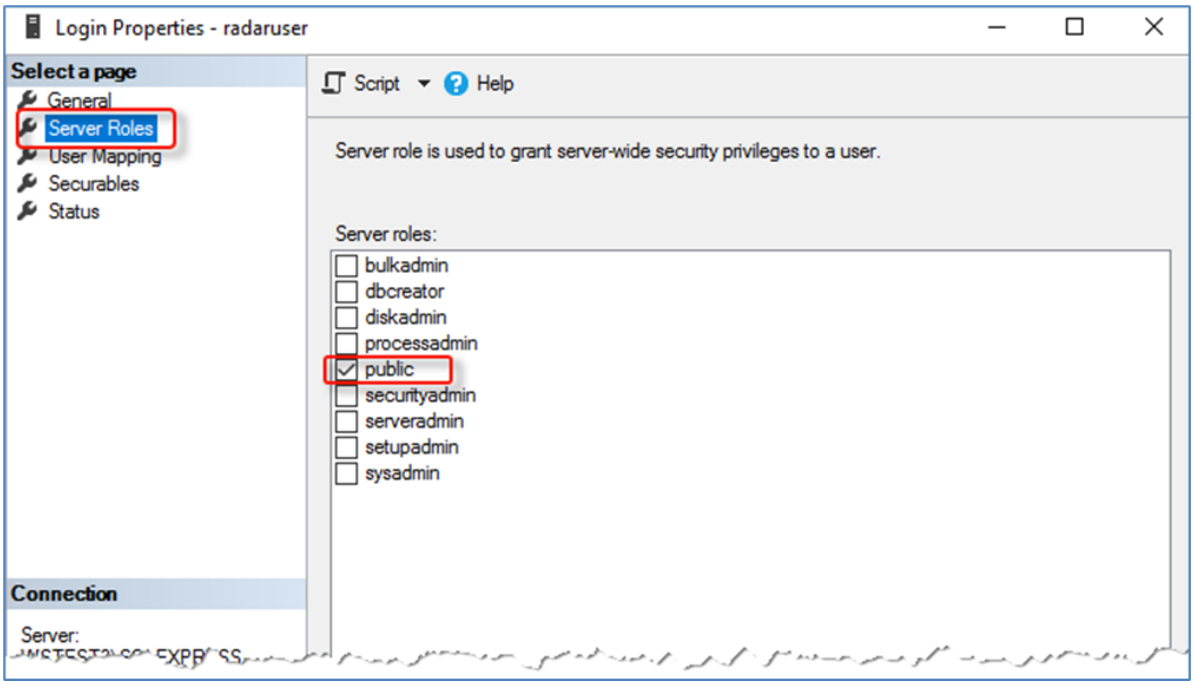

Рисунок 65 - Предоставление роли для создаваемого пользователя

5. В разделе User Mapping (Сопоставления пользователей) для созданного пользователя (radaruser) выполнить следующие настройки:
    - В поле User mapped to this login: (Пользователи, сопоставленные с этим именем для входа:) предоставить разрешение на подключение и чтение к БД *KAV*.
    - В поле Database role membership for: <имя БД> (Членство в роли базы данных для: <имя БД>) установить для выбранной БД роль *db_datareader* (см. рисунок 66).

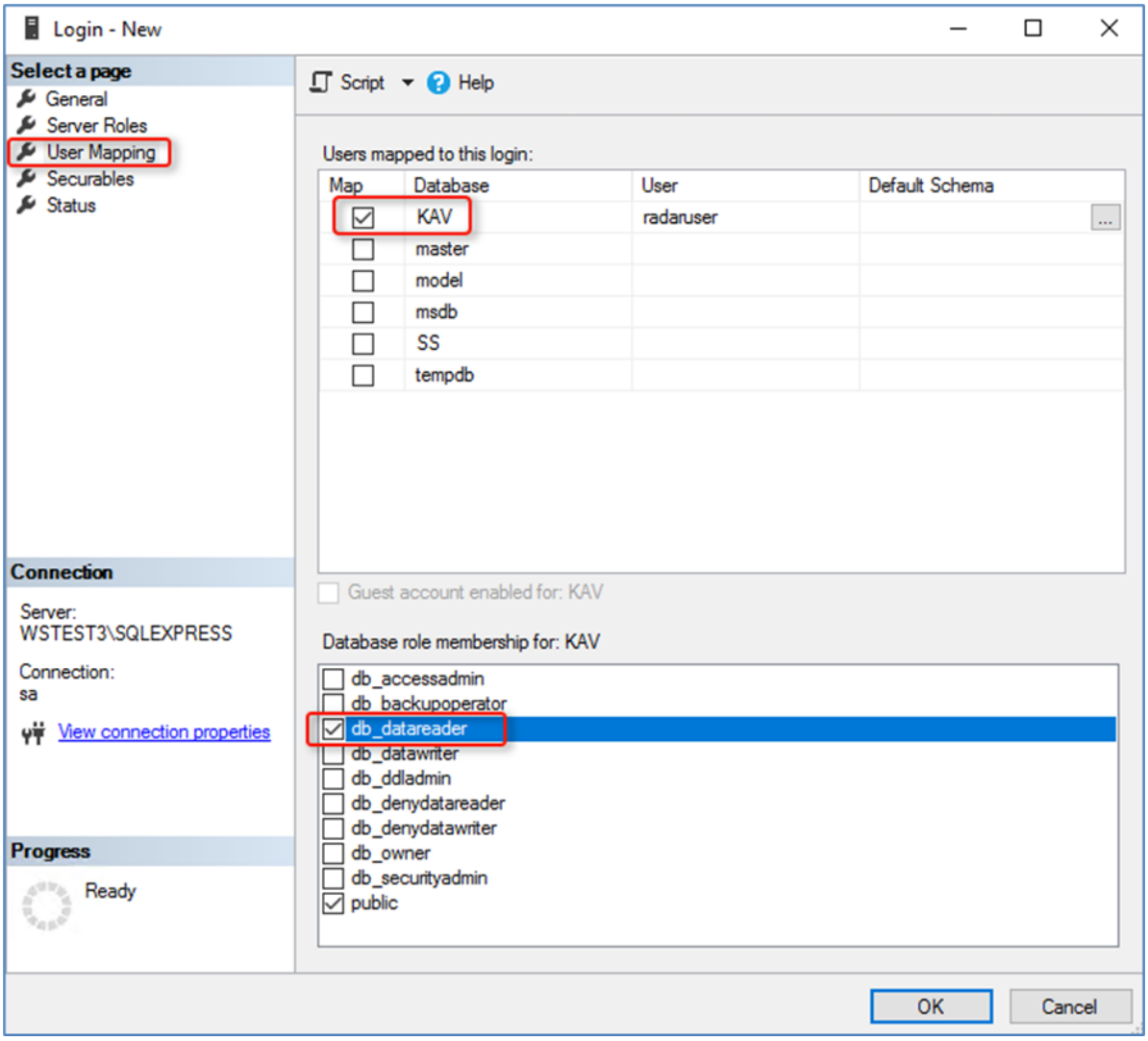

Рисунок 66 - Настройка прав доступа к БД KAV

6. В разделе Securables (Защищаемые объекты) для созданного пользователя (radaruser) установить для выбранного сервера СУБД следующие разрешения в области Permission for: <имя сервера СУБД> (Разрешения для: <имя сервера СУБД>):
    - *Connect SQL (подключение SQL)* (см. рисунок 67).

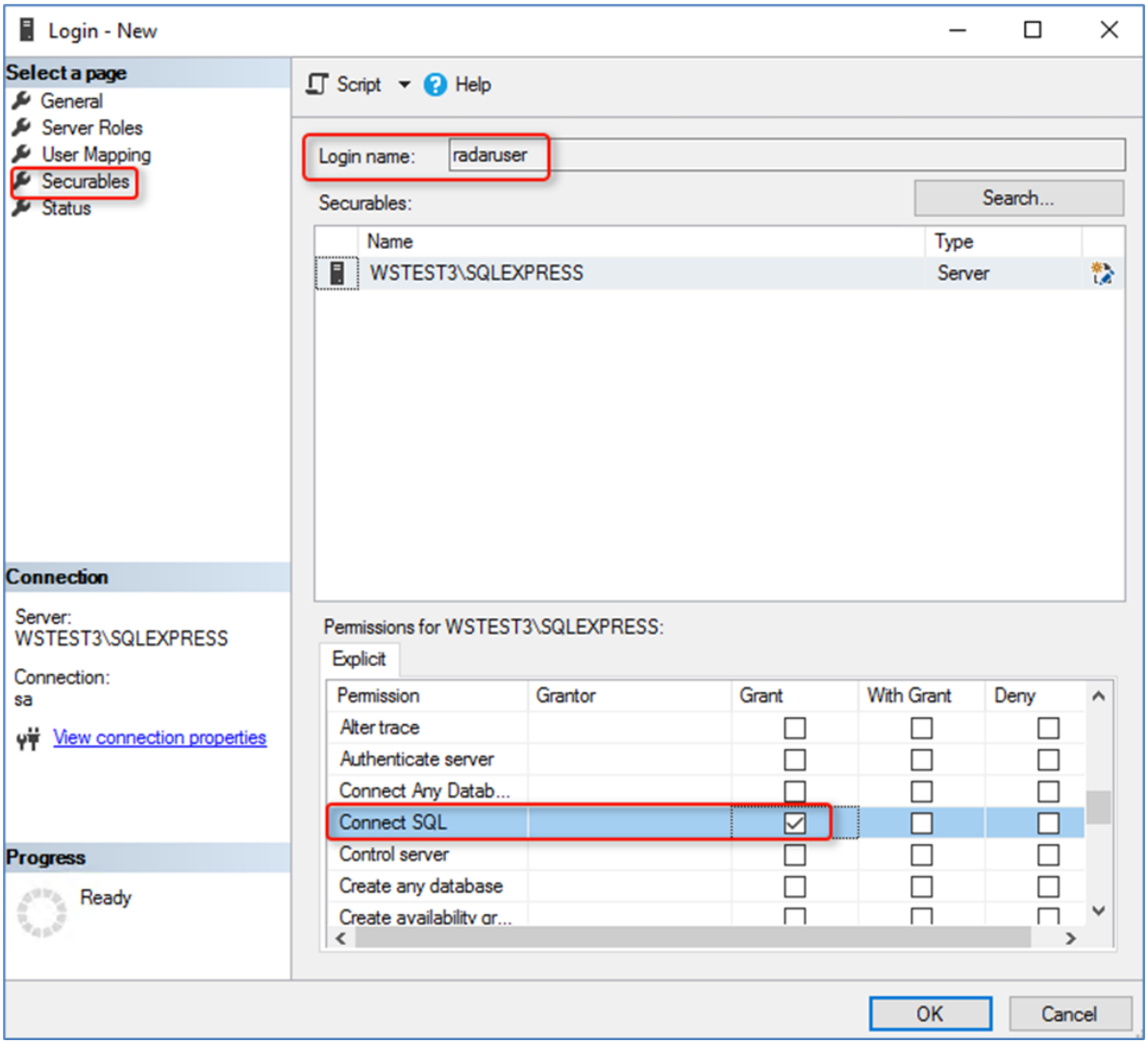

Рисунок 67 - Установка разрешения на подключение к БД

7. Для сохранения введенных настроек для подключения к экземпляру БД нажать кнопку ОК.

Создание пользователя в БД KAV. Для предоставления доступа к БД KAV выполнить следующие действия:

1. В окне Object Explorer (Обозреватель объектов) выбранного экземпляра БД (SQLEXPRESS) выбрать раздел (см. рисунок 68):

   Database → <Имя БД> → Security → Users

   (База данных → <Имя БД> → Безопасность → Пользователи).

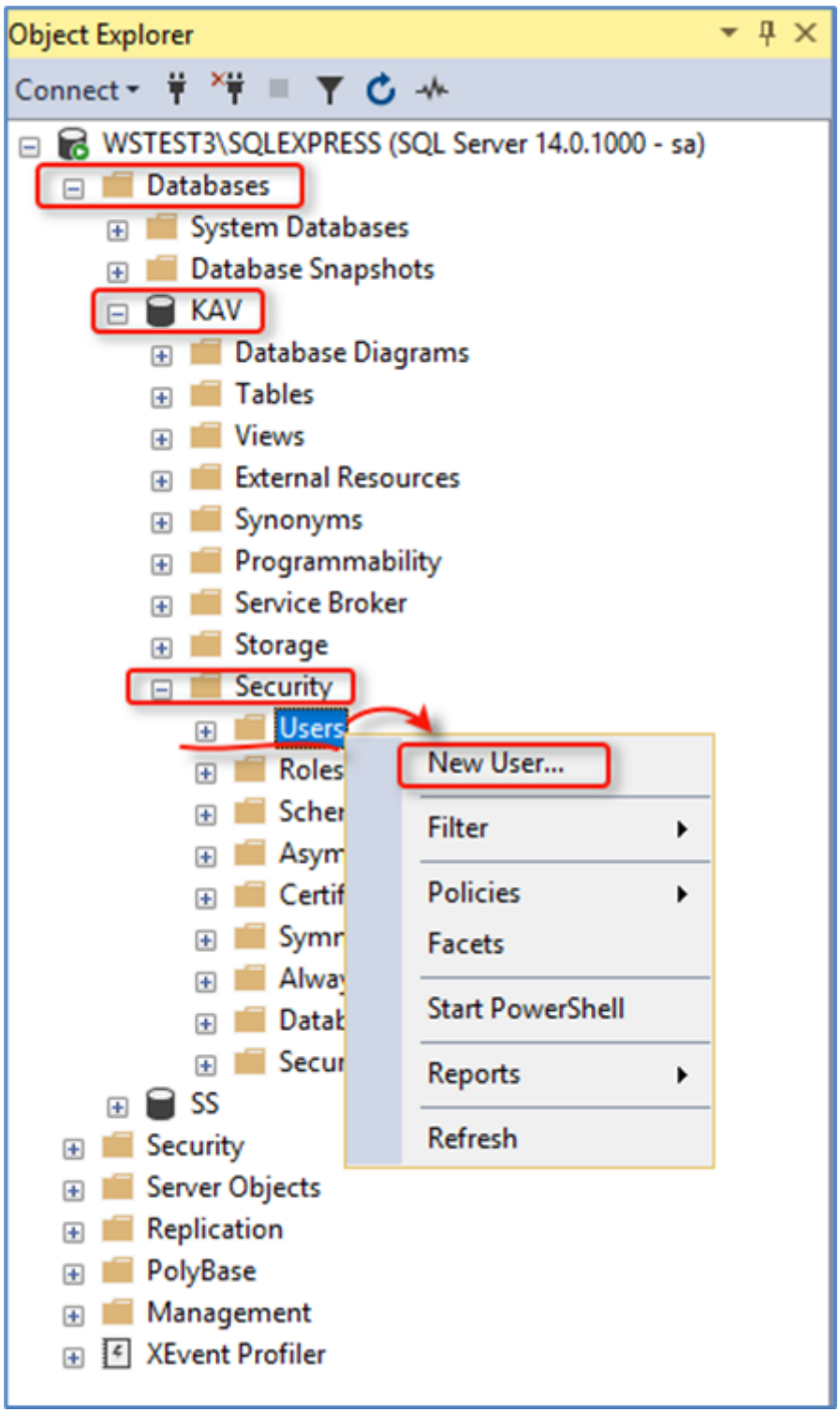

Рисунок 68 - Функция создания пользователя в БД KAV

2. Открыть контекстное меню раздела Users (Пользователи) и выбрать функцию New User (Создать пользователя - см. рисунок 68).
3. В открывшемся окне Database User - New (Пользователь базы данных - Создать) в разделе General (Общие) установить следующие параметры (см. рисунок 69):
    - в поле *User name (Имя пользователя)* установить имя пользователя (dbuser);
    - в поле *Login name (Имя для входа)* указать созданного выше (см. шаг 3) пользователя экземпляра БД (*radaruser*).

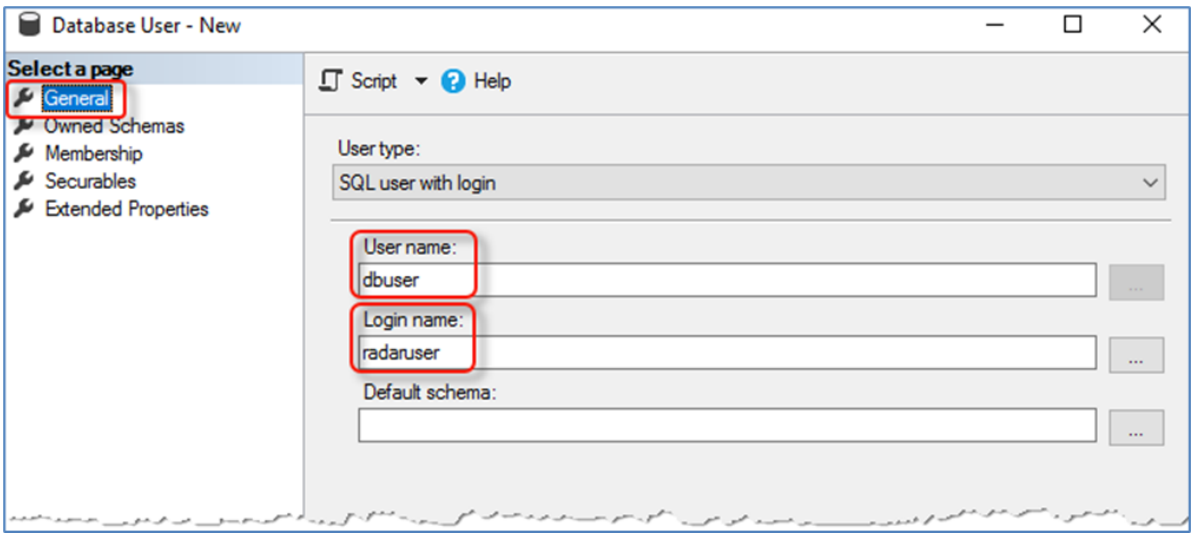

Рисунок 69 - Регистрация пользователя в БД KAV

4. В разделе Membership (Членство) установить для пользователя роль *db_datareader* (см. рисунок 70).

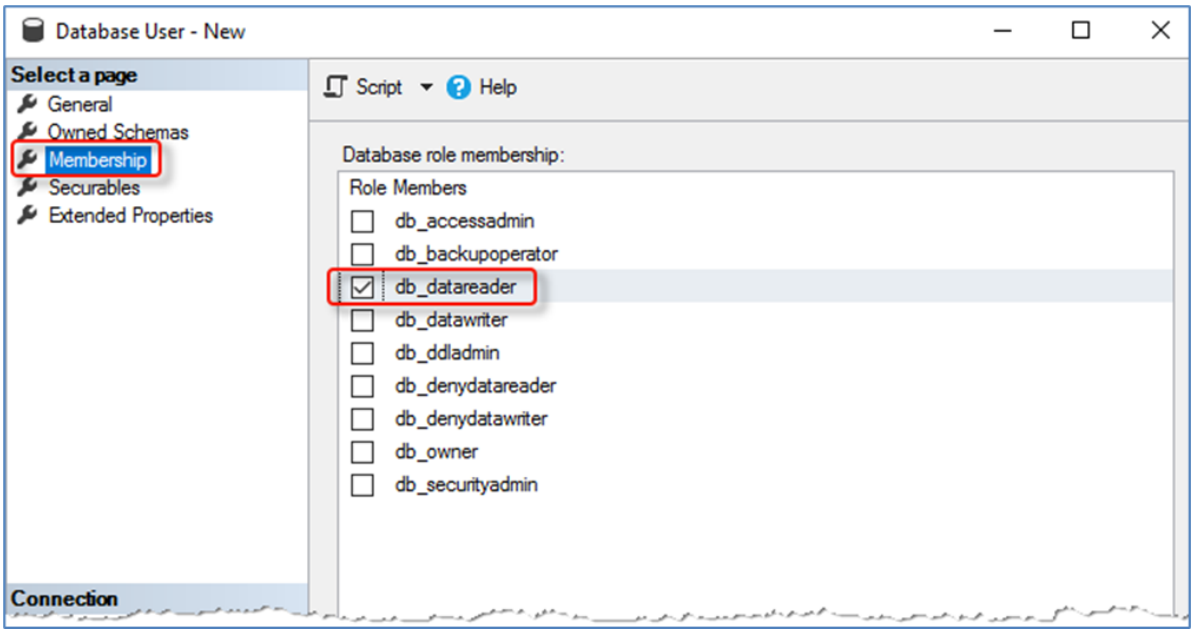

Рисунок 70 - Назначение роли

5. Для сохранения всех введенных настроек при создании пользователя в БД KAV нажать кнопку OK.

Предоставление удаленного сетевого доступа. Для удаленного доступа к данным, необходимо настроить доступность для выбранного экземпляра БД (*SQLEXPRESS*):

1. В меню Пуск необходимо запустить SQL Server Configuration Manager (Диспетчер конфигурации SQL Server).
2. В панели диспетчера конфигурации выбрать службу (см. рисунок 71):

   SQL Server Network Configuration → Protocols for SQLEXPRESS
   (Сетевая конфигурация SQL Server → Протоколы для SQLEXPRESS).
3. В открывшемся справа списке протоколов выбрать протокол TCP/IP и в контекстном меню протокола перевести подключение по данному протоколу в режим «Включено», установив статус *Enabled (Включено* - см. рисунок 71).

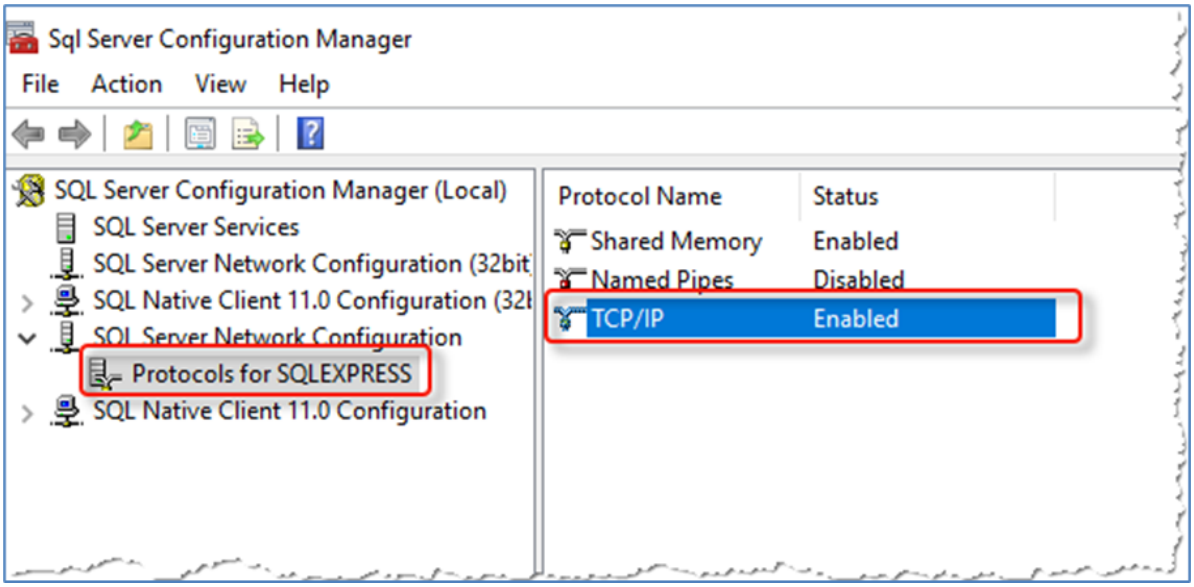

Рисунок 71 - Подключение по протоколу TCP/IP

4. В контекстном меню протокола TCP/IP выбрать функцию Properties (Свойства).
5. В открывшемся окне TCP/IP Properties (Свойства TCP/IP) на вкладке IP Adresses (IP-адреса) выбрать блок параметров *IPAll* и ввести значение порта в поле TCP Port. Например: 1433 (см. рисунок 72).
6. Нажать кнопку OK для сохранения настроек доступа по протоколу TCP/IP.

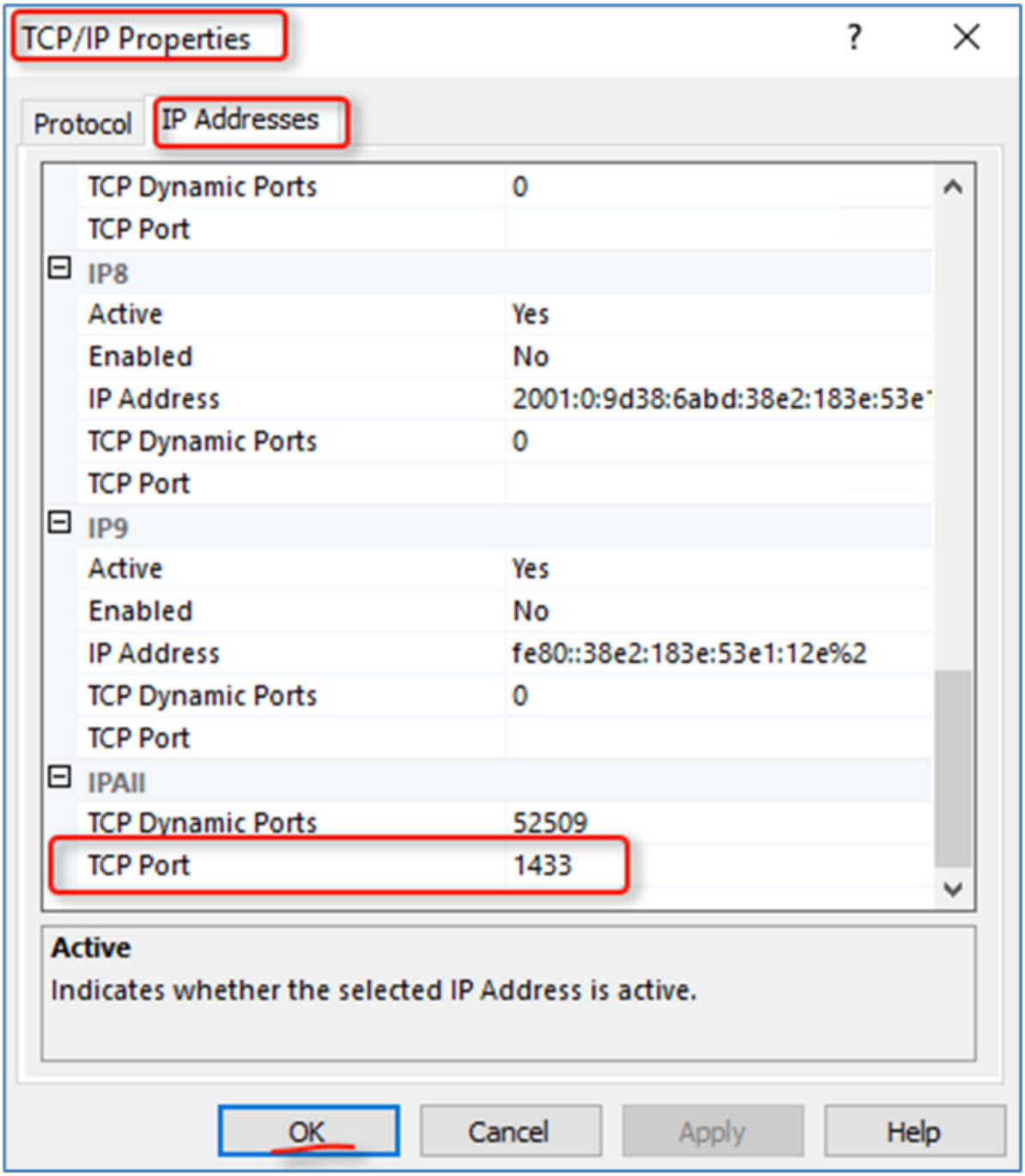

Рисунок 72 - Пример настройки протокола для удаленного доступа к БД

7. Для применения сетевых настроек необходимо перезапустить службу MS SQL Server:
    - В меню Пуск выбрать раздел Servise (Службы).
    - В открывшемся окне Службы (Службы) выбрать службу SQL Server с запущенным экземпляром БД (SQLEXPRESS).
    - Выбрать функцию Restart the servise (Перезапустить службу) (см. рисунок 73).

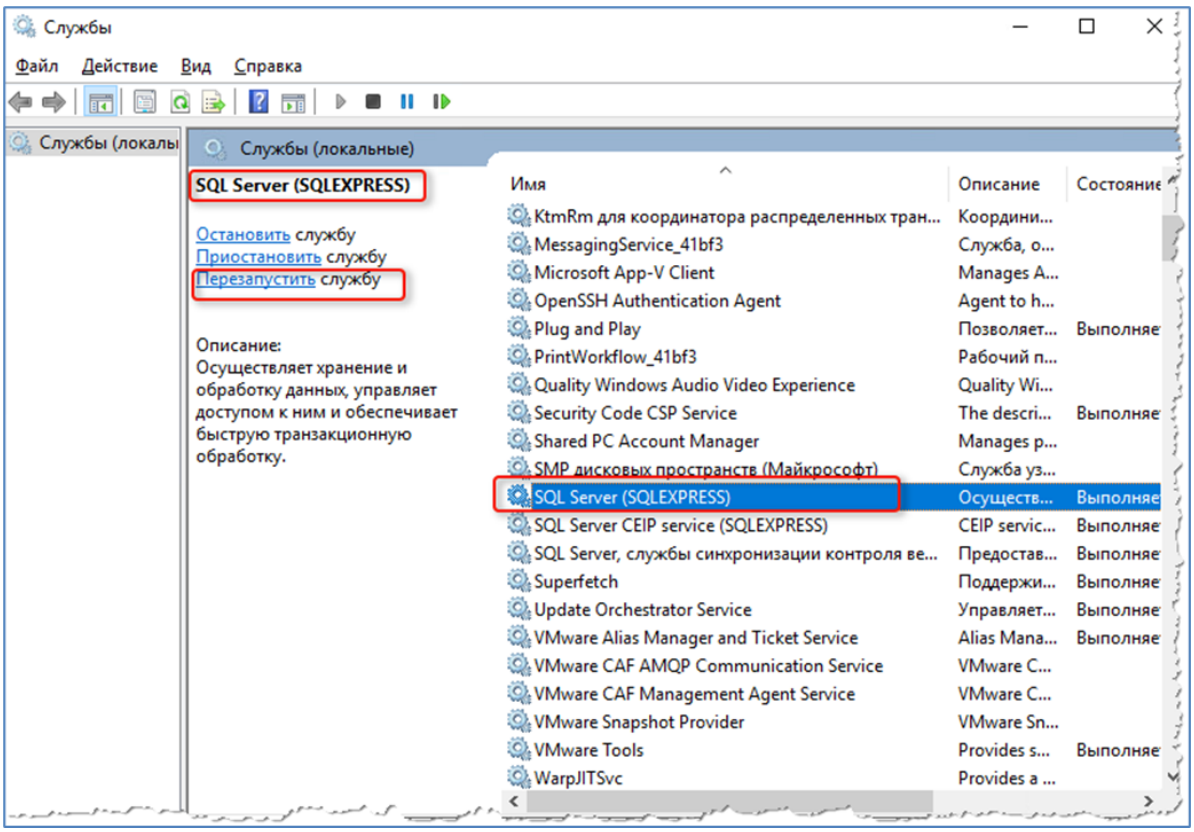

Рисунок 73 - Перезапуск службы MS SQL Server

6.2.5. SQL запрос для KSC {#sqlksc}

```
sql: >
    SELECT
      events.event_id AS event_id,
      events.nHostId AS host_id,
      events.severity AS severity,
      events.group_name AS group_name, event_type,
      events.event_type_display_name AS event_name,
      rise_time AS event_time,
      events.descr AS description,
      events.task_display_name AS task_name,
      events.task_id AS task_id,
      events.product_displ_version AS product_version,
      events.par1,
      events.par2,
      events.par3,
      events.par4,
      events.par5,
      events.par6,
      events.par7,
      events.par8,
      events.product_name,
      hosts_view.strDisplayName AS hostname,
      dnsdomains.strName AS domain,
      fqdns.wstrfqdn AS fqdn,
      CAST(((hosts.nIpAddress / 16777216) & 255) AS varchar(4)) + '.' +
      CAST(((hosts.nIpAddress / 65536) & 255) AS varchar(4)) + '.' +
      CAST(((hosts.nIpAddress / 256) & 255) AS varchar(4)) + '.' +
```

```
      CAST(((hosts.nIpAddress) & 255) AS varchar(4)) AS ip_address,
      hosts_view.nPlatformType AS platform_id,
      hosts_view.tmLastInfoUpdate AS last_update,
      hosts_view.nVirusCount AS virus_count
    FROM KAV.dbo.ev_event AS events
    JOIN KAV.dbo.Hosts AS hosts ON hosts.nId = events.nHostId
    JOIN KAV.dbo.v_hosts AS hosts_view ON hosts_view.nId = hosts.nId
    JOIN KAV.dbo.v_hst_fqdns AS fqdns ON fqdns.nId = hosts.nId
    RIGHT JOIN KAV.dbo.DnsDomains AS dnsdomains ON dnsdomains.nId =
hosts.nDnsDomain
    WHERE events.event_type IN (
      'FSEE_AKPLUGIN_CRITICAL_PATCHES_AVAILABLE',
      'FSEE_AKPLUGIN_PEP_APPLICATION_AUDIT_DENIED',
      'GNRL_EV_APP_LAUNCH_TESTED_DENIED',
      'GNRL_EV_APPLICATION_LAUNCH_DENIED',
      'GNRL_EV_ATTACK_DETECTED',
      'GNRL_EV_DEVCTRL_DEV_PLUGGED',
      'GNRL_EV_OBJECT_BLOCKED',
      'GNRL_EV_OBJECT_CURED',
      'GNRL_EV_OBJECT_DELETED',
      'GNRL_EV_OBJECT_NOTCURED',
      'GNRL_EV_OBJECT_QUARANTINED',
      'GNRL_EV_PTOTECTION_LEVEL_CHANGED',
      'GNRL_EV_SUSPICIOUS_OBJECT_FOUND',
      'GNRL_EV_VIRUS_FOUND',
      'GNRL_EV_VIRUS_OUTBREAK',
      'KLAUD_EV_ADMGROUP_CHANGED',
      'KLAUD_EV_SERVERCONNECT',
      'KLNAG_EV_INV_APP_INSTALLED',
      'KLNAG_EV_INV_APP_UNINSTALLED',
      'KLNAG_EV_INV_CMPTR_APP_INSTALLED',
      'KLPRCI_TaskState',
      'KLSRV_EVENT_HOSTS_CONFLICT',
      'KLSRV_EVENT_HOSTS_NEW_DETECTED',
      'KLSRV_HOST_STATUS_CRITICAL',
      'KLSRV_HOST_STATUS_WARNING',
      'KLSRV_SEAMLESS_UPDATE_REGISTERED',
      'KLSRV_UPD_BASES_UPDATED',
      '000000d1',
      '000000d3',
      '000000d4',
      '000000d5',
      '000000d6',
      '000000dd',
      '000000de',
      '000000df',
      '0000012f',
      '0000014d',
      '0000014e',
      '0000014f',
      '00000192',
      '00000193',
      '000000cf'
    )
    AND event_id > ?;
```

6.3. Kaspersky Security Center через MariaDB

Для подключения в качестве источника Kaspersky Security Center, работающего на базе данных MariaDB, необходимо выполнить несколько шагов:

1. Зайти в CMD MariaDB

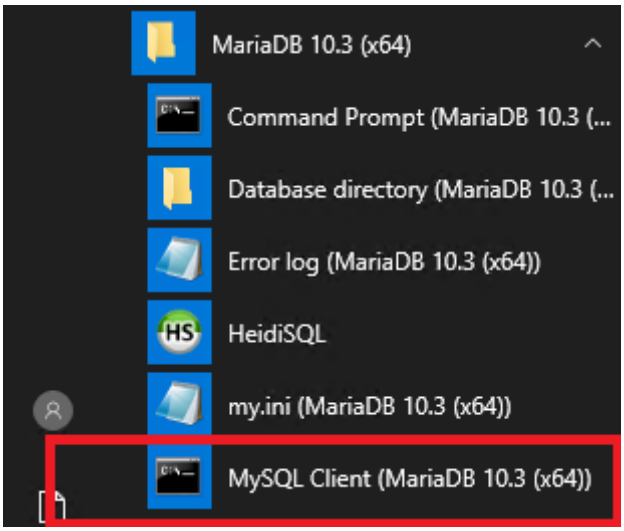

Рисунок 74 - Запуск CMD MariaDB.

2. Ввести пароль от БД (пароль задавался при установке MariaDB)

Рисунок 75 - Ввод пароля от БД.

3. Создать пользователя с правами удаленного подключения командой:

```
CREATE USER '<имя пользователя>'@'<ip-адрес лог-коллектора>' IDENTIFIED BY '<Пароль Пользователя>';
```

> Пример:
>
> `CREATE USER 'radar_reader'@'192.168.100.254' IDENTIFIED BY 'P@ssw0rd';`

4. Дать права на чтение определенных таблиц в базе антивируса Касперского, для этого ввести следующие команды по очереди (предварительно заменив <имя пользователя>,<ip-адрес лог-коллектора> и <Пароль Пользователя> на данные, указанные в пункте 3:

```
GRANT SELECT ON KAV.ev_event TO '<имя пользователя>'@'<ip-адрес лог-
коллектора>' IDENTIFIED BY '<Пароль Пользователя>';
GRANT SELECT ON KAV.dnsdomains TO '<имя пользователя>'@'<ip-адрес лог-
коллектора>' IDENTIFIED BY '<Пароль Пользователя>';
GRANT SELECT ON KAV.v_hst_fqdns TO '<имя пользователя>'@'<ip-адрес лог-
коллектора>' IDENTIFIED BY '<Пароль Пользователя>';
GRANT SELECT ON KAV.hosts TO '<имя пользователя>'@'<ip-адрес лог-
коллектора>' IDENTIFIED BY '<Пароль Пользователя>';
GRANT SELECT ON KAV.v_hosts TO '<имя пользователя>'@'<ip-адрес лог-
коллектора>' IDENTIFIED BY '<Пароль Пользователя>';
```

> Пример:
>
> ```
> GRANT SELECT ON KAV.ev_event TO 'radar_reader'@'192.168.100.254'
> IDENTIFIED BY 'P@ssw0rd';
> GRANT SELECT ON KAV.dnsdomains TO 'radar_reader'@'192.168.100.254'
> IDENTIFIED BY 'P@ssw0rd';
> GRANT SELECT ON KAV.v_hst_fqdns TO 'radar_reader'@'192.168.100.254'
> IDENTIFIED BY 'P@ssw0rd';
> GRANT SELECT ON KAV.hosts TO 'radar_reader'@'192.168.100.254'
> IDENTIFIED BY 'P@ssw0rd';
> GRANT SELECT ON KAV.v_hosts TO 'radar_reader'@'192.168.100.254'
> IDENTIFIED BY 'P@ssw0rd';
> ```

5. Зайти в Web-интерфейс **Платформы Радар** по пути "Кластер" -> "Узлы" -> Выбрать "ip-адрес вашего лог-коллектора" ->

   В разделе Секреты Агента нажать "Добавить"

   - Указать "Название секрета" и указать его "Значение секрета" для "Имени пользователя"

     > Пример:
     >
     > "Название секрета" - User_DB , "Значение секрета" - radar_reader

   - Указать "Название секрета" и указать его "Значение секрета" для "Пароля пользователя"

     > Пример:
     >
     > "Название секрета" - User_DB_pwd , "Значение секрета" - P@ssw0rd

6. Указать в конфигурационном файле лог-коллектора данные секретов в формате {{.User_DB}} - для пользователя и {{.User_DB_pwd}} для пароля, в строке "connection_string"

> Пример:
>
> ```
> connection_string: "server=192.168.100.253;port=3306;driver={MySQL ODBC
> 8.0 Unicode Driver};database=kav;user={{.User_DB}};password=
> {{.User_DB_pwd}};"
> ```

7. Перезапустить сервис лог-коллектора.

6.4. Kaspersky Security Center через Syslog

6.4.1. Настройка Kaspersky Security Center для экспорта событий в Платформу Радар

Вы можете включить автоматический экспорт событий в Kaspersky Security Center.

Только общие события могут быть экспортированы от управляемых программ в форматах CEF и LEEF. Специфические события программ не могут быть экспортированы от управляемых программ в форматах CEF и LEEF. Если необходимо экспортировать события управляемых программ или пользовательский набор событий, который настроен с помощью политик управляемых программ, используйте экспорт событий в формате Syslog.

Чтобы включить автоматический экспорт общих событий:

1. В дереве консоли Kaspersky Security Center выберите узел с именем Сервера администрирования, события которого необходимо экспортировать.
2. В рабочей области выбранного Сервера администрирования перейдите на закладку События.
3. Нажмите на стрелку рядом со ссылкой Настроить параметры уведомлений и экспорта событий и в раскрывающемся списке выберите пункт Настроить экспорт в SIEM-систему. Откроется окно свойств событий на разделе Экспорт событий.
4. В разделе Экспорт событий укажите следующие параметры экспорта:
    - Автоматически экспортировать события в базу SIEM-системы
    - SIEM-система
    - Адрес сервера SIEM-системы
    - Порт сервера SIEM-системы
    - Протокол
5. Если вы выбрали формат Syslog, вы должны указать:
    - Максимальный размер сообщения в байтах

Если требуется выполнить экспорт в **Платформу Радар** событий, произошедших после определенной даты в прошлом, нажмите на кнопку Экспортировать архив и укажите дату, начиная с которой будет выполнен экспорт событий. По умолчанию экспорт событий начинается сразу после включения.

Нажмите на кнопку ОК.

Автоматический экспорт событий включен.

После включения автоматического экспорта событий необходимо выбрать, какие события будут экспортироваться в **Платформу Радар**.

6.4.2. Выбор событий для экспорта в Платформу Радар в формате Syslog

После включения автоматического экспорта событий необходимо выбрать, какие события будут экспортироваться в **Платформу Радар**.

Вы можете настроить экспорт событий в формате Syslog в **Платформу Радар** на основе одного из следующих условий:

- Выбор общих событий. Если вы выберите экспортируемые события в политике, в свойствах события или в свойствах Сервера администрирования, то в **Платформу Радар** будут переданы выбранные события, которые произошли во всех программах, управляемых данной политикой. Если экспортируемые события были выбраны в политике, вам не удастся их переопределить для отдельной программы, управляемой этой политикой.
- Выбор событий для управляемой программы. Если вы выбираете экспортируемые события для управляемой программы, установленной на управляемых устройствах, то в **Платформу Радар** будут переданы только события, которые произошли в этой программе.

Если вы хотите выполнить экспорт событий, произошедших в отдельной управляемой программе, установленной на управляемом устройстве, выберите для программы события для экспорта. В случае, если ранее экспортируемые события были выбраны в политике, вам не удастся переопределить выбранные события для отдельной программы, управляемой этой политикой.

Чтобы выбрать события для отдельной управляемой программы:

1. В дереве консоли Kaspersky Security Center выберите узел Управляемые устройства и перейдите на закладку Устройства.
2. Откройте контекстное меню требуемого устройства по правой клавише мыши и выберите пункт Свойства.
3. В открывшемся окне свойств устройства выберите раздел Программы.
4. В появившемся списке программ выберите программу, события которой требуется экспортировать, и нажмите на кнопку Свойства.
5. В окне свойств программы выберите раздел Настройка событий.
6. В появившемся списке событий выберите одно или несколько событий, которые требуется экспортировать в **Платформу Радар**, и нажмите на кнопку Свойства.
7. В открывшемся окне свойств событий выберите параметр Экспортировать в SIEM-систему по протоколу Syslog, чтобы отметить выбранные события для экспорта в формате Syslog. Выключите параметр Экспортировать в SIEM-систему по протоколу Syslog, чтобы отменить выбор событий для экспорта в формате Syslog.
8. Если свойства события заданы в политике, поля этого окна недоступны для редактирования.
9. Нажмите на кнопку ОК, чтобы сохранить изменения.
10. Нажмите на кнопку ОК в окне свойств программы и в окне свойств устройства.

Если вы хотите выполнить экспорт событий, произошедших во всех программах, управляемых определенной политикой, выберите экспортируемые события в политике. В этом случае вы не можете выбрать события для отдельной управляемой программы.

Чтобы выбрать общие события для экспорта в **Платформу Радар**:

1. В дереве консоли Kaspersky Security Center выберите узел Политики.
2. Откройте контекстное меню требуемой политики по правой клавише мыши и выберите пункт Свойства.
3. В открывшемся окне свойств политики выберите раздел Настройка событий.
4. В появившемся списке событий выберите одно или несколько событий, которые требуется экспортировать в **Платформу Радар**, и нажмите на кнопку Свойства.
5. Если требуется выбрать все события, нажмите на кнопку Выделить все.

6. В открывшемся окне свойств событий выберите параметр Экспортировать в SIEM-систему по протоколу Syslog, чтобы отметить выбранные события для экспорта в формате Syslog. Снимите флажок Экспортировать в SIEM-систему по протоколу Syslog, чтобы отменить выбор событий для экспорта в формате Syslog.
7. Нажмите на кнопку ОК, чтобы сохранить изменения.
8. В окне свойств политики нажмите на кнопку ОК.

6.5. Настройка Kaspersky Anti Targeted Attack для отправки событий в Платформу Радар

Для настройки отправки событий Kaspersky Anti Targeted Attack в **Платформу Радар** выполните шаги:

1. Зайдите в веб-интерфейс системы Kaspersky Anti Targeted Attack под учетной записью с правами администратора системы.
2. Выберите последовательно пункты меню: Settings -> SIEM system
3. В открывшемся окне выполните настройки:
    - отметить чек-бокс напротив «Activity log» и «Alerts»;
    - заполнить имя хоста или IP-адрес лог-коллектора в поле «Host/IP»;
    - заполнить порт, открытый на лог-коллекторе для приема событий от данного источника, в поле «Port»;
    - выбрать протокол взаимодействия (TCP/UDP);
    - заполнить ID устройства в поле «Host ID»;
    - выбрать интервал отправки событий с информацией о состоянии системы;
    - установить переключатель в Enable в поле «TLS encryption» при необходимости шифрования отправки событий.
4. Для сохранения изменений нажмите «Apply» (см. рисунок 76).

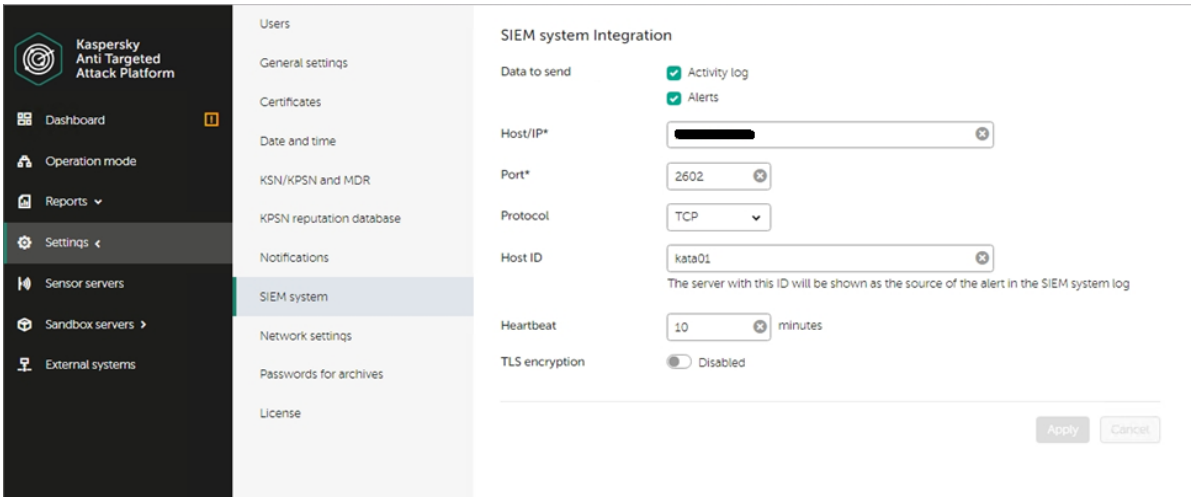

Рисунок 76 - Применение настройки отправки событий Kaspersky Anti Targeted Attack.

6.6. Kaspersky Web Traffic Security {#kwts}

> Данную инструкцию необходимо выполнить на каждом узле кластера KWTS.

Для настройки источника Kaspersky Web Traffic Security на отправку событийв **Платформу Радар** выполните следующие шаги:

1. Подключитесь к устройству Kaspersky Web Traffic Security с помощью интерфейса командной строки под пользователем root.

2. Создайте конфигурационный файл для rsyslog:

   `vim /etc/rsyslog.d/kwts_to_siem.conf`

3. Настройте отправку следующих объектов:

   `local0.*,local1.*,local2.*,authpriv.*,local7.* @@<Ip-адрес лог-коллектора>:<port>`

4. Перезапустите сервис rsyslog:

   `service rsyslog restart`

5. На машине с лог-коллектором добавьте изменения в файл конфигурации для сбора событий от источника и отправки их в **Платформу Радар**:

   - добавить Компонент сбора событий:

     ```
     tcp_input_kwts: & tcp_input_kwts
         id: "tcp_input_kwts"
       host: "0.0.0.0"
         port: <указать порт>
         sock_buf_size: 0
       format: "json"
         log_level: "INFO"
     ```

   - добавить Компонент отправки событий:

     ```
     tcp_output_kwts: & tcp_output_kwts
     id: "tcp_output_kwts"
     target_host: "<ip адрес  Платформы Радар/или балансера>"
     port: <указать порт>
     sock_buf_size: 0
     log_level: "INFO"
     ```

   - указать добавленные компоненты сбора и отправки в разделы collectors и senders соответственно:

     ```
     collectors:
         tcp_receiver:
         - <<: *tcp_input_kwts

     senders:
       port: 48002
         tcp:
         - <<: *tcp_output_kwts
     ```

   - добавить маршрут взаимодействия между компонентами сбора событий и компонентами отправки событий:

     ```
     route_1_kwts: &route_1_kwts
         collector_id:
         - "udp_input_kwts"
         sender_id:
         - "tcp_output_kwts"
     ```

- включить маршрут в разделе конфигурационного файла routers:

```
routers:
  - <<: *route_1_kwts
```

6. Перезапустите службу лог-коллектора
7. Включите или создайте источник KWTS в **Платформе Радар** и нажмите кнопку «Синхронизировать».
8. Проверьте поступающие события в **Платформе Радар** в разделе «Просмотр событий».

6.7. FireEye HX {#fireeye}

Для настройки источника FireEye HX на отправку событий в **Платформу Радар** выполните следующие шаги:

1. Подключитесь к устройству FireEye HX с помощью интерфейса командной строки.
2. Чтобы активировать режим конфигурации, введите поочередно следующие команды:

```
enable
configure terminal
```

3. Чтобы добавить назначение удаленного сервера системного журнала, введите следующие команды:

```
logging <IP_address лог-коллектора> trap none
logging <IP_address лог-коллектора> trap override class cef priority info
```

4. Чтобы сохранить изменения введите следующую команду:

`write mem`

5. На машине с лог-коллектором добавьте изменения в файл конфигурации для сбора событий от источника и отправки их в **Платформу Радар**:
   - добавить Компонент сбора событий

```
udp_input_FireEye: & udp_input_FireEye
    id: "udp_input_FireEye"
  host: "0.0.0.0"
    port: 514
    sock_buf_size: 0
  format: "json"
    log_level: "INFO"
```

   - добавить Компонент отправки событий

```
tcp_output_FireEye: & tcp_output_FireEye
id: "tcp_output_FireEye"
target_host: "<ip адрес  Платформы Радар/или балансера>"
port: 4560
sock_buf_size: 0
log_level: "INFO"
```

- указать добавленные компоненты сбора и отправки в разделы collectors и senders соответственно

```
collectors:
    udp_receiver:
    - <<: *udp_input_FireEye

senders:
  port: 48002
    tcp:
    - <<: *tcp_output_FireEye
```

- добавить маршрут взаимодействия между компонентами сбора событий и компонентами отправки событий

```
route_1_FireEye: &route_1_FireEye
    collector_id:
    - "udp_input_FireEye"
    sender_id:
    - "tcp_output_FireEye"
```

- включить маршрут в разделе конфигурационного файла routers

```
routers:
  - <<: *route_1_FireEye
```

6. Перезапустите службу лог-коллектора.
7. Включите источник FireEye-HX в **Платформе Радар** и нажмите кнопку «Синхронизировать».
8. Проверьте поступающие события в **Платформе Радар** в разделе «Просмотр событий».

7. Сетевые устройства.

7.1. Cisco IOS. System logging. {#ciscoios}

7.1.1. Настройка источника

1. Подключиться к консоли устройства;
2. Для включения логирования всех попыток подключения к устройству, введите команды:

   `(config)# service timestamps log datetime localtime show-timezone year`

   `(config)# logging userinfo`

   `(config)# login on-failure log`

   `(config)# login on-success log`
3. Для включения логирования изменений конфигурации, введите команды:

   `(config)# archive`

   `(config-archive)# log config`

   `(config-archive-log-cfg)# logging enable`

   `(config-archive-log-cfg)# notify syslog`

`(config-archive-log-cfg)# hidekeys`

4. Для отправки событий на коллектор, введите команды:

   `(config)# logging facility local5`

   `(config)# logging host <IP-адрес коллектора> transport tcp port <порт коллектора> (порт по умолчанию 2523)`

7.1.2. Включение источника на Платформе Радар

> Включение источника на **Платформе Радар** представлено в разделе *Управление источниками в Платформе, Включение/выключение поддерживаемых источников и их синхронизация*

1. Зайти в веб-консоль **Платформы Радар**, перейти в раздел «Источники», «Управление источниками»;
2. Найти в списке доступных источников (Cisco-IOSswitch) и включить его;
3. Кликнуть на кнопку «Синхронизировать».

7.1.3. Настройка коллектора событий

> Пример конфигурационного файла с настройкой данного источника представлен в разделе *Пример конфигурационного файла лог-коллектора*. Более подробная информация о настройках лог-коллектора представлена в разделе *Руководство по настройке лог-коллектора*

1. В конфигурационный файл лог-коллектора (config.yaml) необходимо добавить input компонента TCP.

   Пример настройки по умолчанию можно найти в документации: *Руководство по настройке лог-коллектора, Компонент TCP*

   Основные параметры, которые необходимо указать:

   `target_host: <"ip адрес или имя удаленного узла"> (адрес Платформы Радар)`

   `port: <"порт"> (стандартный порт для данного источника 2523)`

2. После настройки компонента сбора событий (input) - необходимо настроить компонент отправки событий (output).

   В качестве компонента отправки событий для данного источника предусмотрено использование отправки по протоколу TCP.

   Пример настройки по умолчанию можно найти в документации: *Руководство по настройке лог-коллектора, Компонент отправки событий по протоколу TCP*

   Основные параметры, которые необходимо указать:

   `target_host: <"ip адрес или имя удаленного узла"> (адрес Платформы Радар)`

   `port: <"порт"> (стандартный порт для данного источника 2523)`

3. Далее необходимо включить компоненты сбора (collectors) и отправки (senders).

   Пример настройки по умолчанию можно найти в документации: *Руководство по настройке лог-коллектора, Включение компонентов*

   Основные параметры, которые нужно указать при включении компонентов сбора:

   `collectors:`

   `tcp_receiver:`

`- <<: *<"id компонента сбора"> (ID компонента сбора, который указывали при объявлении компонента сбора)`

Основные параметры, которые нужно указать при включении компонентов отправки:

`senders:`

`tcp:`

`- <<: *<"id компонента отправки"> (ID компонента отправки, который указывали при объявлении компонента отправки)`

4. После чего необходимо произвести настройку маршрутизации событий.

   Пример настройки по умолчанию можно найти в документации: *Руководство по настройке лог-коллектора, Маршрутизация событий*

   Основные параметры, которые нужно указать при настройке маршрута:

   `route_1: &route_1`

   `collector_id:`

   `- <"id компонента сбора">`

   `sender_id:`

   `- <"id компонента отправки">`

5. После нужно включить маршрут в разделе routers. Пример включения маршрута:

   `routers:`

   `- <<: *<название маршрута> (например - <<: *route_1)`

7.2. Cisco IOS. Netflow v5. {#netflow}

7.2.1. Настройка источника

1. Подключиться к консоли устройства;
2. Для включения экспорта статистики сетевого трафика по протоколу NetFlow введите команды:

   `(config)# ip-flow-export destination <IP-адрес коллектора> <порт коллектора> (по умолчанию 2162)`

   `(config)# ip flow-export version 5`

   `(config)# interface <интерфейс, с которого необходимо собирать статистику>`

   `(config)# ip flow ingress`

   `(config)# ip flow egress`

7.2.2. Включение источника в Платформе Радар

> Включение источника в **Платформе Радар** представлено в разделе *Управление источниками в Платформе, Включение/выключение поддерживаемых источников и их синхронизация*

1. Зайти в веб-консоль **Платформы Радар**, перейти в раздел «Источники», «Управление источниками»;
2. Найти в списке доступных источников (Cisco-NetFlow) и включить его;
3. Кликнуть на кнопку «Синхронизировать».

7.2.3. Настройка коллектора событий

> Пример конфигурационного файла с настройкой данного источника представлен в разделе *Пример конфигурационного файла лог-коллектора*. Более подробная информация о настройках лог-коллектора представлена в разделе *Руководство по настройке лог-коллектора*

1. В конфигурационный файл лог-коллектора (config.yaml) необходимо добавить input компонента NetFlow.

   Пример настройки по умолчанию можно найти в документации: *Руководство по настройке лог-коллектора, Компонент NetFlow*

   Основные параметры, которые необходимо указать:

   `host: "<ip адрес лог-коллектора>" (адрес на котором запущен коллектор)`

   `port: <порт для приема соединений> (порт, на который будут приниматься события, если при настройке источника оставили стандартный - 2162)`

2. После настройки компонента сбора событий (input) - необходимо настроить компонент отправки событий (output).

   В качестве компонента отправки событий для данного источника предусмотрено использование отправки по протоколу TCP.

   Пример настройки по умолчанию можно найти в документации: *Руководство по настройке лог-коллектора, Компонент отправки событий по протоколу TCP*

   Основные параметры, которые необходимо указать:

   `target_host: <"ip адрес или имя удаленного узла"> (адрес Платформы Радар)`

   `port: <"порт"> (стандартный порт для данного источника 2162)`

3. Далее необходимо включить компоненты сбора (collectors) и отправки (senders).

   Пример настройки по умолчанию можно найти в документации: *Руководство по настройке лог-коллектора, Включение компонентов*

   Основные параметры, которые нужно указать при включении компонентов сбора:

   `collectors:`

   `nf_receiver:`

   `- <<: *<"id компонента сбора"> (ID компонента сбора, который указывали при объявлении компонента сбора)`

   Основные параметры, которые нужно указать при включении компонентов отправки:

   `senders:`

   `tcp:`

   `- <<: *<"id компонента отправки"> (ID компонента отправки, который указывали при объявлении компонента отправки)`

4. После чего необходимо произвести настройку маршрутизации событий.

   Пример настройки по умолчанию можно найти в документации: *Руководство по настройке лог-коллектора, Маршрутизация событий*

   Основные параметры, которые нужно указать при настройке маршрута:

   `route_1: &route_1`

`collector_id:`

`- <"id компонента сбора">`

`sender_id:`

`- <"id компонента отправки">`

5. После нужно включить маршрут в разделе routers. Пример включения маршрута:

   `routers:`

   `- <<: *<название маршрута> (например - <<: *route_1)`

7.3. D-link xStack {#dlinkxstack}

Для настройки D-link xStack на отправку событий в **Платформу Радар** выполните следующие шаги:

1. В веб-интерфейсе (Web-based User Interface) D-link Nexus перейдите по пути:

   `Administration > System Log > System Log Host`

2. Откройте System Log Server или System Log Server-Add
3. Установите следующие значения в каждом поле (см. рисунок 77):
   - В поле Index установите значение 1. Если устройство уже настроено на отправку на другие серверы, то выберите свободный ключ в диапазоне (1-4).
   - В поле Server IP укажите <ip-адрес лог-коллектора>
   - В поле Severity установите значение ALL
   - В поле Facility установите значение 4 (security/authorization messages)
   - В поле UDP Port укажите <номер-порта> (по умолчанию 514)
   - В поле Status установите Enabled
   - Нажмите кнопку Apply

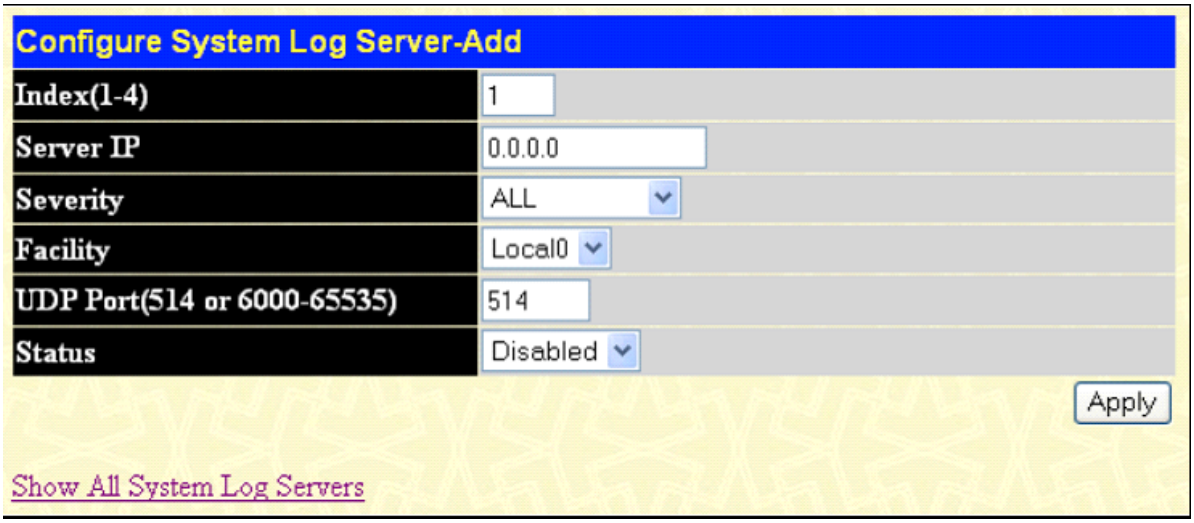

Рисунок 77 - Пример окна настройки добавления отправки событий.

4. В конфигурационном файле лог-коллектора добвьте настройку для получения событий от источника и отправки их в **Платформу Радар**

```
#####################################################################
            Часть настройки лог-коллектора
#####################################################################
#  Так как в 3м пункте был выбран шаблон отправки по UDP, поэтому настройка
на лог-коллекторе соответствует протоколу UDP
```

```
udp_input: & udp_input
    id: "udp_input"
  host: "0.0.0.0"
    port: 514
    sock_buf_size: 0
  format: "json"
    log_level: "INFO"

    tcp_output: &  tcp_output
  id: "tcp_output"
    target_host: "<ip адрес Платформы Радар>"
  port: 2773
    sock_buf_size: 0
    log_level: "INFO"

collectors:
    udp_receiver:
    - <<: *udp_input

senders:
  port: 48002
    tcp:
    - <<: *tcp_output

route_1: &route_1
    collector_id:
    - "udp_input"
    sender_id:
    - "tcp_output"
routers:
  - <<: *route_1
```

5. В **Платформе Радар** включите «Тип Источника» «Dlink xStack» и нажмите кнопку «Синхронизировать»
6. Проверьте приходящие события в **Платформе Радар**.

7.4. Коммутаторы Huawei {#huawei}

В разделе описана настройка источников Huawei S Series Switch , Huawei AR Series Router, Huawei USG Series Firewall.

Для настройки выполните шаги:

1. Войдите в интерфейс командной строки (CLI) маршрутизатора Huawei серии AR, коммутатора Huawei серии S или межсетевого экрана серии USG.
2. Введите команду для доступа к системному представлению:

   `system-view`
3. Введите команду, чтобы включить информационный центр:

   `info-center enable`
4. Введите команду для отправки сообщений информационного уровня на канал по умолчанию:

`info-center source default channel loghost log level informational debug state off trap state off`

5. Проверьте исходную конфигурацию маршрутизатора Huawei серии AR, коммутатора Huawei серии S или межсетевого экрана серии USG, введите команду:

   `display channel loghost`

6. Введите команду, чтобы настроить IP-адрес лог-коллектора в качестве хоста журнала для вашего коммутатора:

   `info-center loghost <IP-address log-collector> facility <local>`

   Где:
   `<IP-адрес>` — это IP-адрес лог-коллектора
   `<local>` — средство системного журнала, например, local0

   Например,

   `info-center loghost <IP-address log-collector> facility local0`

7. Введите команду, чтобы выйти из конфигурации:

   `quit`

8. Системы защиты электронной почты

8.1. FortiSandbox {#fortisandbox}

Для настройки FortiSandbox на отправку событий в **Платформу Радар** выполните следующие шаги:

1. Зайдите в веб-интерфейс системы под учетной записью с правами администратора системы.
2. Выберите последовательно пункты меню: «Log&Reports» -> «Log Servers» (см. рисунок 78).

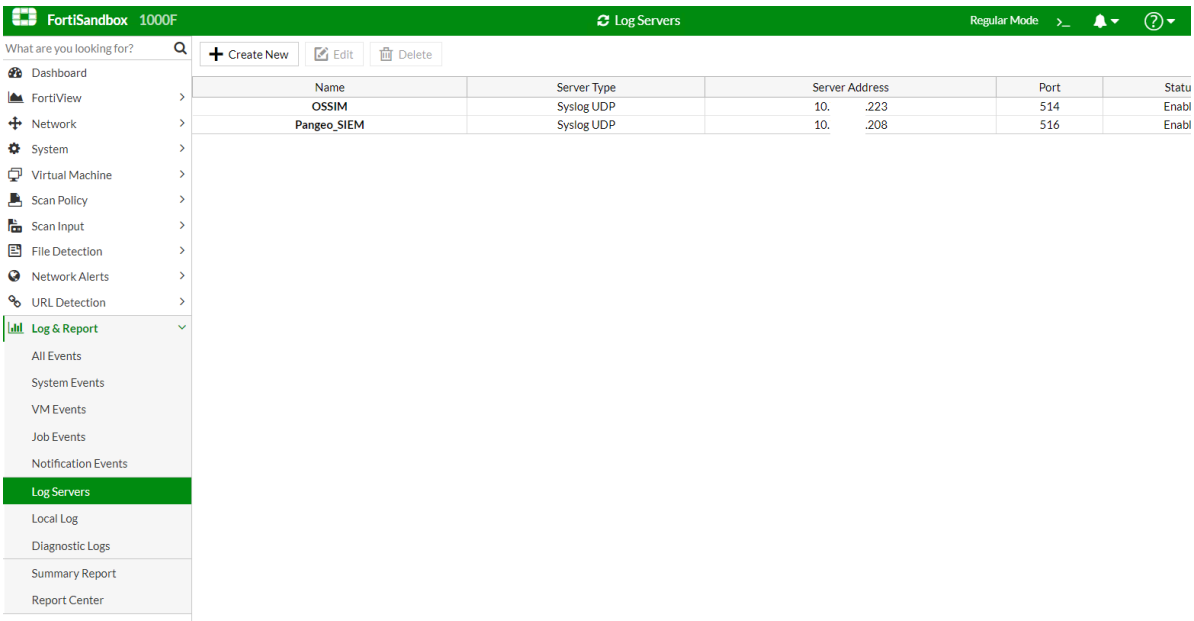

Рисунок 78 - Выбор логов.

3. Нажмите «Create New» и в открывшемся окне внесите следующие настройки:
   - заполнить название в поле «Name»;
   - выбрать протокол взаимодействия и формат отправки событий в поле «Type»;
   - заполнить IP-адрес лог-коллектора в поле «Log Server Address»;

- заполнить порт, открытый на лог-коллекторе для приема событий от данного источника, в поле «Port»;
- для включения отправки выбрать «Enable» в поле «Status»;
- выбрать уровень логирования отправляемых событий, проставив чекбоксы напротив соответствующих полей «Alert Logs», «Critical Logs» и т.п.

4. Нажмите «OK» для сохранения изменений (см. рисунок 79).

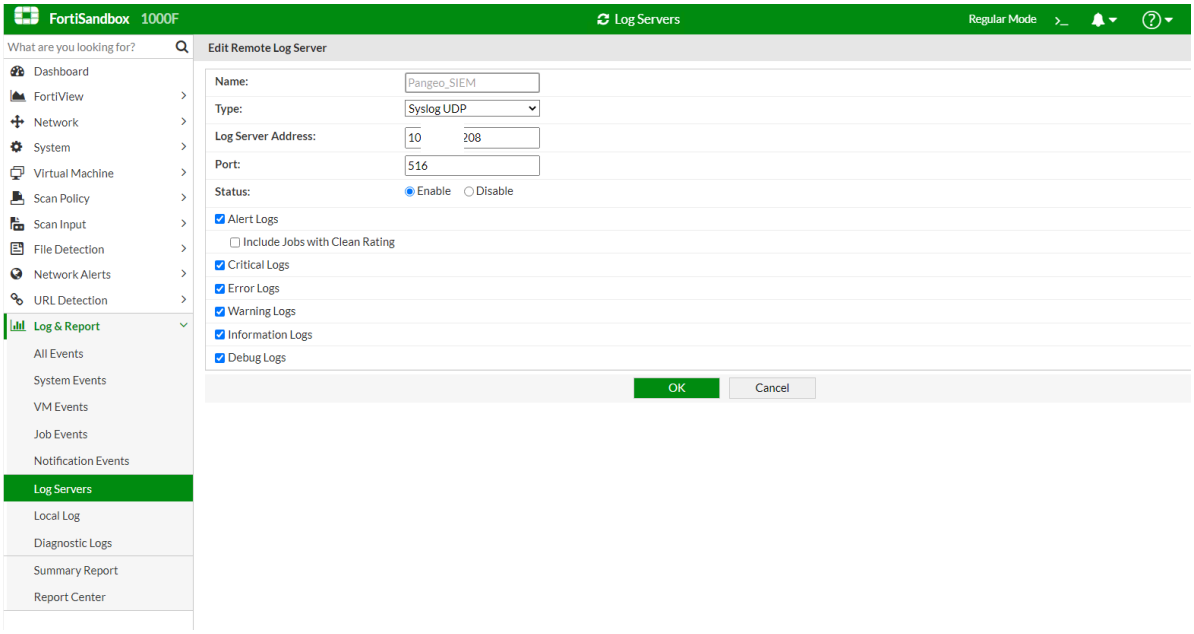

Рисунок 79 - Сохранение изменений.

8.2. Microsoft Exchange Server {#mes}

Данное руководство описывает механизм сбора событий MS Exchange версий 2013/2016/2019 и отправки их в **Платформу Радар**.

Предполагается, что для анализа и корреляции будут собираться следующие данные:

- OWA (IIS) logs.
- SMTP protocol logs
- Message tracking logs
- Exchange CosmosQueue Logs (Audit logs)

8.2.1. Настройка сбора OWA (IIS) logs

Расположение по умолчанию: C:\inetpub\logs\LogFiles\W3SVC1; C:\inetpub\logs\LogFiles\W3SVC2

После развертывания Exchange начинает писать эти логи автоматически и в большинстве случаев никаких дополнительных настроек этого типа источника не требуется.

8.2.2. Настройка SMTP protocol logs

Расположение по умолчанию: C:\Program Files\Microsoft\Exchange Server\V15\TransportRoles\Logs\FrontEnd\ProtocolLog

Включение ведения журнала протокола SMTP с помощью Центра администрирования Exchange

1. Откройте консоль Exchange Administration Center.
2. Перейдите во вкладку Mail Flow > Receive Connectors (см. рисунок 80).

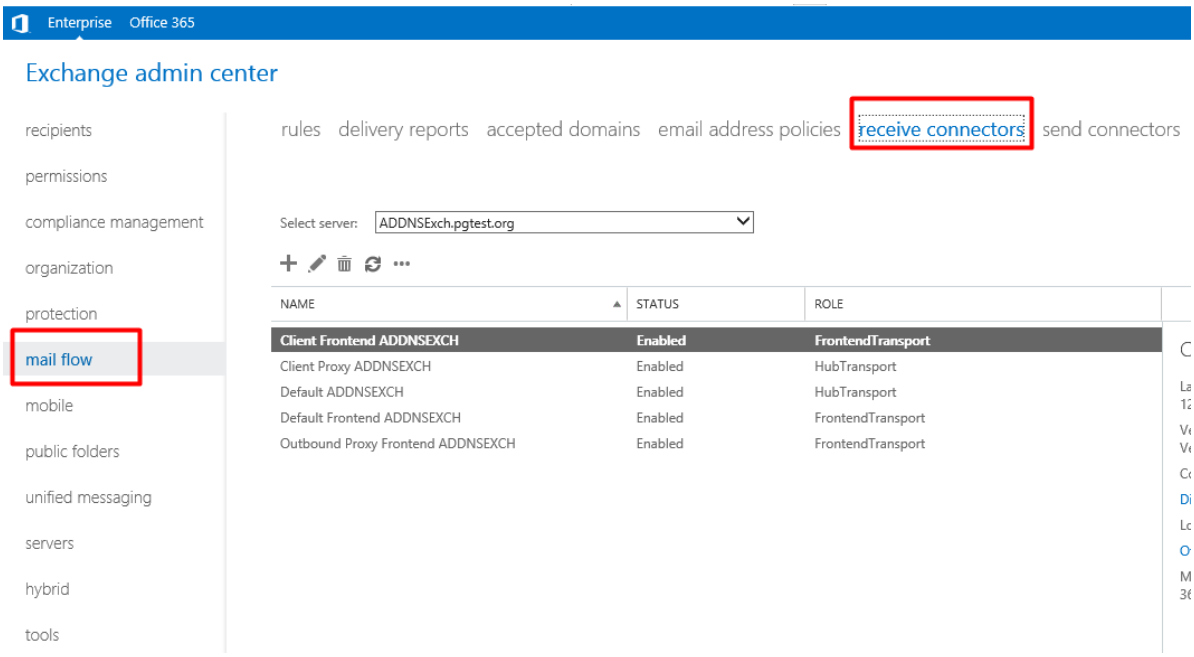

Рисунок 80 - Вкладка "Mail Flow".

3. Выберите нужный коннектор и нажмите Edit.
4. Перейдите во вкладку General
5. В поле Protocol logging level list выберите Verbose (см. рисунок 81).

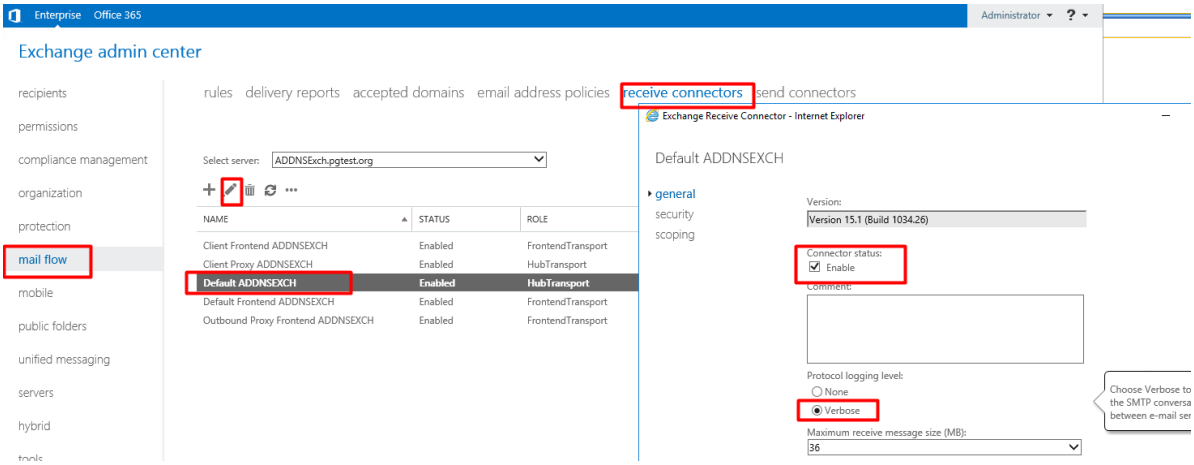

Рисунок 81 - Настройка параметров логирования.

6. Сохраните изменения, нажав кнопку Save.

8.2.3. Настройка Message tracking logs

Расположение по умолчанию: C:\Program Files\Microsoft Exchange Server\V15\TransportRoles\Logs\MessageTracking

Настройка отслеживания сообщений на серверах почтовых ящиков с помощью Центра администрирования Exchange

1. Откройте консоль Exchange Administration Center.
2. Перейдите в раздел Servers,выберите почтовый сервер, который нужно настроить, и нажмите кнопку Edit (см. рисунок 82).

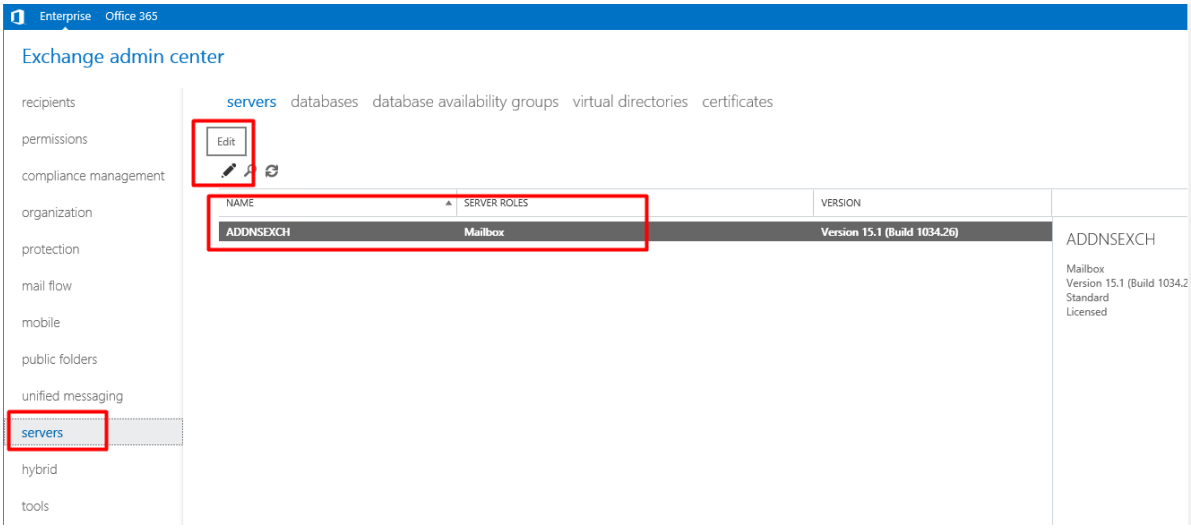

Рисунок 82 - Выбор почтового сервера.

3. На странице свойств сервера кликните на Transport logs. В разделе Message tracking log измените следующие параметры (см. рисунок 83):
    - Enable message tracking log. Чтобы отключить отслеживание сообщений на сервере, снимите флажок. Чтобы включить отслеживание сообщений на сервере, установите этот флажок.
    - Message tracking log path. Указанное значение должно находиться на локальном сервере Exchange Server. Если папка не существует, она будет создана после нажатия кнопки Сохранить.

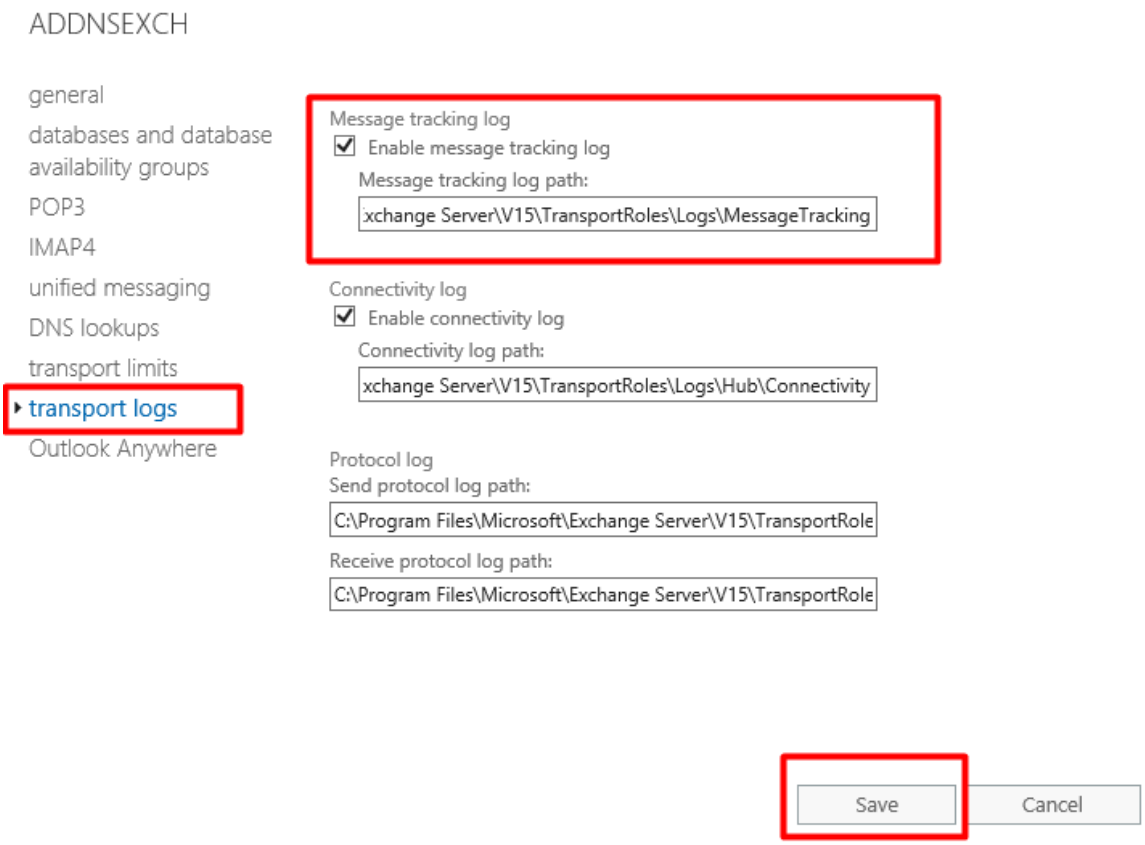

Рисунок 83 - Настройка логирования.

4. Сохраните изменения, нажав кнопку Save.

8.2.4. Настройка Exchange CosmosQueue Logs (Audit logs)

Расположение: C:\Program Files\Microsoft\Exchange Server\V15\Logging\CosmosQueue

После развертывания Exchange начинает писать эти логи автоматически и в большинстве случаев никаких дополнительных настроек этого типа источника не требуется.

8.2.5. Настройка лог-коллектора

Предпочтительным способом сбора данных является установка лог коллектора на серверах Exchange, поскольку в данном случае не придется открывать сетевой доступ к каталогам с логами. По возможности используйте только этот способ как наиболее безопасный.

Альтернативный способ - открыть сетевой доступ к каталогам с логами и настроить удаленный лог коллектор для сбора данных из сетевой шары.

Первый способ:

1. Установите лог-коллектор на Exchange сервер согласно инструкции.
2. Настройте лог-коллектор. Пример конфигурационного файла приведен ниже. Убедитесь, что файлы логов находятся по путям, указанным в лог коллекторе. Для примера приведены стандартные пути к логам, но если администратор изменил местонахождение файлов, исправьте пути в файле конфигурации.
3. При необходимости откройте необходимые порты на межсетевом экране (порты указаны в файле конфигурации).
4. Запустите службу лог-коллектора.
5. Проверьте наличие событий в интерфейсе **Платформы Радар**.

Пример файла конфигурации:

В данном примере указаны логин/пароль в открытом виде для наглядности, но есть возможность маскировать учетные данные в файле конфигурации с помощью защищенного хранилища (см. инструкцию к лог коллектору).

```
license_path: "C:/Program Files/Log Collector//pgr-agent.lic"
cluster:
  url: "https://адрес_сервера"
  api_key: "api_key"
controller:
  port: 48000
metric_server:
  port: 48005
secret_file: "C:/Program Files/Log Collector/secret"
secret_storage: "C:/Program Files/Log Collector/secret_storage"
api_server:
  address: "server ip or address"
  port: 8080
  read_timeout: 60
  write_timeout: 60
  wait: 5
  enable_tls: false
  cert_file: "C:/Program Files/Log Collector/certs/server.crt"
  key_file: "C:/Program Files/Log Collector/certs/server.key"
  cert_key_pass: ""
  require_client_cert: false
```

```
  ca_file: "C:/Program Files/Log Collector/certs/ca.crt"
  log_level: "WARN"
journal:
  port: 48004
  log_level: "INFO"
  log_path: "C:/Program Files/Log Collector/journal.log"
  rotation_size: 30
  max_backups: 7
  max_age: 7

owa_logs: &owa_logs
  id: "owa_logs"
  poll_interval: 1
  files: ["C:\\\\inetpub\\logs\\LogFiles"]

  using_regexp: true
  regexp_starting_dir: "c://inetpub/logs/LogFiles/"
  regexp_expression: ".log$"

  dir_check_interval: 2
  read_from_last: true

  enable_watcher: true
  log_level: "INFO"
  format: "json"
  encoding:
    change_to_utf8: false
    original_encoding: "cp1251"
  filters:
    blacklist:
      - "^#"

smtp_logs: &smtp_logs
  id: "smtp_logs"
  poll_interval: 1
  files: ["C:\\\\Program\ Files\\Microsoft\\Exchange\
Server\\V15\\TransportRoles\\Logs\\FrontEnd\\ProtocolLog"]

  using_regexp: true
  regexp_starting_dir: "c://Program Files/Microsoft/Exchange
Server/V15/TransportRoles/Logs/FrontEnd/ProtocolLog/"
  regexp_expression: ".LOG$"

  dir_check_interval: 2
  read_from_last: true

  enable_watcher: true
  log_level: "INFO"
  format: "json"
  encoding:
    change_to_utf8: false
    original_encoding: "cp1251"
  filters:
    blacklist:
      - "^#"
```

```
message_tracking: &message_tracking
  id: "message_tracking"
  poll_interval: 1
  files: ["C:\\\\Program\ Files\\Microsoft\\Exchange\
Server\\V15\\TransportRoles\\Logs\\MessageTracking"]

  using_regexp: true
  regexp_starting_dir: "C://Program Files/Microsoft/Exchange
Server/V15/TransportRoles/Logs/MessageTracking"
  regexp_expression: ".LOG$"

  dir_check_interval: 2
  read_from_last: true

  enable_watcher: true
  log_level: "INFO"
  format: "json"
  encoding:
    change_to_utf8: false
    original_encoding: "cp1251"
  filters:
    blacklist:
      - "^#"

audit_log: &audit_log
  id: "audit_log"
  poll_interval: 1
  files: ["C:\\\\Program\ Files\\Microsoft\\Exchange\
Server\\V15\\Logging\\CosmosQueue"]

  using_regexp: true
  regexp_starting_dir: "C://Program Files/Microsoft/Exchange
Server/V15/Logging/CosmosQueue"
  regexp_expression: ".LOG$"

  dir_check_interval: 2
  read_from_last: true

  enable_watcher: true
  log_level: "INFO"
  format: "json"
  encoding:
    change_to_utf8: false
    original_encoding: "cp1251"
  filters:
    blacklist:
      - "^#"

tcp_output_owa: &tcp_output_owa
  id: "tcp_output_owa"
  target_host: "pangeo_server_address"
  port: 1530
tcp_output_smtp: &tcp_output_smtp
  id: "tcp_output_smtp"
```

```
  target_host: "pangeo_server_address"
  port: 1531
tcp_output_message_tracking: &tcp_output_message_tracking
  id: "tcp_output_message_tracking"
  target_host: "pangeo_server_address"
  port: 1532
tcp_output_audit_log: &tcp_output_audit_log
  id: "tcp_output_audit_log"
  target_host: "pangeo_server_address"
  port: 1533

senders:
  port: 48002
  tcp:
    - <<: *tcp_output_owa
    - <<: *tcp_output_smtp
    - <<: *tcp_output_message_tracking
    - <<: *tcp_output_audit_log
collectors:
  event_log:
    - <<: *eventlog_collector
  files:
    - <<: *owa_logs
    - <<: *smtp_logs
    - <<: *message_tracking
    - <<: *audit_log

route_owa_logs: &route_owa_logs
  collector_id:
    - "owa_logs"
  sender_id:
    - "tcp_output_owa"
route_smtp_logs: &route_smtp_logs
  collector_id:
    - "smtp_logs"
  sender_id:
    - "tcp_output_smtp"
route_message_tracking: &route_message_tracking
  collector_id:
    - "message_tracking"
  sender_id:
    - "tcp_output_message_tracking"
route_audit_log: &route_audit_log
  collector_id:
    - "audit_log"
  sender_id:
    - "tcp_output_audit_log"
routers:
  - <<: *route_owa_logs
  - <<: *route_smtp_logs
  - <<: *route_message_tracking
  - <<: *route_audit_log
```

Второй способ:

1. Откройте сетевой доступ к каталогам с логами.

2. Создайте пользователя с правами доступа к этим каталогам по сети.
3. На удаленном лог-коллекторе настройте файл конфигурации (пример конфигурационного файла лог коллектора для сбора логов по протоколу smb приведен ниже). Обратите внимание, что пути к логам нужно будет прописать корректно, в соответствии с их расположением в вашей системе.
4. Проверьте доступность необходимых адресов и портов, в случае недоступности откройте их на межсетевом экране.
5. Перезапустите службу лог-коллектора
6. Проверьте наличие событий в интерфейсе **Платформы Радар**.

Пример файла конфигурации лог коллектора для сбора логов по протоколу smb:

В данном примере указаны логин/пароль в открытом виде для наглядности, но есть возможность маскировать учетные данные в файле конфигурации с помощью защищенного хранилища (см. инструкцию к лог-коллектору).

```
license_path: "C:/Program Files/Log Collector//pgr-agent.lic"
cluster:
  url: "https://адрес_сервера"
  api_key: "api_key"
controller:
  port: 48000
metric_server:
  port: 48005
secret_file: "C:/Program Files/Log Collector/secret"
secret_storage: "C:/Program Files/Log Collector/secret_storage"
api_server:
  address: "server ip or address"
  port: 8080
  read_timeout: 60
  write_timeout: 60
  wait: 5
  enable_tls: false
  cert_file: "C:/Program Files/Log Collector/certs/server.crt"
  key_file: "C:/Program Files/Log Collector/certs/server.key"
  cert_key_pass: ""
  require_client_cert: false
  ca_file: "C:/Program Files/Log Collector/certs/ca.crt"
  log_level: "WARN"
journal:
  port: 48004
  log_level: "INFO"
  log_path: "C:/Program Files/Log Collector/journal.log"
  rotation_size: 30
  max_backups: 7
  max_age: 7

smb_owa_logs: &smb_owa_logs
  id: "smb_owa_logs"
  remote_servers: ["<ip адрес удаленного узла>", "<имя удаленного узла>"]
  port: 445
  share: "<путь к общему ресурсу>"
  domain: "."
  user: "user"
  password: "password"
```

```
  poll_interval: 1
  files: [ "<путь к общему ресурсу>\\LogFiles" ]
  using_regexp: true
  regexp_starting_dir: "<путь к общему ресурсу>/LogFiles"
  regexp_expression: ".log$"
  dir_check_interval: 2
  read_from_last: true
  enable_watcher: true
  format: "json"
  log_level: "INFO"
    filters:
    blacklist:
      - "^#"

smb_smtp_logs: &smb_smtp_logs
  id: "smb_smtp_logs"
  remote_servers: ["<ip адрес удаленного узла>", "<имя удаленного узла>"]
  port: 445
  share: "<путь к общему ресурсу>"
  domain: "."
  user: "user"
  password: "password"

  poll_interval: 1
  files: [ "<путь к общему ресурсу>\\Logs\\FrontEnd\\ProtocolLog" ]
  using_regexp: true
  regexp_starting_dir: "<путь к общему ресурсу>/Logs/FrontEnd/ProtocolLog/"
  regexp_expression: ".LOG$"
  dir_check_interval: 2
  read_from_last: true
  enable_watcher: true
  format: "json"
  log_level: "INFO"
    filters:
    blacklist:
      - "^#"

smb_message_tracking: &smb_message_tracking
  id: "smb_message_tracking"
  remote_servers: ["<ip адрес удаленного узла>", "<имя удаленного узла>"]
  port: 445
  share: "<путь к общему ресурсу>"
  domain: "."
  user: "user"
  password: "password"

  poll_interval: 1
  files: [ "<путь к общему ресурсу>\\Logs\\MessageTracking" ]
  using_regexp: true
  regexp_starting_dir: "<путь к общему ресурсу>/Logs/MessageTracking"
  regexp_expression: ".LOG$"
  dir_check_interval: 2
  read_from_last: true
  enable_watcher: true
```

```
  format: "json"
  log_level: "INFO"
    filters:
    blacklist:
      - "^#"

smb_audit_log: &smb_audit_log
  id: "smb_audit_log"
  remote_servers: ["<ip адрес удаленного узла>", "<имя удаленного узла>"]
  port: 445
  share: "<путь к общему ресурсу>"
  domain: "."
  user: "user"
  password: "password"

  poll_interval: 1
  files: [ "<путь к общему ресурсу>\\Logging\\CosmosQueue" ]
  using_regexp: true
  regexp_starting_dir: "<путь к общему ресурсу>/Logging/CosmosQueue"
  regexp_expression: ".LOG$"
  dir_check_interval: 2
  read_from_last: true
  enable_watcher: true
  format: "json"
  log_level: "INFO"
    filters:
    blacklist:
      - "^#"

tcp_output_owa: &tcp_output_owa
  id: "tcp_output_owa"
  target_host: "pangeo_server_address"
  port: 1530
tcp_output_smtp: &tcp_output_smtp
  id: "tcp_output_smtp"
  target_host: "pangeo_server_address"
  port: 1531
tcp_output_message_tracking: &tcp_output_message_tracking
  id: "tcp_output_message_tracking"
  target_host: "pangeo_server_address"
  port: 1532
tcp_output_audit_log: &tcp_output_audit_log
  id: "tcp_output_audit_log"
  target_host: "pangeo_server_address"
  port: 1533

senders:
  port: 48002
  tcp:
    - <<: *tcp_output_owa
    - <<: *tcp_output_smtp
    - <<: *tcp_output_message_tracking
    - <<: *tcp_output_audit_log
collectors:
```

```
  smb:
    - <<: *smb_owa_logs
    - <<: *smb_smtp_logs
    - <<: *smb_message_tracking
    - <<: *smb_audit_log

route_owa_logs: &route_owa_logs
  collector_id:
    - "owa_logs"
  sender_id:
    - "tcp_output_owa"
route_smtp_logs: &route_smtp_logs
  collector_id:
    - "smtp_logs"
  sender_id:
    - "tcp_output_smtp"
route_message_tracking: &route_message_tracking
  collector_id:
    - "message_tracking"
  sender_id:
    - "tcp_output_message_tracking"
route_audit_log: &route_audit_log
  collector_id:
    - "audit_log"
  sender_id:
    - "tcp_output_audit_log"
routers:
  - <<: *route_owa_logs
  - <<: *route_smtp_logs
  - <<: *route_message_tracking
  - <<: *route_audit_log
```

8.3. Kaspersky Secure Mail Gateway {#ksmg}

Данное руководство описывает механизм сбора событий Kaspersky Secure Mail Gateway и отправки их в **Платформу Радар**.

8.3.1. Подключение к узлам кластера Kaspersky Secure Mail Gateway по протоколу SSH

Для начала сгенерируйте ключ SSH.

Откройте терминал и выполните команду:

`$ ssh-keygen -t rsa`

На консоль будет выведен следующий диалог:

`Enter file in which to save the key (/home/user/.ssh/id_rsa):`

Нажмите на клавишу Enter. Далее система предложит ввести кодовую фразу для дополнительной защиты SSH-подключения:

`Enter passphrase (empty for no passphrase):`

Этот шаг можно пропустить. При ответе на этот и следующий вопрос просто нажмите клавишу Enter.

После этого ключ будет создан, а на консоль будет выведено следующее сообщение:

```
Your identification has been saved in /home/user/.ssh/id_rsa.
Your public key has been saved in /home/user/.ssh/id_rsa.pub.
The key fingerprint is:
476:b2:a8:7f:08:b4:c0:af:81:25:7e:21:48:01:0e:98 user@localhost

The key's randomart image is:

+--[ RSA 2048]----+
|+.o.             |
|ooE              |
|oo               |
|o.+..            |
|.+.+..  S .      |
|....+  o +       |
|  .o ....        |
|  .  .. .        |
|    ....         |
+-----------------+
```

Далее выполните в терминале команду:

```
$ cat ~/.ssh/id_rsa.pub
```

На консоль будет выведен ключ. Скопируйте его.

Далее нужно загрузить открытый ключ SSH через веб-интерфейс программы. В окне веб-интерфейса Kaspersky Secure Mail Gateway выберите раздел Settings → Application access → SSH access (см. рисунок 84).

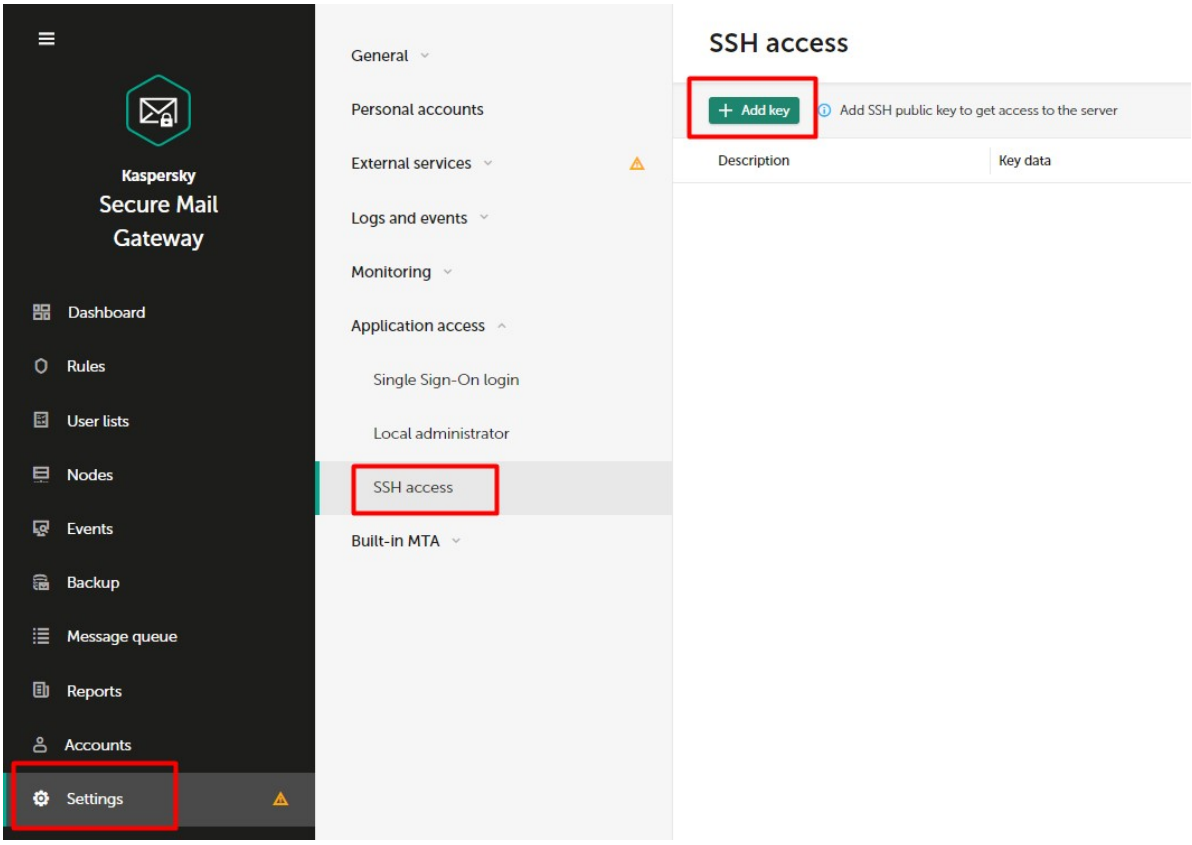

Рисунок 84 - Загрузка открытого ключа.

Нажмите на кнопку Add key. Откроется окно Add an SSH public key (см. рисунок 85).

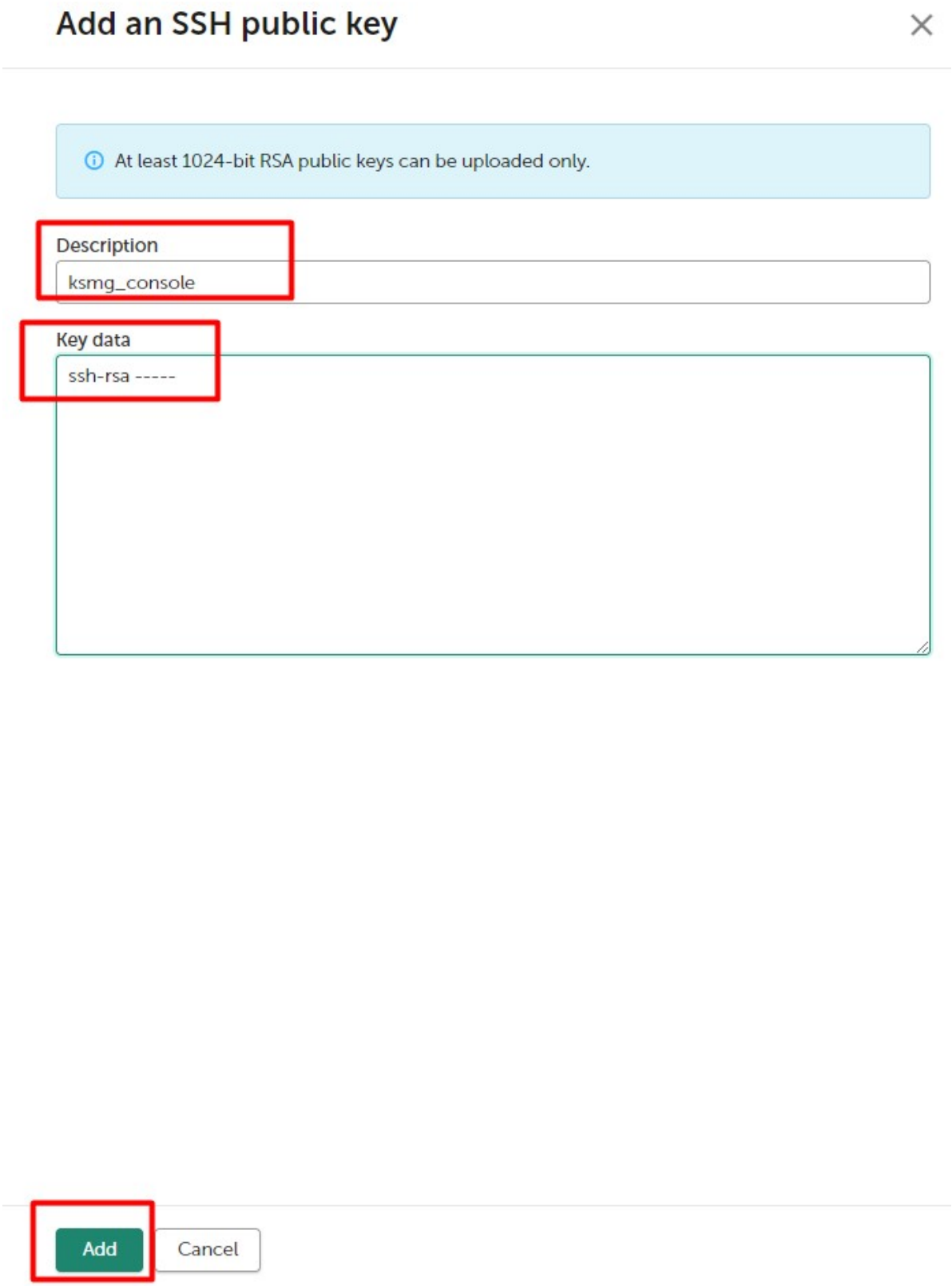

Рисунок 85 - Добавление открытого ключа.

В поле Description введите любую информацию о загружаемом ключе SSH. В поле Key Data скопируйте сгенерированный ранее открытый ключ SSH. Нажмите на кнопку Add.

Открытый ключ SSH будет добавлен. Администратор Kaspersky Secure Mail Gateway сможет подключиться к любому узлу кластера при наличии соответствующего закрытого ключа SSH.

Протестируйте подключение (приведен пример команды, для подключения к вашей консоли; введите путь к вашему ключу и правильный адрес сервера):

```
# ssh -vvv -i .ssh/ksmg_rsa root@your-ksmg-ip-address
```

8.3.2. Настройка экспорта событий в формате CEF

Чтобы настроить экспорт событий в формате CEF:

Подключитесь к консоли управления виртуальной машиной Kaspersky Secure Mail Gateway под учетной записью root, используя закрытый ключ SSH. Вы войдете в режим Technical Support Mode.
Внесите следующие изменения в файл с параметрами экспорта событий `/opt/kaspersky/ksmg/share/templates/core_settings/event_logger.json.template`.
Если вы хотите выбрать категорию (facility) для syslog, в которую будут экспортироваться события, в блоке siemSettings укажите одно из следующих значений параметра facility:

```
    Auth.
    Authpriv.
    Cron.
    Daemon.
    Ftp.
    Lpr.
    Mail.
    News.
    Syslog.
    User.
    Uucp.
    Local0.
    Local1.
    Local2.
    Local3.
    Local4.
    Local5.
    Local6.
    Local7.
```

Рекомендуется указать такую категорию (facility) для syslog, которая не используется другими программами на сервере. По умолчанию установлено значение local2.

Установите значение параметра enabled равным true.

Задайте уровень детализации экспорта, установив одно из следующих значений параметра logLevel:

- Error – экспорт событий, связанных с возникновением ошибок.
- Info – экспорт всех событий.

Пример:

```
"siemSettings":
    {
    "enabled": true,
    "facility": "Local2",
    "logLevel": "Info",
    }
```

Также в файле `/opt/kaspersky/ksmg/share/templates/core_settings/event_logger.json.template` поставьте пробел в следующей строке (см. рисунок 86).

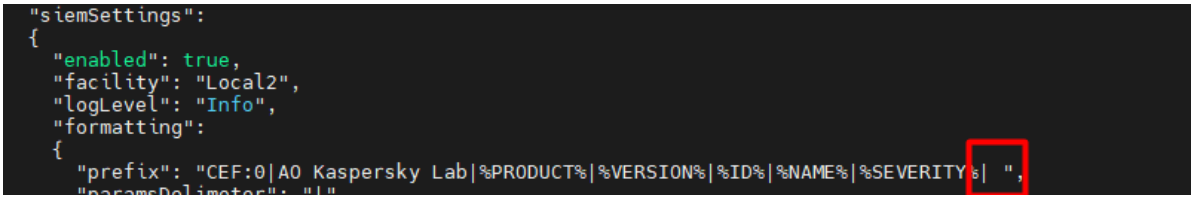

Рисунок 86 - Редактирование файла конфигурации.

Это необходимо сделать для корректного парсинга всех логов. Проблема заключается в следующем: формат CEF, в котором пересылаются логи, выглядит следующим образом:

`CEF:Version|Device Vendor|Device Product|Device Version|Signature ID|Name|Severity|Extension`

Но KSMG отправляет часть логов без обязательного поля Extension. Пробел решает эту проблему и все логи парсятся правильно.

В файле `/etc/rsyslog.conf` измените строку

`*.info;mail.none;authpriv.none;cron.none;local0.none;local1.none /var/log/messages`

на

`*.info;mail.none;authpriv.none;cron.none;local0.none;local1.none;<категория (facility), выбранная на шаге 2>.none /var/log/messages`

Добавьте в файл `/etc/rsyslog.conf` следующую строку:

`<категория (facility), выбранная на шаге 2>.* -/var/log/ksmg-cef-messages`

Создайте файл `/var/log/ksmg-cef-messages` и настройте права доступа к нему. Для этого выполните команды:

```
touch /var/log/ksmg-cef-messages
chown root:klusers /var/log/ksmg-cef-messages
chmod 640 /var/log/ksmg-cef-messages
```

Настройте правила ротации файлов с экспортированными событиями. Для этого добавьте в файл `/etc/logrotate.d/ksmg-syslog` следующие строки:

```
/var/log/ksmg-cef-messages

{
    size 500M
    rotate 10
    notifempty
    sharedscripts
    postrotate
    /usr/bin/systemctl kill -s HUP rsyslog.service >/dev/null 2>&1 || true
    endscript
}
```

Перезапустите службу `rsyslog`. Для этого выполните команду:

```
service rsyslog restart
```

В веб-интерфейсе программы в разделе Параметры → Журналы и события → События внесите изменение в значение любого параметра и нажмите на кнопку Сохранить.
Это необходимо для синхронизации параметров между узлами кластера и применения изменений, внесенных в конфигурационный файл. После этого вы можете вернуть исходное значение измененного параметра.
Экспорт событий в формате CEF будет настроен.

8.3.3. Настройка публикации событий Kaspersky Secure Mail Gateway в Платформу Радар

Выполните инструкцию ниже на каждом узле кластера, события с которого вы хотите публиковать в **Платформу Радар**.

Подключитесь к консоли управления виртуальной машиной Kaspersky Secure Mail Gateway под учетной записью root, используя закрытый ключ SSH. Вы войдете в режим Technical Support Mode.
Укажите адрес и порт подключения к серверу с SIEM-системой. Для этого добавьте в конец файла `/etc/rsyslog.conf` следующие строки:

```
$WorkDirectory /var/lib/rsyslog
$ActionQueueFileName ForwardToSIEM
$ActionQueueMaxDiskSpace 1g
$ActionQueueSaveOnShutdown on
$ActionQueueType LinkedList
$ActionResumeRetryCount -1
<категория (facility)>.* @@<IP-адрес лог коллектора>:<порт(TCP)>
```

Перед внесением изменений в файл `/etc/rsyslog.conf` рекомендуется сделать его резервную копию.

Перезапустите службу `rsyslog`. Для этого выполните команду:

```
service rsyslog restart
```

Публикация событий настроена.

8.3.4. Настройка лог-коллектора на прием событий от Kaspersky Secure Mail Gateway

Пример конфигурационного файла лог коллектора:

```
cluster:
  url: "https://адрес_сервера"
  api_key: "api_key"
controller:
  port: 48000
metric_server:
  port: 48005
secret_file: "/opt/pangeoradar/configs/logcollector/secret"
secret_storage: "/opt/pangeoradar/configs/logcollector/secret_storage"
api_server:
  address: "server ip or address"
  port: 8001
  read_timeout: 60
  write_timeout: 60
  wait: 5
  enable_tls: true
  cert_file: "/opt/pangeoradar/certs/agent.crt"
  key_file: "/opt/pangeoradar/certs/agent.key"
  cert_key_pass: ""
  require_client_cert: false
  ca_file: "/opt/pangeoradar/certs/pgr.crt"
  log_level: "INFO"

journal:
  port: 48004
  log_level: "INFO"
  log_path: "/var/log/logcollector/journal.log"
  rotation_size: 30
  max_backups: 7
  max_age: 7

tcp_input_ksmg: &tcp_input_ksmg
  id: "tcp_input_ksmg"
  host: "collector ip or address"
  port: 2609 # задайте любой незанятый порт, не забудьте указать его в конфиг
файле rsyslog на сервере KSMG
  enable_tls: false
  compression_enabled: false
  connections_limit: 10
  format: "raw"
  log_level: "INFO"

tcp_output_ksmg: &tcp_output_ksmg
  id: "tcp_output_ksmg"
  target_host: "Pangeo server ip"
  port: 2608

senders:
  port: 48001
```

```
  tcp:
    - <<: *tcp_output_ksmg

collectors:
  log_level: "INFO"
  tcp_receiver:
    - <<: *tcp_input_ksmg

route_ksmg: &route_ksmg
  collector_id:
    - "tcp_input_ksmg"
  sender_id:
    - "tcp_output_ksmg"

routers:
  - <<: *route_ksmg
```

При необходимости откройте необходимые порты на межсетевом экране (порты указаны в файле конфигурации).

Перезапустите служб лог-коллектора.

Проверьте наличие событий в интерфейсе **Платформы Радар**.

8.4. IBM Postfix {#postfix}

Настройка отправки событий Postfix MTA осуществляется с помощью rsyslog.

Для того, чтобы настроить отправку событий Postfix с помощью rsyslog, необходимо выполнить следующие действия:

1. Подключиться по SSH к узлу с установленным Postfix MTA.
2. Открыть файл /etc/rsyslog.conf.
3. Добавить строку:

   `mail.*@<IP address>:<Port>`

   где - адрес коллектора событий SIEM.
4. Сохранить файл;
5. Перезапустить службу rsyslog или перезагрузить ее настройки:

   ```
   # systemctl restart rsyslog
   # systemctl reload rsyslog
   ```

При необходимости, в конфигурационном файле Postfix `/etc/postfix/main.cf` можно указать нестандартное значение параметра `syslog_facility` (идентификатора источника событий - по умолчанию mail), затем перезапустить службу Postfix и внести новое значение в файл конфигурации rsyslog.conf.

В свою очередь, настройка конфигурационного файла коллектора событий SIEM является стандартной для типового источника событий syslog:

```
udp_input: &udp_input
  id: "udp_input"
```

```
  host: "0.0.0.0"
  port: 514
  sock_buf_size: 0
  format: "json"
  log_level: "INFO"

tcp_output: &tcp_output
  id: "tcp_output"
  target_host: "<log_collector_ip>"
  port: <log_source_port>
  sock_buf_size: 0
  log_level: "INFO"
  ssl_enable: false
  require_cert: false
  ssl_compression: false
  batch_mode_enable: false

collectors:
  udp_receiver:
    - <<: *udp_input

senders:
  port: 48002
  tcp:
    - <<: *tcp_output

route_1: &route_1
  collector_id:
    - "udp_input"
  sender_id:
    - "tcp_output"

routers:
  - <<: *route_1
```

9. Инфраструктурные системы

9.1. vGate {#vgate}

9.1.1. Настройка подключения источника vGate

Перейдите в настройки и сначала включите отправку событий в Syslog.

Для этого выберите:

Настройки>Аудит>Настройки сбора сообщений.

Выберите чекбоксы - Включить аудит событий и отправка в Syslog.

Нажмите ОК.

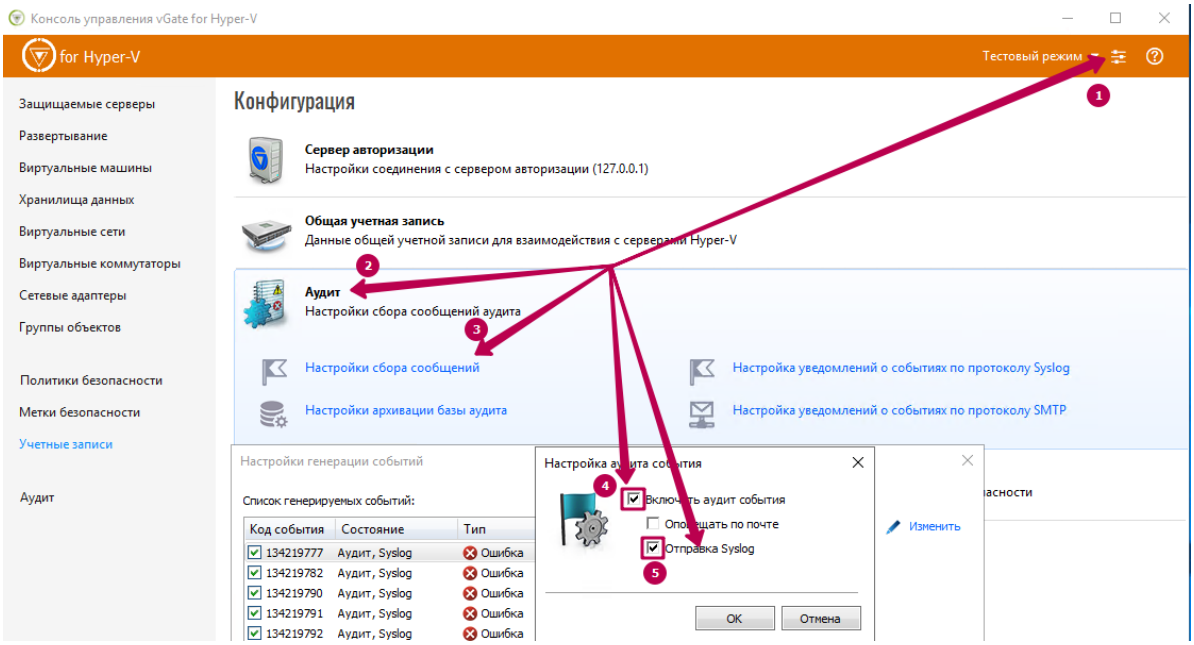

Рисунок 87 - Настройка отправки событий в Syslog.

Для отправки можно включить все уведомления или только необходимые. Для включения всех уведомлений выделите их с помощью комбинации клавиш Ctrl+A и выберите любой чекбокс, после чего будут добавлены все события (см. рисунок 88).

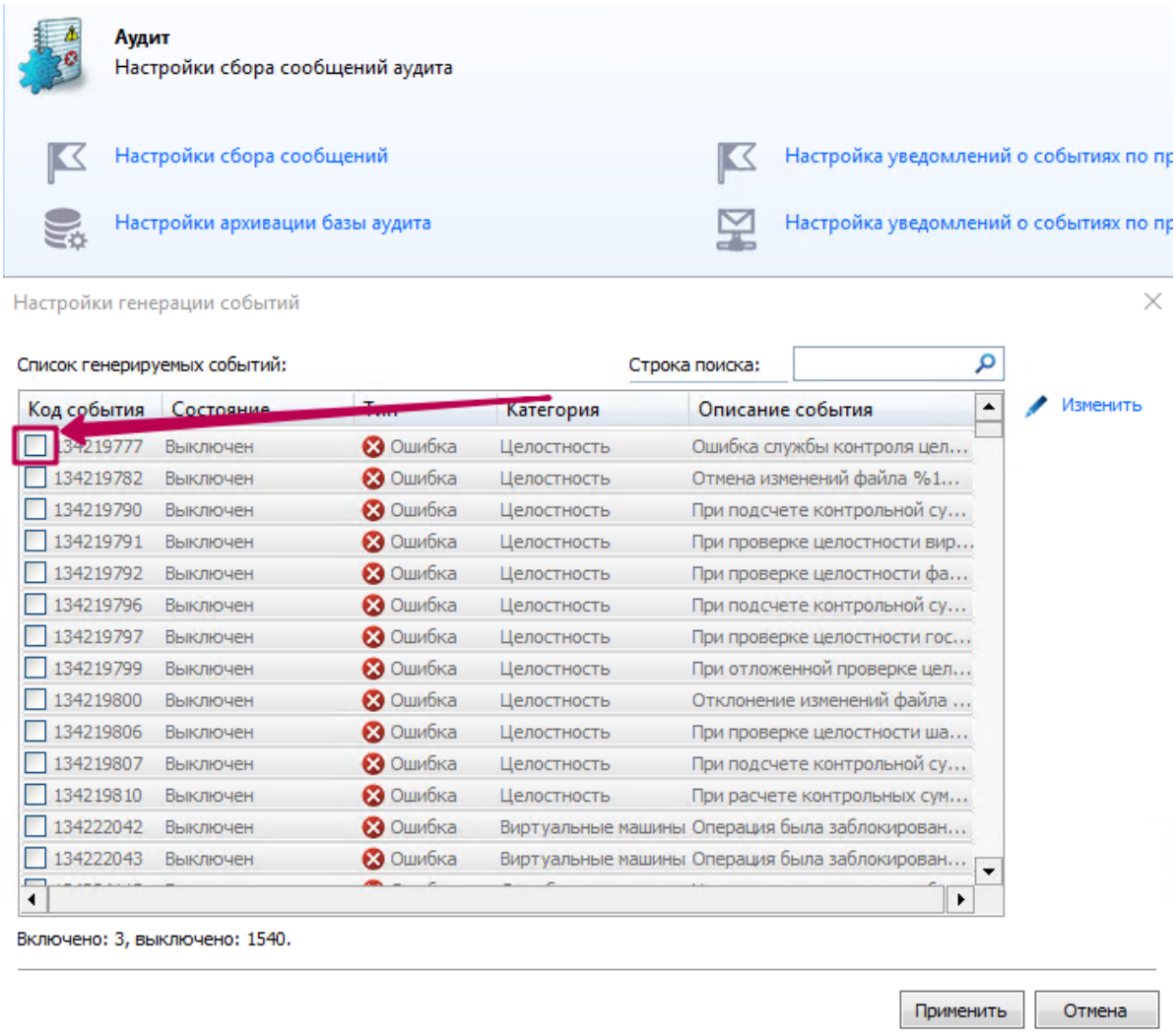

Рисунок 88 - Добавление всех уведомлений.

После того как вы включили возможность отправки событий, необходимо указать адрес log-collector'a и порт.

Для этого выберите:

Настройки>Аудит>Настройка уведомлений о событиях по протоколу Syslog.

Выберите чекбокс - Включить отправку уведомлений.

В поле "Сервер" укажите адрес log-collector'a. В поле "Порт" укажите порт log-collector'a.

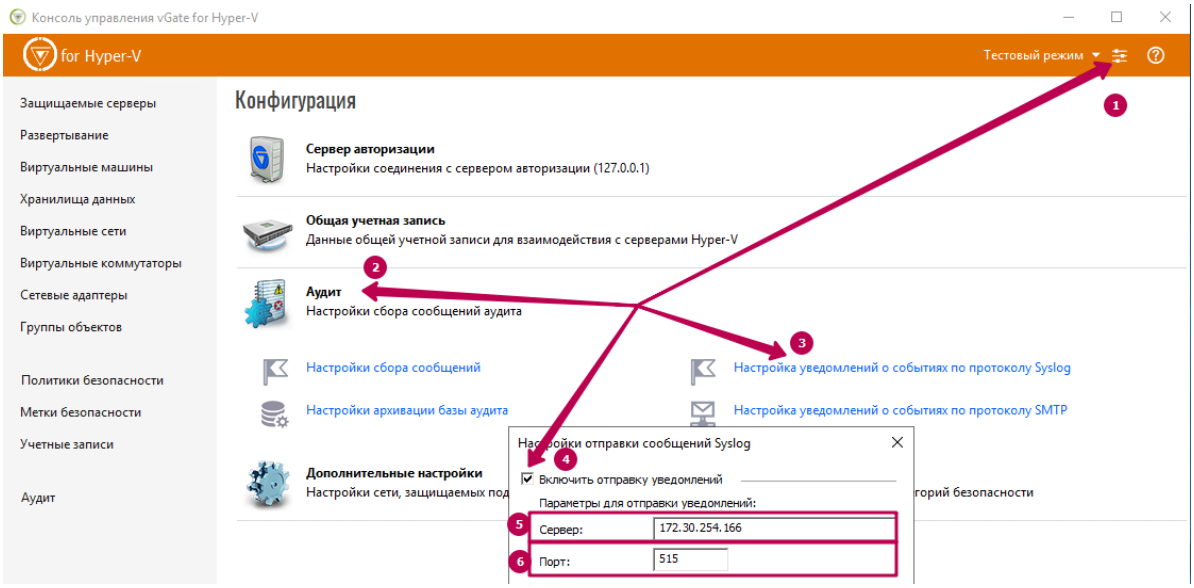

Рисунок 89 - Настройка адреса и порта.

9.1.2. Настройки конфигурации log-collectora

```
# = vGate =
udp_input_515: &udp_input_515
  id: "udp_input_515"
  host: "172.30.254.166"
  port: 515
  sock_buf_size: 0
  format: "json"

tcp_output_2745: &tcp_output_2745
  id: "tcp_output_2745"
  target_host: "172.30.254.67"
  port: 2745

#=====
senders:
  port: 48002
  log_level: "INFO"
  tcp:
    - <<: *tcp_output_2745
collectors:
  log_level: "INFO"
  udp_receiver:
    - <<: *udp_input_515
#====
route_1: &route_1
  collector_id:
    - "udp_input_515"
  sender_id:
```

```
    - "tcp_output_2745"
#====
routers:
  - <<: *route_1
```

9.2. ISC Bind DNS {#bind}

9.2.1. Настройка логирования bind

В файл `/etc/bind/named.conf` добавьте следующие строки:

```
logging {
    channel named {
        file "/var/log/named/named.log" versions 10 size 20M;
        severity info;
        print-time yes;
        print-category yes;
        print-severity yes;
    };

    channel security {
        file "/var/log/named/security.log" versions 10 size 20M;
        severity info;
        print-time yes;
        print-severity yes;
    };

    channel dnssec {
        file "/var/log/named/dnssec.log" versions 10 size 20M;
        severity info;
        print-time yes;
        print-severity yes;
    };

    channel resolver {
        file "/var/log/named/resolver.log" versions 10 size 20M;
        severity info;
        print-time yes;
        print-severity yes;
    };

    channel query_log {
        file "/var/log/named/query.log" versions 10 size 80M;
        severity info;
        print-time yes;
        print-severity yes;
    };

    channel query_error {
        file "/var/log/named/query_errors.log" versions 10 size 20M;
        severity info;
        print-time yes;
        print-severity yes;
    };
```

```
    channel lame_servers {
        file "/var/log/named/lame-servers.log" versions 10 size 20M;
        severity info;
        print-time yes;
        print-severity yes;
    };

    channel capacity {
        file "/var/log/named/capacity.log" versions 10 size 20M;
        severity info;
        print-time yes;
        print-severity yes;
    };

    channel database {
        file "/var/log/named/database.log" versions 10 size 20M;
        severity info;
        print-time yes;
        print-severity yes;
    };

    channel update {
        file "/var/log/named/update.log" versions 10 size 10M;
        severity info;
        print-time yes;
        print-severity yes;
    };

    category default        { default_syslog; named; };
    category general        { default_syslog; named; };
    category security       { security; };
    category queries        { query_log; };
    category query-errors   { query_error; };
    category lame-servers   { lame_servers; };
    category dnssec         { dnssec; };
    category edns-disabled  { default_syslog; resolver; };
    category config         { default_syslog; named; };
    category resolver       { resolver; };
    category cname          { resolver; };
    category spill          { capacity; };
    category rate-limit     { capacity; };
    category database       { database; };
    category client         { default_syslog; named; };
    category network        { default_syslog; named; };
    category unmatched      { named; };
    category delegation-only { named; };
    category update         { default_syslog; update; };
    category update-security { default_syslog; update; };
};
```

Далее необходимо сохранить, выйти и проверить конфигурацию.

```
sudo named-checkconf /etc/bind/named.conf.options
```

Далее создать директорию где будут хранится журналы и нужные файлы под них, задать необходимые разрешения и владельцев, перезапустить сервис Bind9.

```
mkdir -p /var/log/named
touch /var/log/named/named.log
touch /var/log/named/security.log
touch /var/log/named/dnssec.log
touch /var/log/named/resolver.log
touch /var/log/named/query.log
touch /var/log/named/query_errors.log
touch /var/log/named/lame-servers.log
touch /var/log/named/capacity.log
touch /var/log/named/database.log
touch /var/log/named/update.log
chown bind:bind /var/log/named
chown bind:bind /var/log/named/*.log
chmod 640 /var/log/named/*.log
service bind9 restart
```

Это позволит начать отслеживание логов bind.

9.2.2. Настройка rsyslog на сервере bind

Создайте шаблон для rsyslog'a по пути `/etc/rsyslog.d/`. Например bind.conf

```
sudo nano /etc/rsyslog.d/bind.conf
```

Содержимое файла представлено ниже:

```
module(load="imfile" PollingInterval="10")

input(type="imfile"
      reopenOnTruncate="on"
      File="/var/log/named/capacity.log"
      Tag="tag_dns_log")
input(type="imfile"
      reopenOnTruncate="on"
      File="/var/log/named/dnssec.log"
      Tag="tag_dns_log")
input(type="imfile"
      reopenOnTruncate="on"
      File="/var/log/named/named.log"
      Tag="tag_dns_log")
input(type="imfile"
      reopenOnTruncate="on"
      File="/var/log/named/query.log"
      Tag="tag_dns_log")
input(type="imfile"
      reopenOnTruncate="on"
      File="/var/log/named/security.log"
      Tag="tag_dns_log")
input(type="imfile"
      reopenOnTruncate="on"
      File="/var/log/named/database.log"
```

```
      Tag="tag_dns_log")
input(type="imfile"
      reopenOnTruncate="on"
      File="/var/log/named/lame-servers.log"
      Tag="tag_dns_log")
input(type="imfile"
      reopenOnTruncate="on"
      File="/var/log/named/query_errors.log"
      Tag="tag_dns_log")
input(type="imfile"
      reopenOnTruncate="on"
      File="/var/log/named/resolver.log"
      Tag="tag_dns_log")
input(type="imfile"
      reopenOnTruncate="on"
      File="/var/log/named/update.log"
      Tag="tag_dns_log")
if $syslogtag == 'tag_dns_log' then @x.x.x.x:2800
& stop
```

Где вместо x.x.x.x укажите ip-адрес лог-коллектора и порт после двоеточия.

Перезапустите службу rsyslog.

```
systemctl restart rsyslog
```

9.2.3. Настройки конфигурации log-collectora

```
udp_input_2800: &udp_input_2800
  id: "udp_input_2800"
  host: "192.168.1.100"
  port: 2800
  sock_buf_size: 0
  format: "json"

tcp_output_2800: &tcp_output_2800
  id: "tcp_output_2800"
  target_host: "192.168.1.200"
  port: 2800

senders:
  port: 48002
  log_level: "INFO"
  tcp:
    - <<: *tcp_output_2800

collectors:
  udp_receiver:
    - <<: *udp_input_2800

route_1: &route_1
  collector_id:
    - "udp_input_2800"
  sender_id:
```

```
    - "tcp_output_2800"

routers:
  - <<: *route_1
```

Вместо x.x.x.x укажите ip-адрес лог-коллектора и выбранный ранее порт для tcp_input.

9.3. Dell IDRAC {#idrac}

9.3.1. Включение аудита IDRAC

1. Войдите в Chasis Management Controller UI . Выберите Server Overview > Properties > Status.

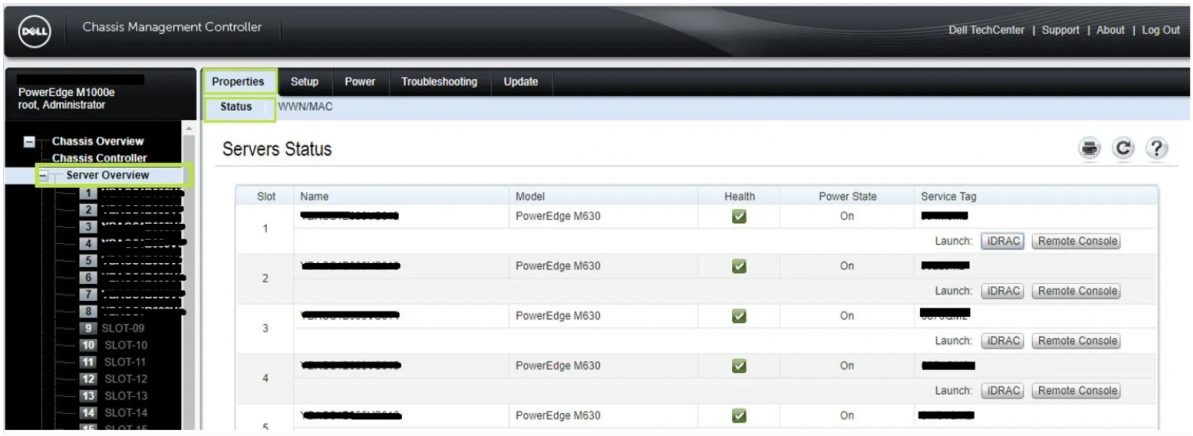

Рисунок 90 - Выбор настроек.

2. Нажмите на кнопку IDRAC для серверов, на которых вы хотите включить логирование.

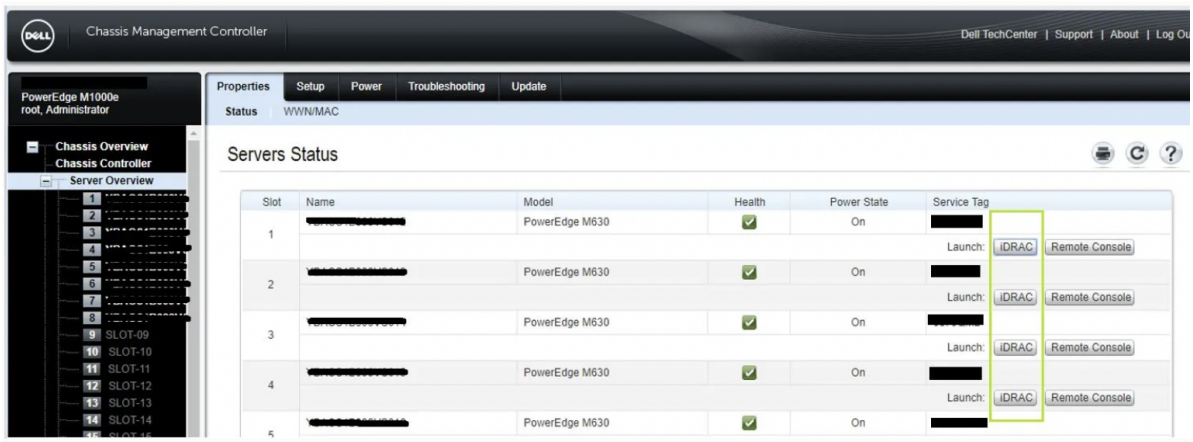

Рисунок 91 - Выбор серверов.

3. Выберите Server > Logs > Settings. На странице Settings отметьте галочкой Enable the Remote Syslog Settings , затем введите IP лог-коллектора в поле Syslog Server и выбранный номер порта для подключения в поле Port Number.

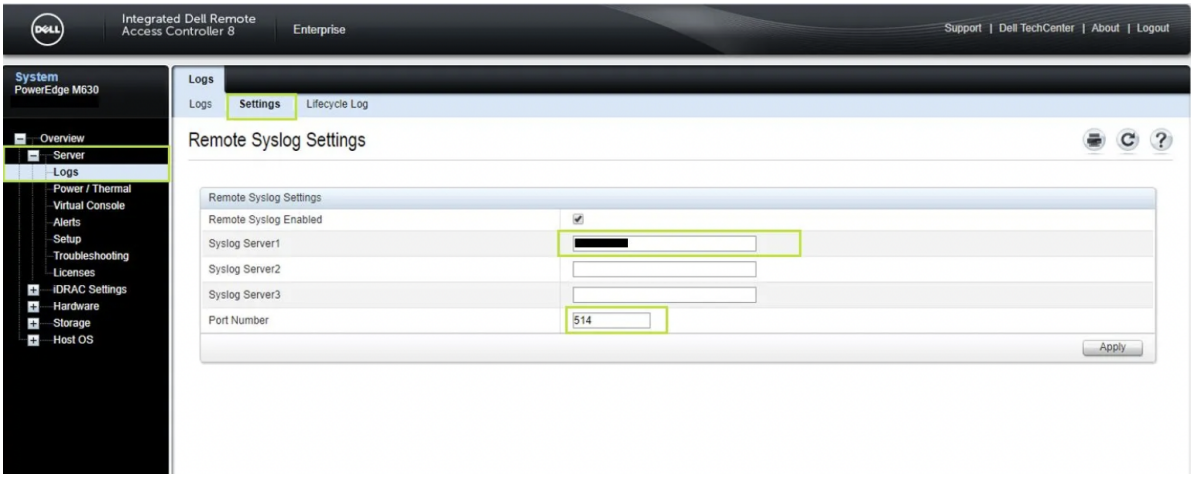

Рисунок 92 - Настройка параметров логгирования.

4. Выберите Server > Alerts > Включите Alerts и Alert Filter в зависимости от ваших требований.

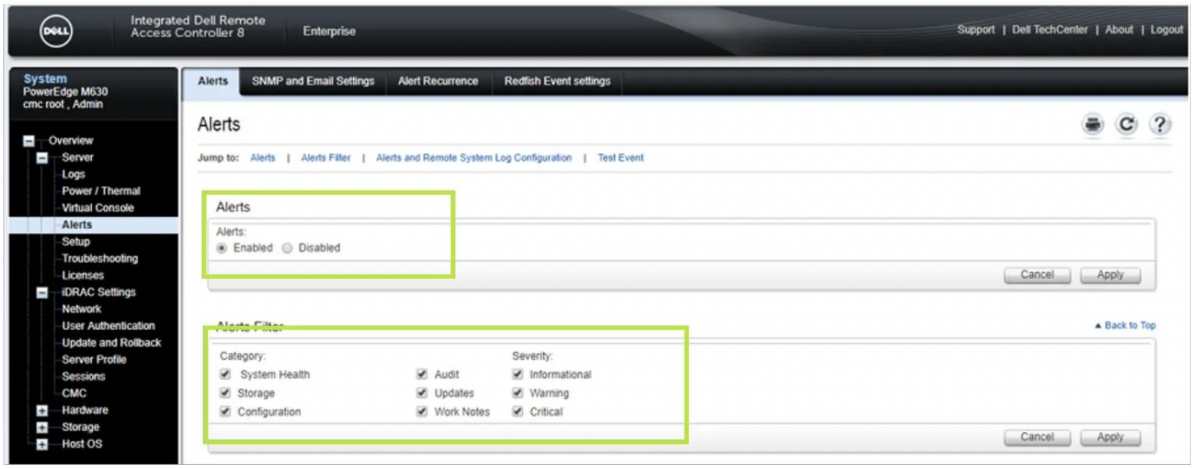

Рисунок 93 - Выбор алертов.

5. Чуть ниже во вкладке Alerts and Remote System Log Configuration > отметьте галочками Remote System Log для создания необходимых алертов.

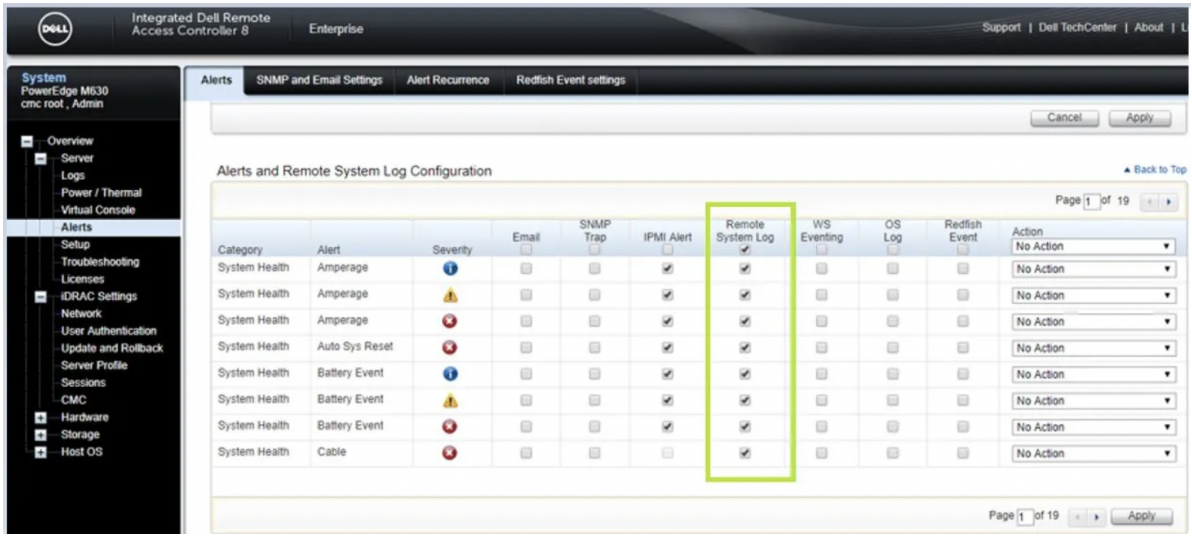

Рисунок 94 - Создание алертов.

```
Повторите создание алертов на всех страницах.
```

9.3.2. Добавление новой конфигурации в коллектор

Приведенные настройки с описанием для добавления в config.yaml ниже:

```
tcp_input_3: &tcp_input_3
  id: "tcp_input_3"
  host: "<IP-адрес платформы>"
  port: 999
  sock_buf_size: 0
  format: "json"
  encoding:
    change_to_utf8: true
    original_encoding: "cp1251"
```

В качестве порта для подключения необходимо указать порт, выбранный на шаге 3.

9.4. Linux NFS Server {#lnfs}

Данное руководство описывает механизм сбора событий Linux NFS server и отправки их в **Платформу Радар**.

9.4.1. Настройка журналирования NFS

> Все настройки, перечисленные ниже, должны осуществляться с правами администратора (root). Если вы работаете под непривилегированным пользователем, используйте команду sudo. Пример конфигурации приведен для сервера под управлением ОС Debian. Если вы используете другой дистрибутив Linux, то расположение файлов и настройки могут немного отличаться, в таком случае обратитесь к документации к своему дистрибутиву.

1. Откройте файл `/etc/default/nfs-kernel-server`

   `# nano /etc/default/nfs-kernel-server`

   и добавьте следующую строку:

   `RPCNFSDOPTS="--syslog"`

   Если параметр RPCNFSDOPTS уже присутствует, то просто нужно добавить ключ --syslog

2. Откройте файл `/etc/idmapd.conf`

   `# nano /etc/idmapd.conf`

   и для переменной Verbosity введите значение 4:

   `Verbosity = 4`

3. Выполните следующую команду:

   `# rpcdebug -m nfsd -s all`

4. Перезапустите службу nfs-kernel-server

   `# systemctl restart nfs-kernel-server.service`

5. Откройте файл /etc/rsyslog.conf

   `# nano /etc/rsyslog.conf`

   и добавьте в конец файла следующую строку:

   `:msg,contains,"nfsd" @@172.30.250.32:4570`

6. Перезапустите службу rsyslog

   `# systemctl restart rsyslog.service`

9.4.2. Конфигурация лог коллектора

Пример конфигурационного файла лог коллектора:

```
license_path: "C:/Program Files/Log Collector//pgr-agent.lic"
cluster:
  url: "https://адрес_Радар_Мастер"
  api_key: "api_key"
controller:
  port: 48000
metric_server:
  port: 48005
secret_file: "C:/Program Files/Log Collector/secret"
secret_storage: "C:/Program Files/Log Collector/secret_storage"
api_server:
  address: "ip-адрес лог коллектора"
  port: 8080
  read_timeout: 60
  write_timeout: 60
  wait: 5
  enable_tls: false
  cert_file: "C:/Program Files/Log Collector/certs/agent.crt"
  key_file: "C:/Program Files/Log Collector/certs/agent.key"
  cert_key_pass: ""
  require_client_cert: false
  ca_file: "C:/Program Files/Log Collector/certs/pgr.crt"
  log_level: "WARN"
journal:
  port: 48004
  log_level: "INFO"
  log_path: "C:/Program Files/Log Collector/journal.log"
  rotation_size: 30
  max_backups: 7
  max_age: 7

tcp_input_linux_nfs: &tcp_input_linux_nfs
  id: "tcp_input_linux_nfs"
  host: "ip-адрес лог коллектора"
  port: 4570
  enable_tls: false
  compression_enabled: false
  connections_limit: 10
  format: "json"
  log_level: "INFO"

tcp_output_linux_nfs: &tcp_output_linux_nfs
  id: "tcp_output_linux_nfs"
  target_host: "ip-адрес балансера"
  port: 4570

senders:
  port: 48002
```

```
  tcp:
    - <<: *tcp_output_linux_nfs

collectors:
  tcp_receiver:
    - <<: *tcp_input_linux_nfs

route_linux_nfs: &route_linux_nfs
  collector_id:
    - "tcp_input_linux_nfs"
  sender_id:
    - "tcp_output_linux_nfs"

routers:
  - <<: *route_linux_nfs
```

9.5. Microsoft DNS {#msdns}

Для получения событий запросов и ответов DNS сервера необходимо включить журнал отладки, по умолчанию он выключен.
Для этого откройте оснастку DNS сервера, далее перейдите в его свойства. В свойствах DNS сервера выберите **"Ведение журнала отладки"**.

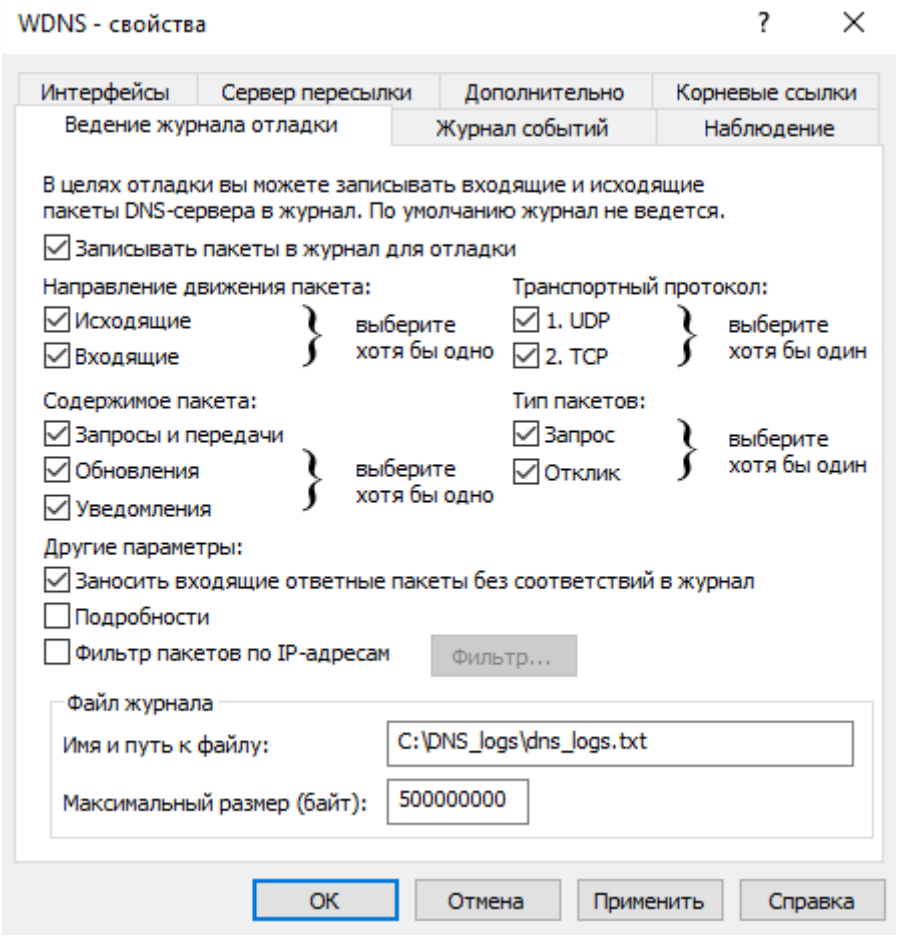

Рисунок 95 - Ведение журнала отладки.

Отметьте чекбоксы (см. рисунок 95) и нажмите кнопку **"ОК"**.

В поле **"Имя и путь к файлу"** укажите Ваш файл, куда DNS сервер будет создавать события.

Нажмите кнопку **"Применить"**.

9.5.1. Настройки Logcollectora

9.5.1.1. Сценарий, когда лог-коллектор развёрнут на том же хосте где и сам DNS сервер.

```
cluster:
  url: "https://192.168.1.200:9000/cm/api/agent/"
  api_key: "bac1e342-f819-1a9f-5a16-c925b8b407b7"

controller:
  port: 48000

metric_server:
  port: 48005

secret_file: "C:\\Program Files\\Log Collector\\secret"
secret_storage: "C:\\Program Files\\Log Collector\\secret.storage"
api_server:
  address: "192.168.1.100"
  port: 8080
  read_timeout: 60
  write_timeout: 60
  wait: 5
  enable_tls: false
  cert_file: "certs/server.crt"
  key_file: "certs/server.key"
  cert_key_pass: ""
  require_client_cert: true
  ca_file: "certs/pgr.crt"
  log_level: "INFO"
journal:
  port: 48004
  log_level: "INFO"
  log_path: "C:/Program Files/Log Collector/journal.log"
  rotation_size: 30
  max_backups: 7
  max_age: 7

win_dns: &win_dns
  id: "win_dns"
  poll_interval: 1
  files: ["C:\\\\DNS_logs\\dns_logs.txt"]
  using_regexp: false
  regexp_starting_dir: "c://DNS_logs/"
  regexp_expression: ".txt"
  dir_check_interval: 2
  read_from_last: true
  enable_watcher: true
  log_level: "INFO"
  format: "json"
  encoding:
    change_to_utf8: false
    original_encoding: "cp1251"
  filters:
    blacklist: ["^M.*", "^D.*", "^L.*", "^\\s+.*"]
```

```
win_dns_out: &win_dns_out
  id: "win_dns_out"
  target_host: "192.168.1.200"
  port: 1516

senders:
  port: 48002
  tcp:
    - <<: *win_dns_out

collectors:
  files:
    - <<: *win_dns

route_1: &route_1
  collector_id:
    - "win_dns"
  sender_id:
    - "win_dns_out"

routers:
  - <<: *route_1
```

9.5.1.2. Сценарий, когда лог-коллектор забирает события с DNS сервера по SMB.

```
cluster:
  url: "https://192.168.1.200:9000/cm/api/agent/"
  api_key: "57dbc2b0-41e8-0f55-95d8-1c19c2e44347"

controller:
  port: 48000

metric_server:
  port: 48005

secret_file: "C:\\Program Files\\Log Collector\\secret"
secret_storage: "C:\\Program Files\\Log Collector\\secret.storage"
api_server:
  address: "192.168.1.100"
  port: 8080
  read_timeout: 60
  write_timeout: 60
  wait: 5
  enable_tls: false
  cert_file: "certs/server.crt"
  key_file: "certs/server.key"
  cert_key_pass: ""
  require_client_cert: true
  ca_file: "certs/pgr.crt"
  log_level: "INFO"

journal:
  port: 48004
```

```
  log_level: "INFO"
  log_path: "C:\\Program Files\\Log Collector\\journal.log"
  rotation_size: 30
  max_backups: 7
  max_age: 7

smb_collector_out: &smb_collector_out
  id: "smb_collector_out"
  target_host: "192.168.1.200"
  port: 1516

smb_collector: &smb_collector
  id: "smb_collector"
  remote_servers: ["192.168.1.2"]
  port: 445
  share: "\\\\192.168.88.1.2\\DNS_logs"
  domain: "."
  user: "logcoll"
  password: "1"
  poll_interval: 5
  files: ["dns_logs.txt"]
  using_regexp: false
  regexp_starting_dir: "."
  regexp_expression: ".(?:txt|log)$"
  dir_check_interval: 5
  read_from_last: true
  format: "json"
  log_level: "INFO"
  filters:
   blacklist: ["^M.*", "^D.*", "^L.*", "^\\s+.*"]

senders:
  port: 48002
  log_level: "INFO"
  tcp:
    - <<: *smb_collector_out

collectors:
  log_level: "INFO"
  smb:
    - <<: *smb_collector

route_1: &route_1
  collector_id:
    - "smb_collector"
  sender_id:
    - "smb_collector_out"

routers:
  - <<: *route_1
```

10. Системы управления базами данных

10.1. Microsoft SQL Server Audit Windows Event Log {#mssql}

Получение событий с Microsoft SQL Server возможно реализовать двумя способами:

- через события Windows events;
- через ODBC коллектор.

10.1.1. Настройка получения событий через windows events.

Включение аудита MS SQL Server:

1. Запустите Microsoft SQL Server Management Studio.
2. В окне подключения к базе данных укажите название экземпляра и введите учетные данные (см. рисунок 96).

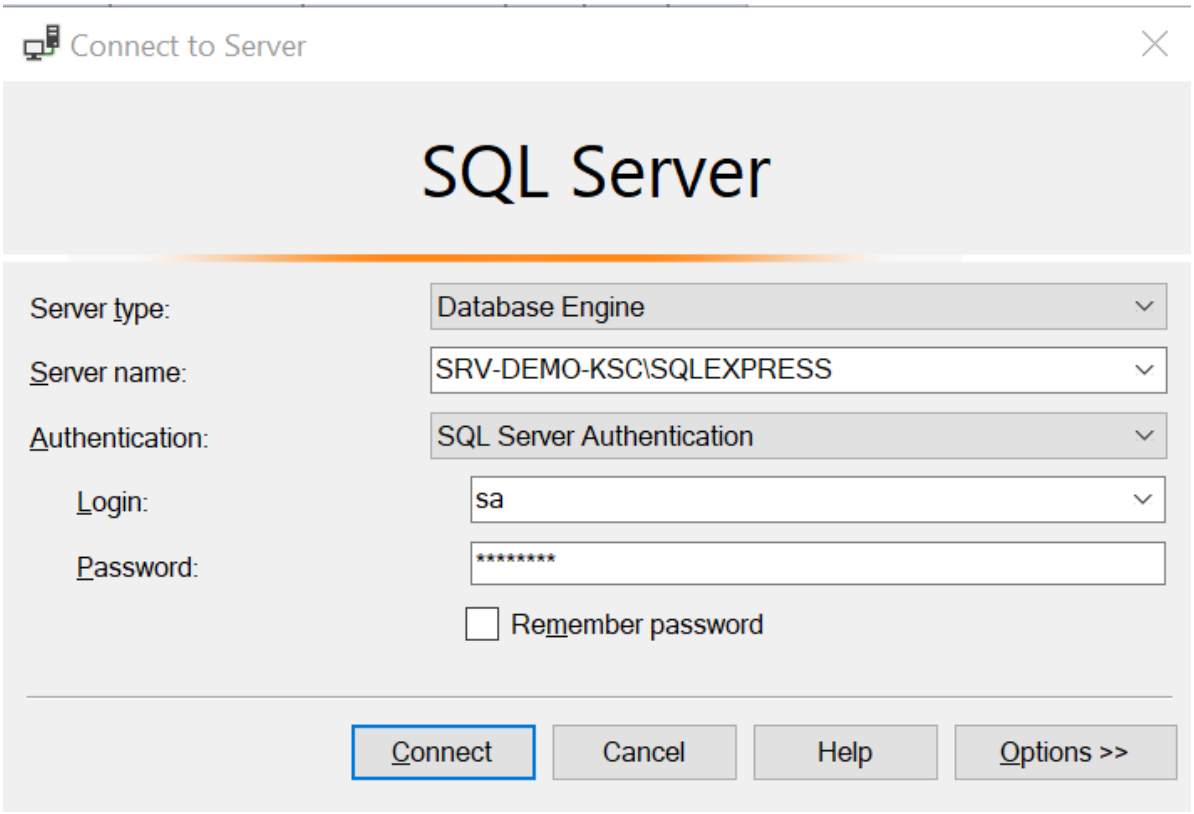

Рисунок 96 - Подключение к базе данных

3. В панели Object explorer перейдите во вкладку Security → Audits. По правому щелчку мыши выберите опцию New Audit... (см. рисунок 97).

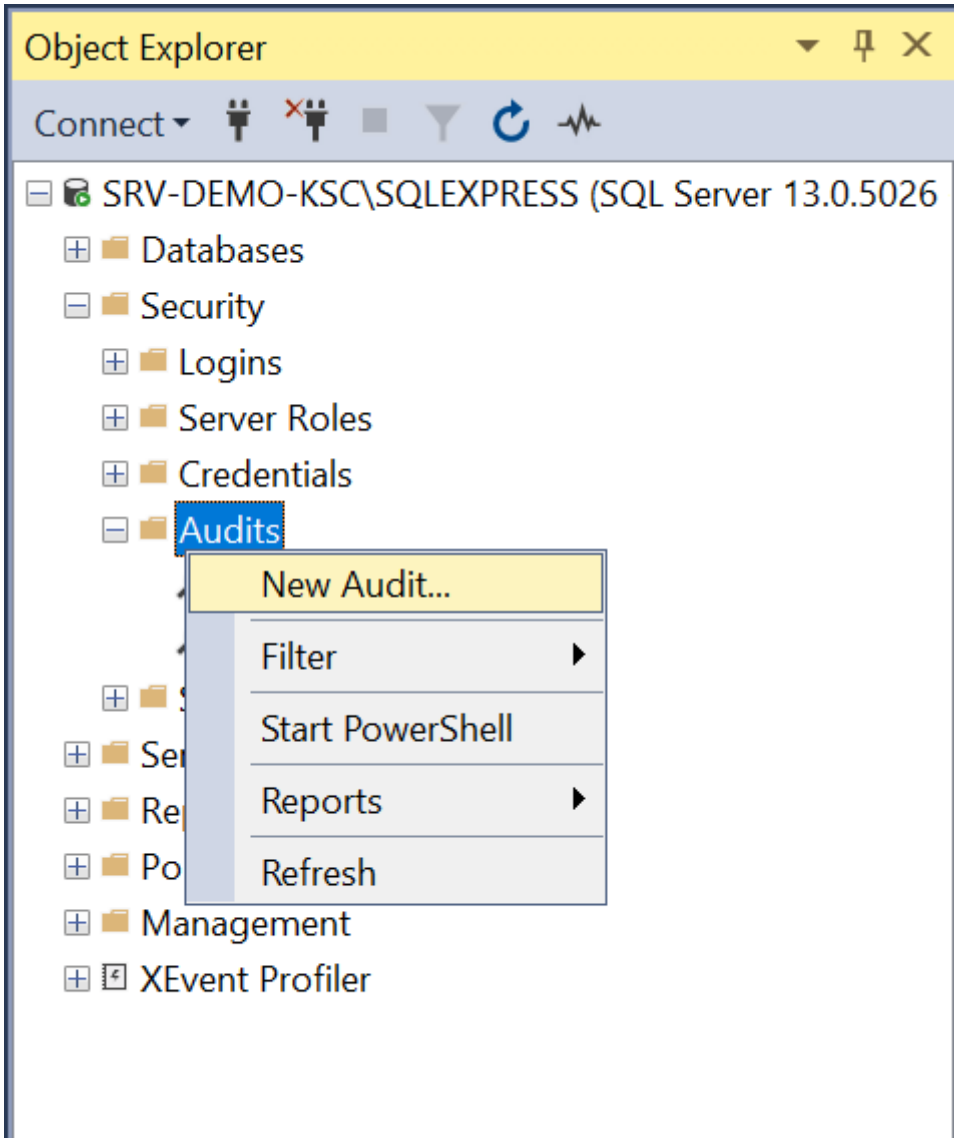

Рисунок 97 - Создание аудита

3. В открывшейся вкладке Create Audit укажите название аудита в поле Audit name. В качестве Audit destination выберите Application Log, нажмите ОК (см. рисунок 98).

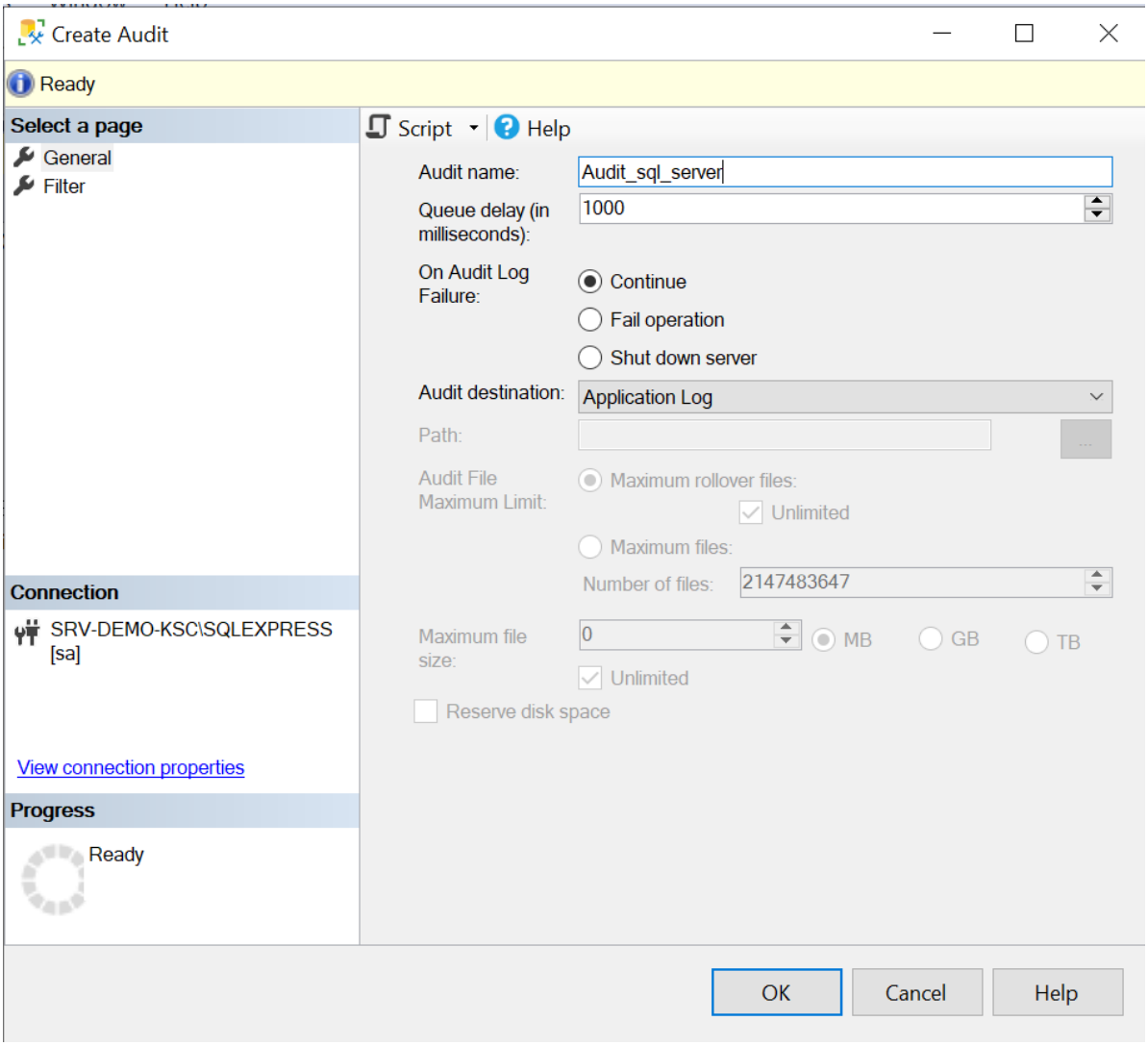

Рисунок 98 - Настройка аудита

4. В панели Object explorer перейдите во вкладку Security → Server Audit Specification. По правому щелчку мыши выберите опцию New Server Audit Specification... (см. рисунок 99).

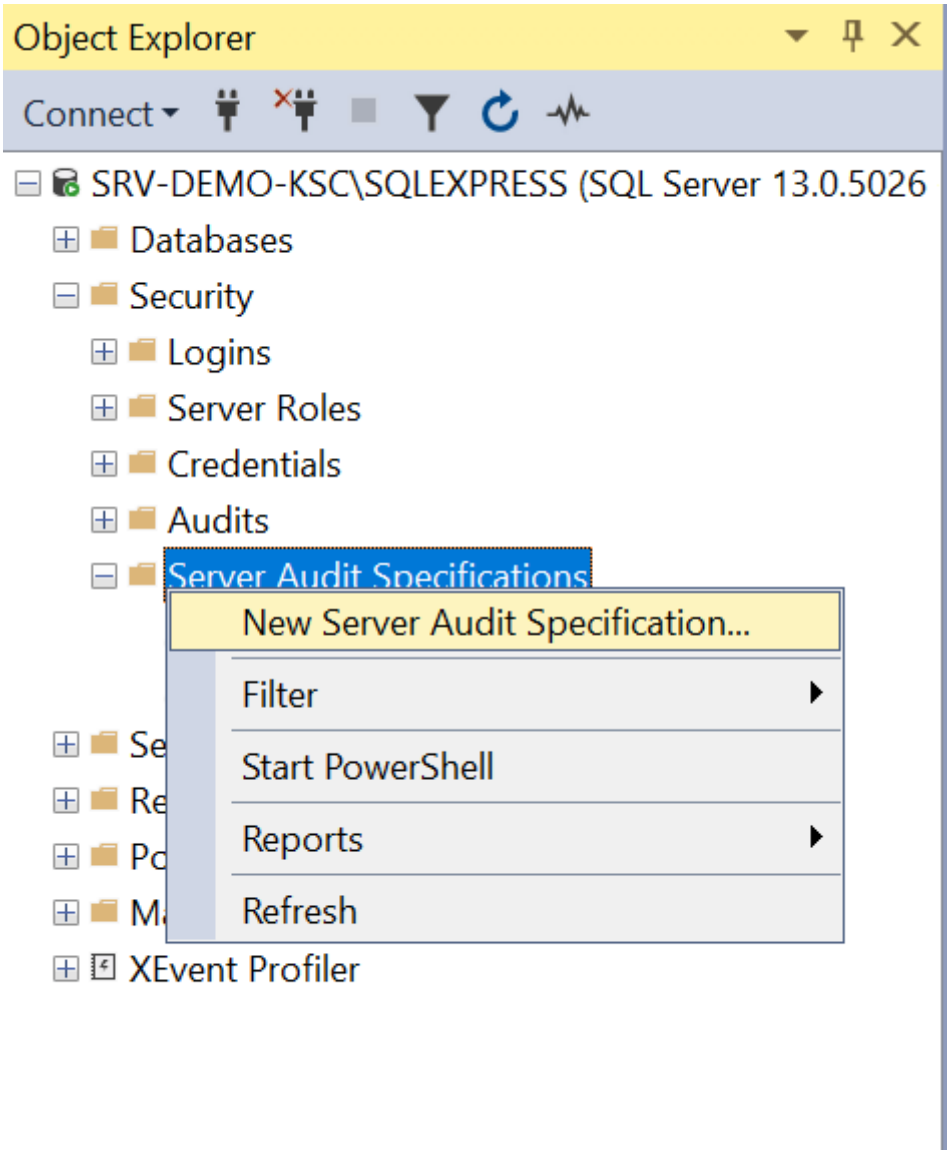

Рисунок 99 - Создание спецификации аудита

5. В открывшейся вкладке Create Server Audit Specification укажите название спецификации аудита в поле Name. В поле Audit выберите ранее созданный аудит из выпадающего списка. В поле Actions выберите типы событий для отслеживания, нажмите OK (см. рисунок 100).

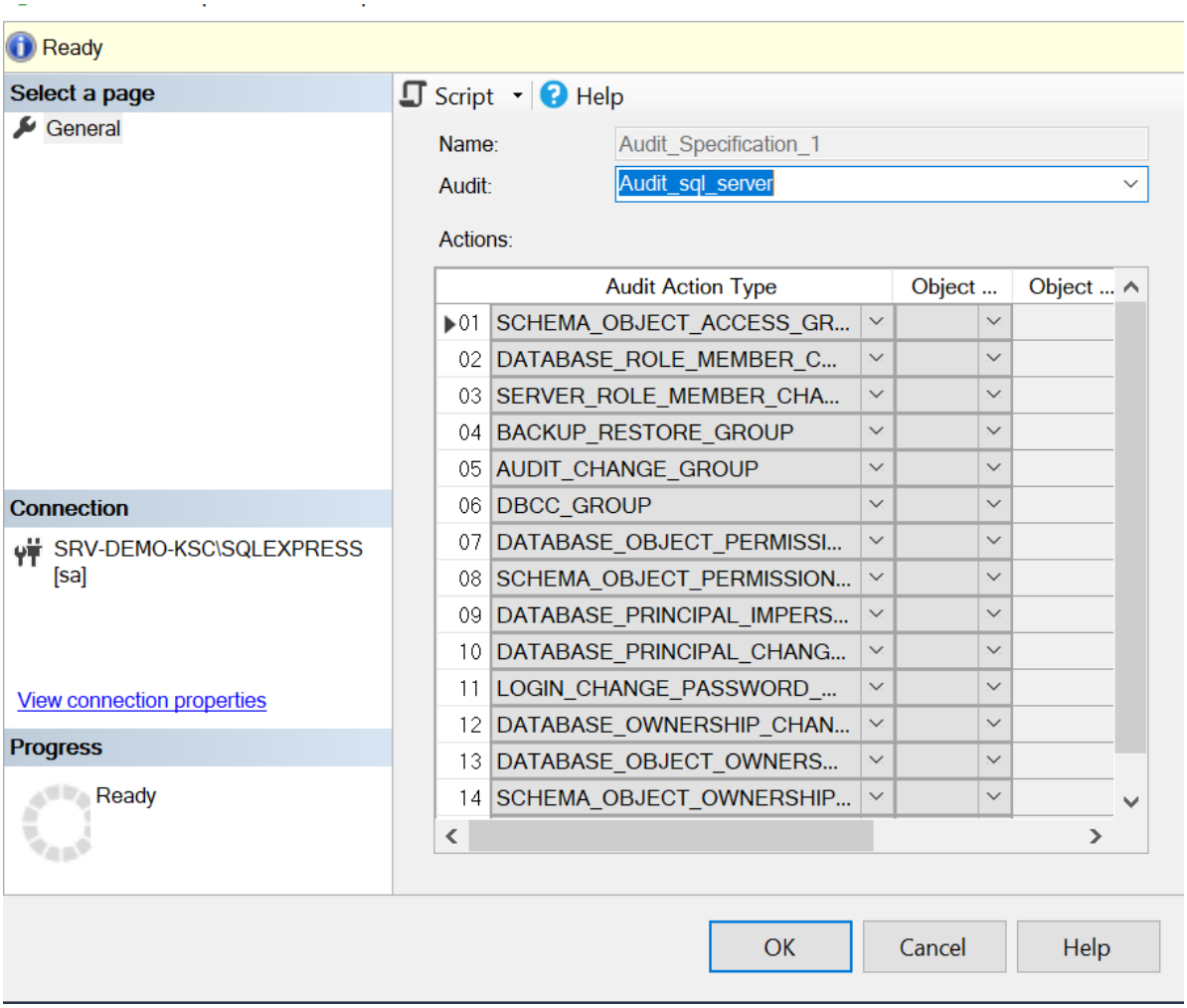

Рисунок 100 - Настройка спецификации аудита

Создание учетной записи windows:

1. В панели управления Windows откройте консоль Computer Management (Управление компьютером).
2. В консоли откройте раздел System Tools (Служебные программы) → Local Users and Groups (Локальные пользователи и группы) → Users (Пользователи).
3. В контекстном меню раздела Users (Пользователи) выберите функцию New User (Новый пользователь) для создания нового пользователя (см. рисунок 101).

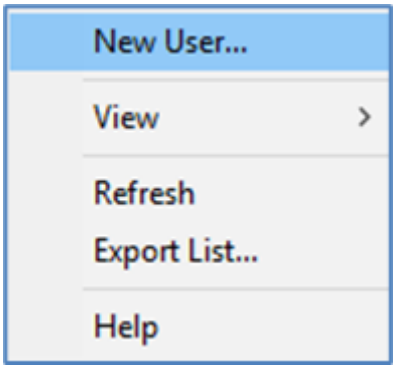

Рисунок 101 - Создание пользователя

4. В открывшемся окне New User (Новый пользователь) введите следующие денные (см. рисунок 102):
    - В поле Name (Имя) ввести имя нового пользователя.
    - В поле Password (Пароль) установить пароль и подтвердить его в поле Confirm Password (Подтвердить).

При необходимости выставить настройки в пунктах:

- User cannot change password (Запретить смену пароля пользователем).
- Password never expires (Срок действия пароля неограничен).
- Для создания пользователя с заданными параметрами нажать кнопку Create.

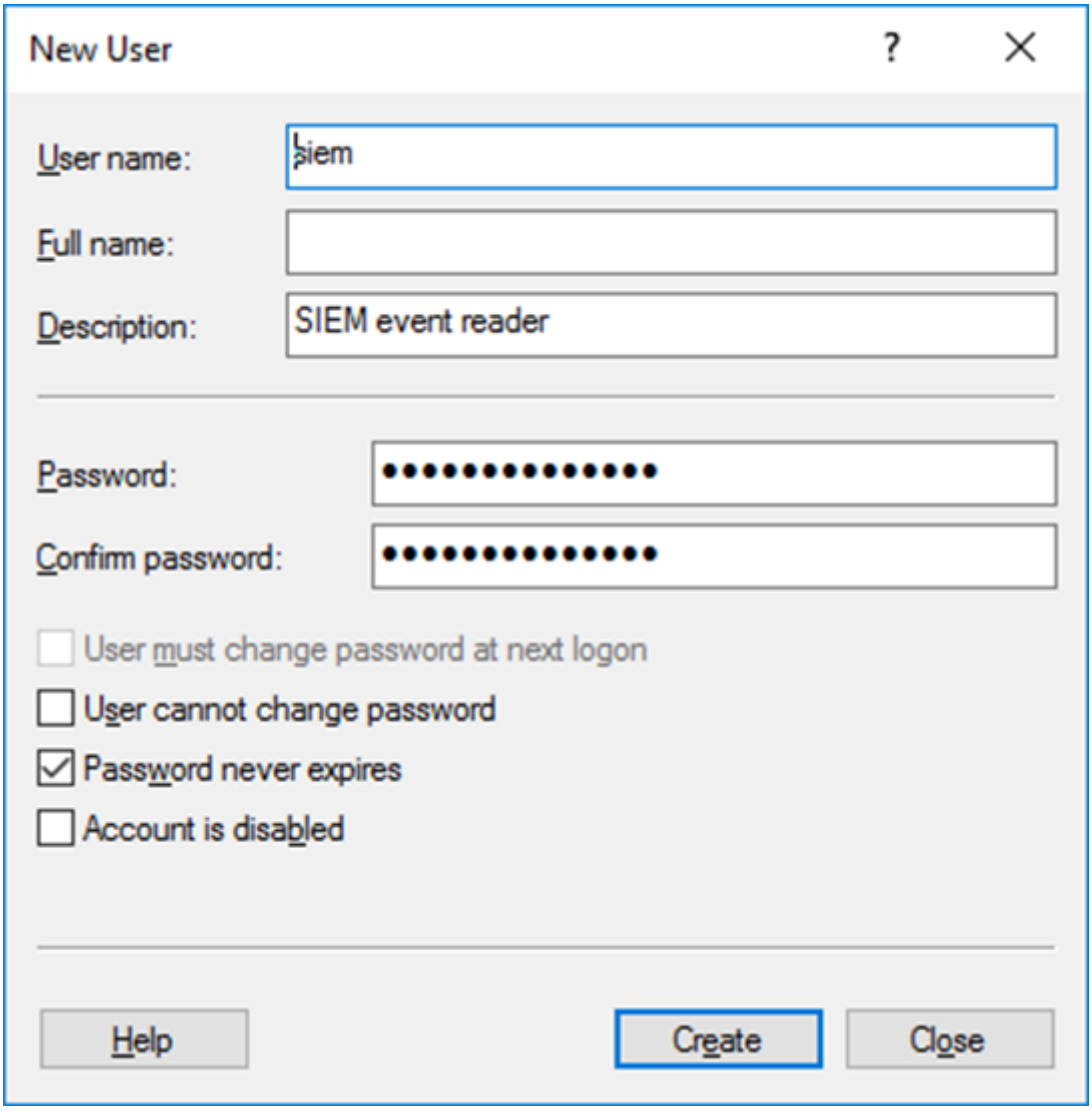

Рисунок 102 - Настройка параметров пользователя

Предоставление пользователю прав доступа к журналу событий:

1. В консоли Computer Management (Управление компьютером) откройте раздел System Tools (Служебные программы) → Local Users and Groups (Локальные пользователи и группы) → Groups (Группы).
2. Выберите в списке группу Event Log Readers (Читатели журнала событий).
3. Откройте правой кнопкой мыши контекстное меню группы Event Log Readers (Читатели журнала событий) и выберите пункт Add To Group (Добавить в группу). Откроется окно Event Log Readers Properties (Свойства: Читатели журнала событий).
4. Для добавления пользователя в группу:
   - Нажать кнопку Add (Добавить).
   - В открывшемся окне Select Users (Выбор: Пользователи) выбрать в списке ранее созданного пользователя и добавить его в группу, нажав кнопку OK.
5. Для сохранения введенных настроек в окне Event Log Readers Properties (Свойства: Читатели журнала событий) нажмите кнопку OK.

Добавление новой конфигурации в коллектор:

Передача событий в **Платформу Радар** осуществляется через eventlog_collector. Ниже приведены настройки с описанием для добавления в config.yaml:

```
eventlog_collector: &eventlog_collector
  id: "eventlog_collector"
  channel: ['Application']
  query: "*[System[Provider[@Name='Имя экземпляра СУБД>']]]"
  batch_size: 31
  timeout: 3
  poll_interval: 1
  read_from_last: false
  resolve_sid: false
  log_level: "INFO"
  worker_count: 1
  remote:
    enabled: true
    user: "<user_name>"
    password: "<password>"
    domain: "."
    remote_servers: ["<IP-адрес сервера с СУБД>"]
    auth_method: "Negotiate"
  encoding:
    change_to_utf8: true
    original_encoding: "cp1251"
```

> В качестве данных для подключения необходимо использовать созданную ранее учетную запись.
> В поле query мы указываем запрос для получения событий только от настраиваемого источника.

10.1.2. Настройка получения событий через odbc коллектор.

Включение аудита MS SQL Server:

1. Запустите Microsoft SQL Server Management Studio.
2. В окне подключения к базе данных укажите название экземпляра и введите учетные данные (см. рисунок 103).

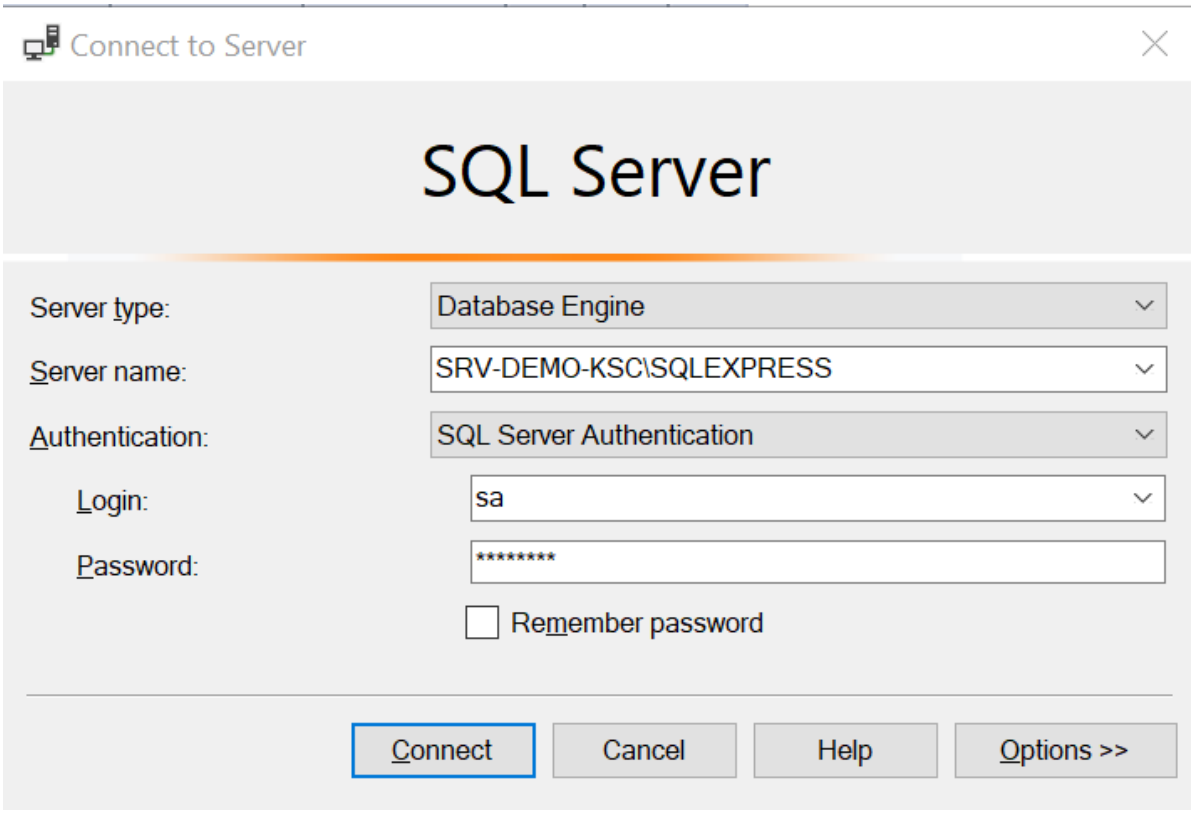

Рисунок 103 - Подключение к базе данных

3. В панели Object explorer перейдите во вкладку Security → Audits. По правому щелчку мыши выберите опцию New Audit... (см. рисунок 104).

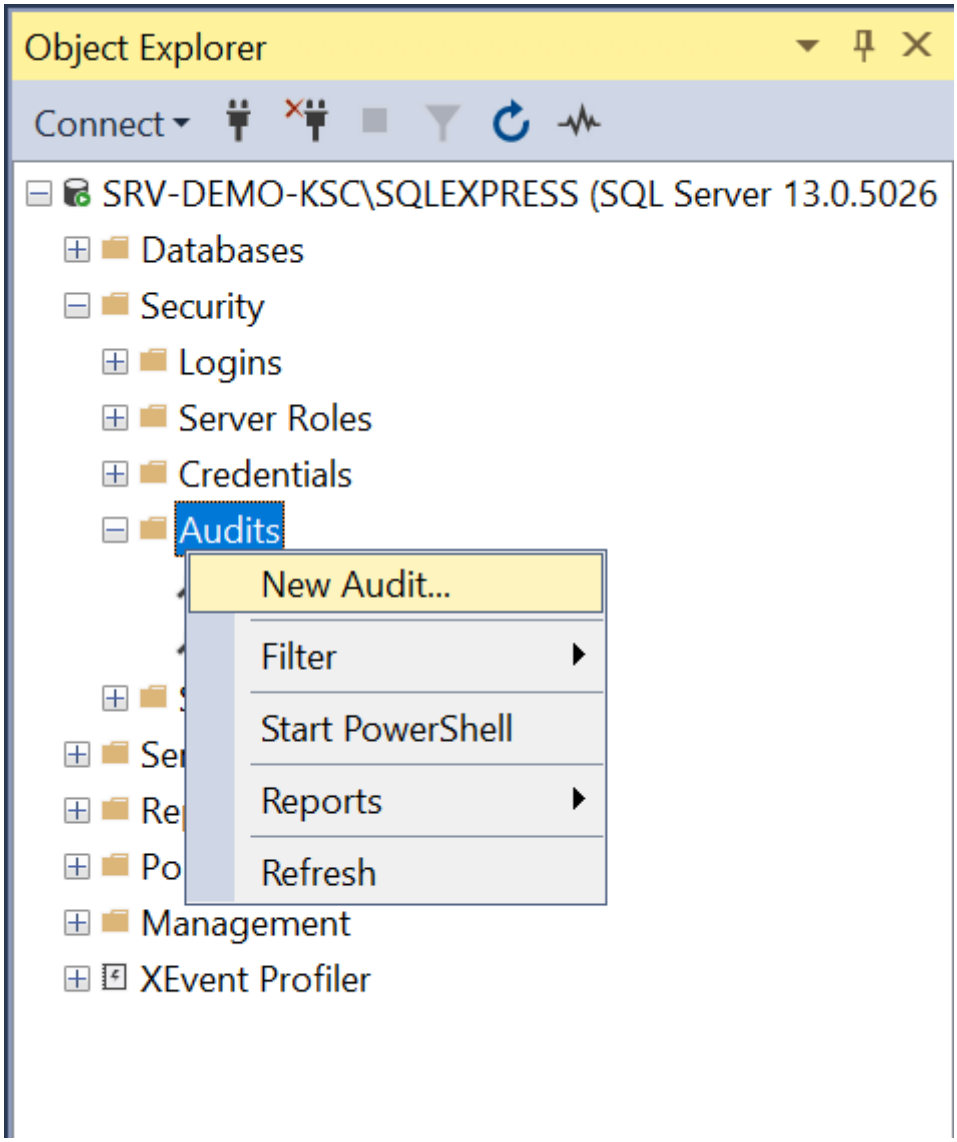

Рисунок 104 - Создание аудита

3. В открывшейся вкладке Create Audit укажите название аудита в поле Audit name. В качестве Audit destination выберите Application Log, нажмите ОК (см. рисунок 105).

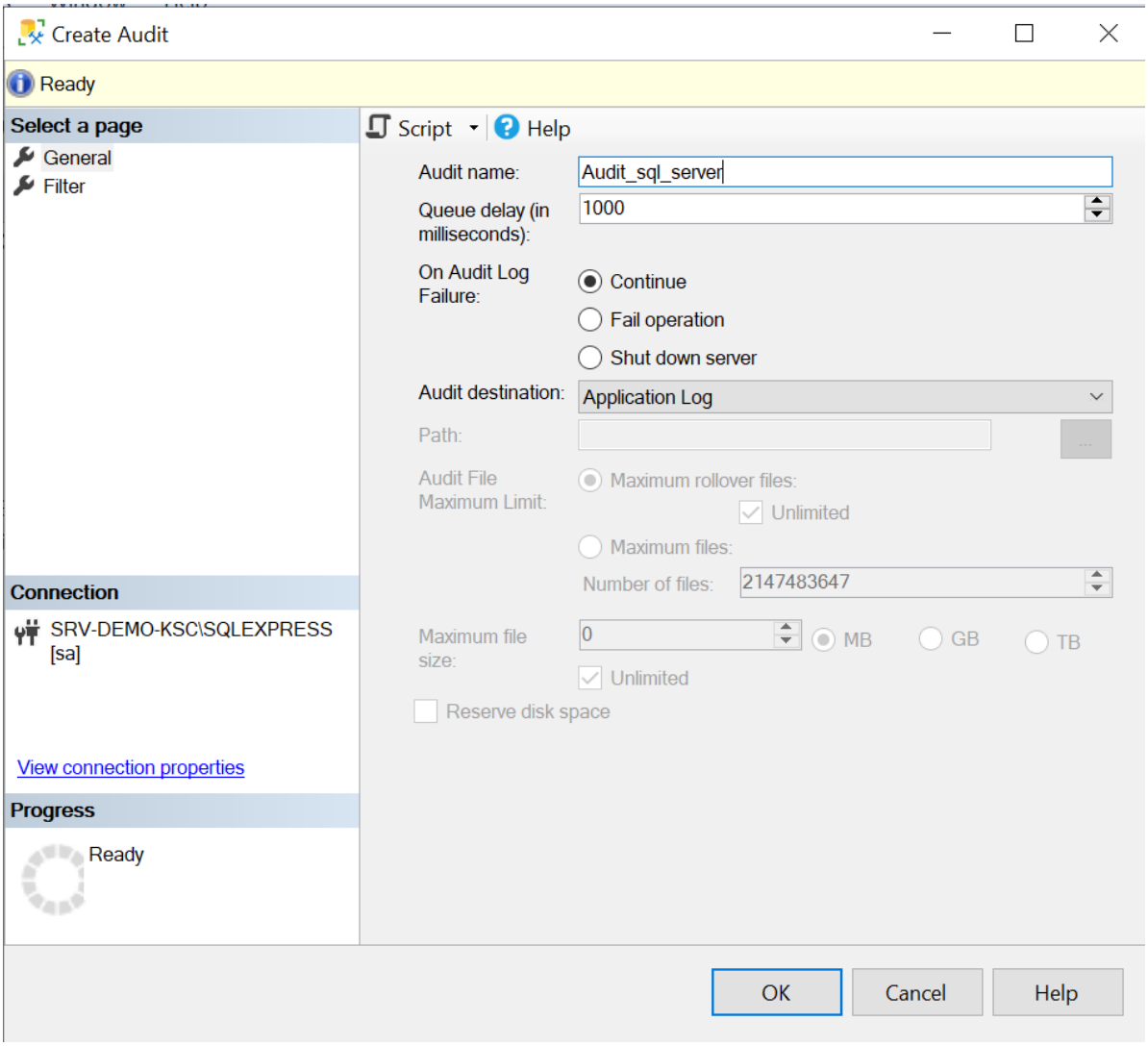

Рисунок 105 - Настройка аудита

4. В панели Object explorer перейдите во вкладку Security → Server Audit Specification. По правому щелчку мыши выберите опцию New Server Audit Specification... (см. рисунок 106).

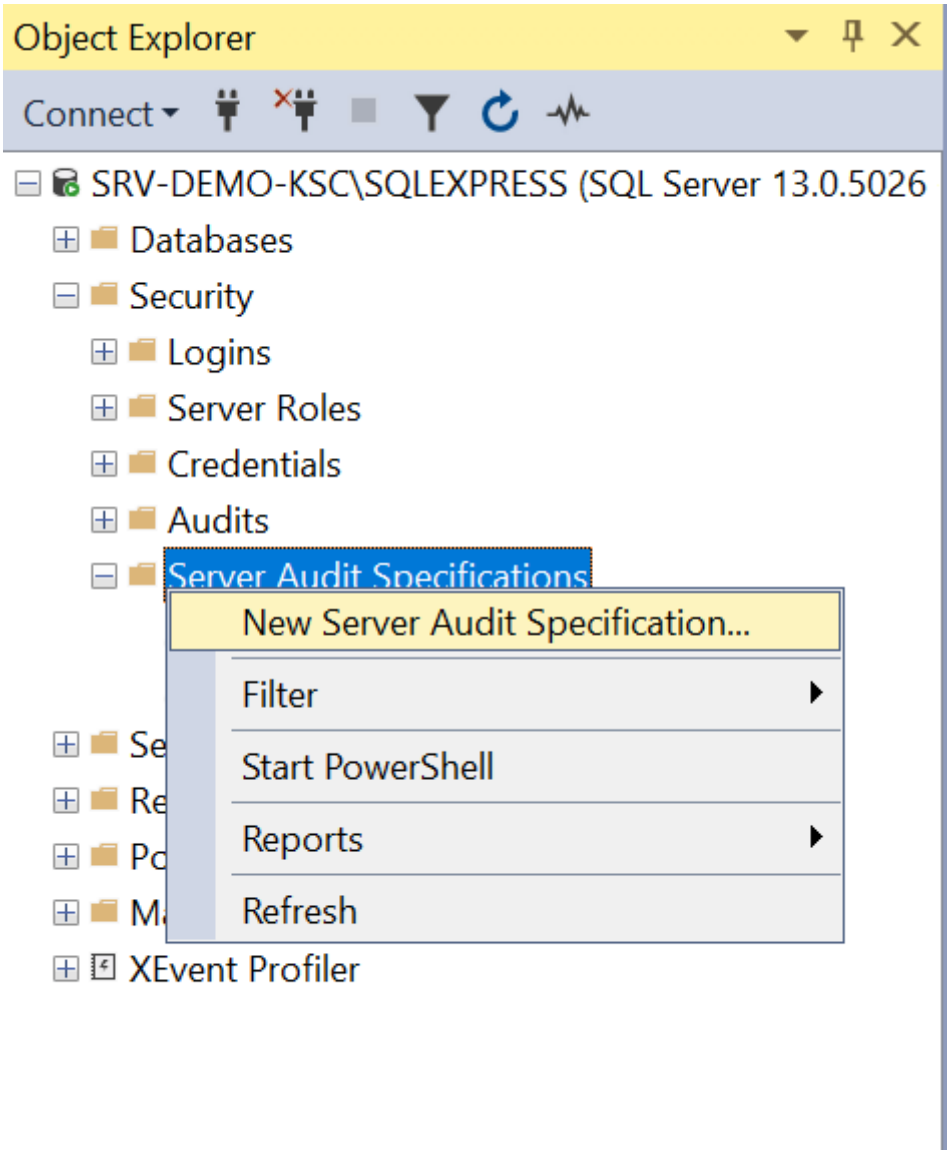

Рисунок 106 - Создание спецификации аудита

5. В открывшейся вкладке Create Server Audit Specification укажите название спецификации аудита в поле Name. В поле Audit выберите ранее созданный аудит из выпадающего списка. В поле Actions выберите типы событий для отслеживания, нажмите OK (см. рисунок 107).

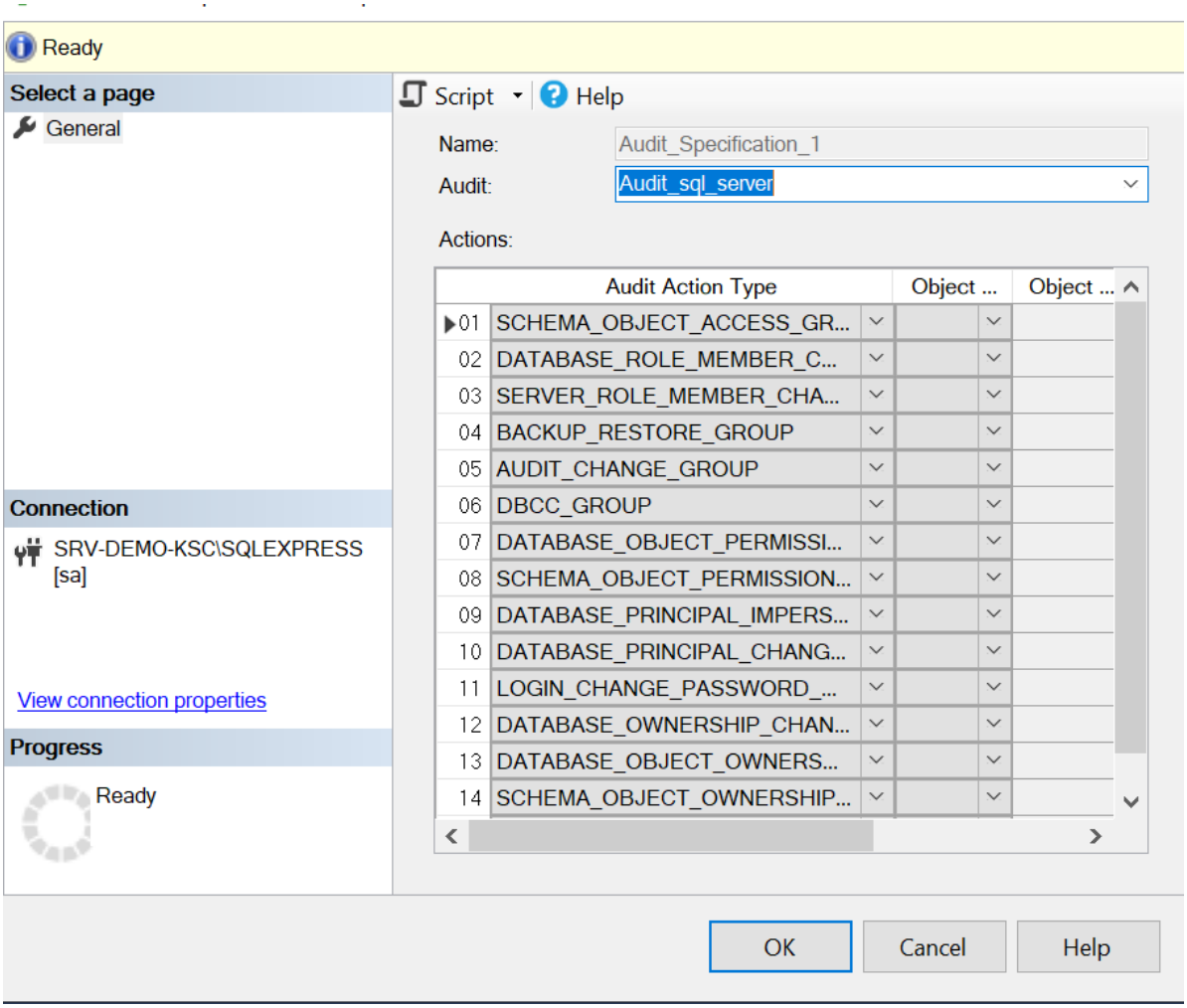

Рисунок 107 - Настройка спецификации аудита

Установка ODBC драйвера:

1. С официального сайта скачайте ODBC Driver for SQL Server.
2. Установите скачанный драйвер на сервер с коллектором.

Добавление новой конфигурации в коллектор:

Передача событий в **Платформу Радар** осуществляется через odbc_collector. Ниже приведены настройки с описанием для добавления в config.yaml:

```
odbc_collector: &odbc_collector
  id: "odbc_collector"
  poll_interval: 5
  read_from_last: true
  connection_string: "server=IP-адрес сервера с СУБД;port=1433;driver={ODBC
Driver 18 for SQL Server};database=master;Encrypt=Optional;UID=<user_name>;PWD=
<Password>"
  sql: >
    SELECT
        CAST(DATEDIFF_BIG(ns, '1970-01-01 00:00:00.0000000', event_time) AS
BIGINT) as epoch,
        event_time,
        action_id,
        succeeded,
        session_id,
        class_type,
```

```
        session_server_principal_name,
        server_principal_name,
        server_principal_sid,
        database_principal_name,
        target_server_principal_name,
        target_server_principal_sid,
        target_database_principal_name,
        server_instance_name,
        database_name,
        schema_name,
        object_name,
        statement,
        additional_information,
        transaction_id
    FROM fn_get_audit_file ('C:\Program Files\Microsoft SQL
Server\MSSQL13.SQLEXPRESS\MSSQL\DATA\*.sqlaudit', default, default)
    WHERE CAST(DATEDIFF_BIG(ns, '1970-01-01 00:00:00.0000000', event_time) AS
BIGINT) > ?;
  bookmark_field: "epoch"
```

В поле connection_string укажите:

- IP-адрес сервера с СУБД
- Порт для подключения к базе данных
- Название драйвера

  Примечание: Название драйвера можно узнать, запустив Administrative Tools → ODBC Data Sources (64-bit) во вкладке Drivers (поле Name)

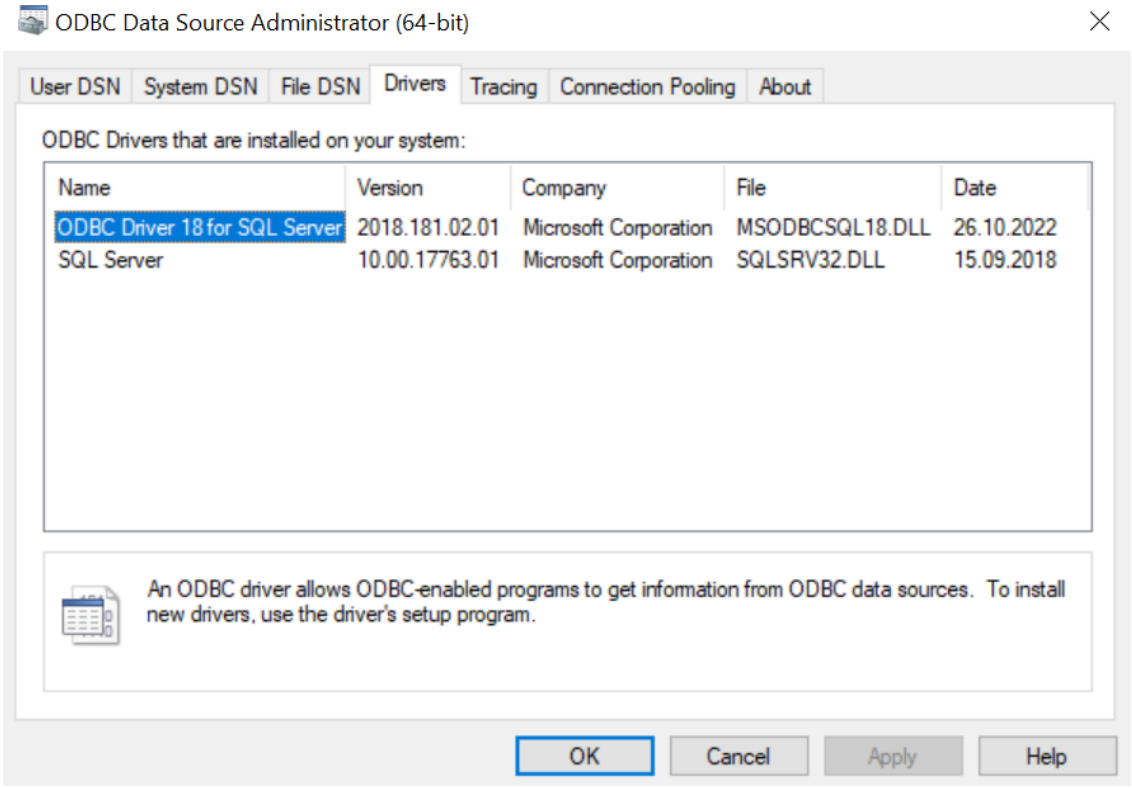

Рисунок 108

- Название базы данных
- Учетные данные для подключения к БД

В разделе с SQL запросом необходимо указать путь к файлам с событиями аудита.

Пример:

`C:\Program Files\Microsoft SQL Server\MSSQL13.SQLEXPRESS\MSSQL\DATA\*.sqlaudit`

В данном случае коллектор будет читать все найденные файлы аудита по адресу C:\Program Files\Microsoft SQL Server\MSSQL13.SQLEXPRESS\MSSQL\DATA\

10.2. PostgreSQL {#postgre}

Для настройки логирования событий из БД PostrgreSQL выполните шаги:

1. В командной строке сервера выполните команду

   ```
   psql -U <username> -c 'SHOW config_file'
   ```

   На выходе будет указан путь к конфигурационному файлу:

   `/var/app/data/postgresql.conf`

2. В конфигурационный файл postgresql.conf (по пути из предыдущей команды) добавьте строки:

   ```
   log_destination = 'syslog'
   logging_collector = off
   syslog_facility = 'LOCAL0'
   syslog_ident = 'postgres'
   syslog_sequence_numbers = on
   syslog_split_messages = off
   client_min_messages = log
   log_min_messages = info
   log_min_error_statement = info
   log_checkpoints = off
   log_connections = on
   log_disconnections = on
   log_duration = off
   log_error_verbosity = default
   log_hostname = on
   log_line_prefix = 'pgmessage: %m %a %u %d %r %i %e '
   log_statement = 'mod'
   lc_messages = 'en_US.UTF-8'
   ```

   Перезапустите службу `postgresql`

3. Настройте Rsyslog для отправки сообщений на коллектор:

   ```
   nano /etc/rsyslog.d/10-pgsql.conf
   if $programname == 'postgres' then @@rsyslog:4000
   ```

   Перезапустите rsyslog

10.2.1. Настройка ODBC PostgreSQL

1. В конфигурационном файле `/var/app/data/postgresql.conf` настройте тип логирования (csvlog) и включите `logging_collector`.

```
log\_destination = 'csvlog'
logging\_collector = on
client\_min\_messages = log
log\_min\_messages = info
log\_min\_error\_statement = info
log\_checkpoints = off
log\_connections = on
log\_disconnections = on
log\_duration = off
log\_error\_verbosity = default
log\_hostname = on
log\_line\_prefix = 'pgmessage: %m %a %u %d %r %i %e '
log\_statement = 'mod'
lc\_messages = 'en\_US.UTF-8'
```

   Перезапустите службу `postgresql`

2. Создайте в нужной БД таблицу для хранения логов.

```
CREATE TABLE postgres_log
(
  log_time timestamp(3) with time zone,
  user_name text,
  database_name text,
  process_id integer,
  connection_from text,
  session_id text,
  session_line_num bigint,
  command_tag text,
  session_start_time timestamp with time zone,
  virtual_transaction_id text,
  transaction_id bigint,
  error_severity text,
  sql_state_code text,
  message text,
  detail text,
  hint text,
  internal_query text,
  internal_query_pos integer,
  context text,
  query text,
  query_pos integer,
  location text,
  application_name text,
  PRIMARY KEY (session_id, session_line_num)
);
```

3. Пример команды для переноса данных из лог-файла в таблицу:

```
COPY postgres_log FROM '/var/app/data/pg_log/postgresql-2020-09-
01_000000.csv' WITH csv;
```

Подробнее о переносе описано в Руководстве PostgreSQL

4. Скачайте и установите драйвер ODBC для PostgreSQL на сервер NXLog.
   Проверьте наличие драйвера и его название, оно пригодится при настройке ConnectionString в ODBC-модуле NXLog

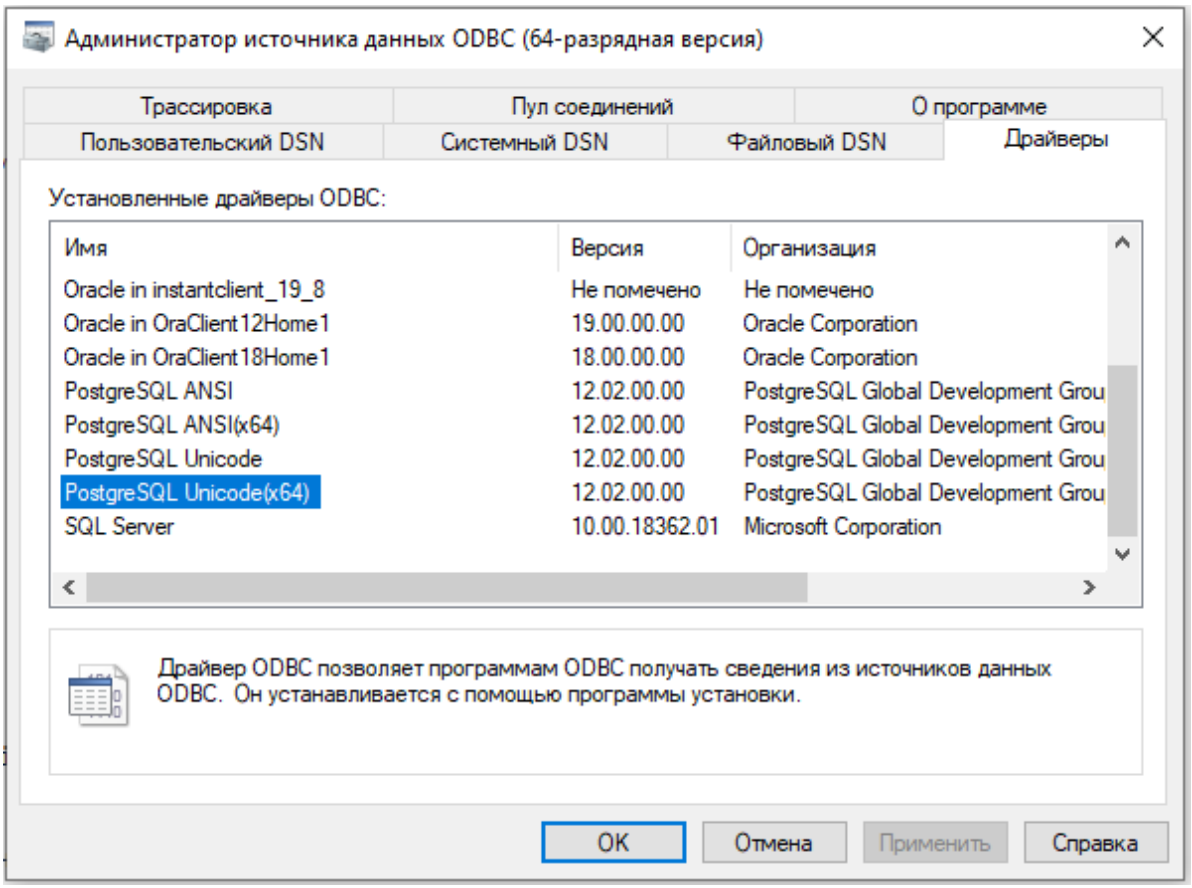

Рисунок 109 - Настройка ODBC

10.2.2. Настройка ODBC-модуля NXLog

Строка для ODBC-подключения:

```
<Input postgres>
Module im\_odbc
ConnectionString Driver={PostgreSQL UNICODE(x64)};Server=<IP or hostname>;Port=
<PostgreSQL port num>;Database=Database\_name;UID=Username;PWD=password;
```

Driver - имя драйвера ODBC из п.5 — PostgreSQL ODBC Driver(UNICODE) или PostgreSQL ODBC Driver(ANSI).
Server - имя сервера PostgreSQL.
Port - порт, используемый для подключения к серверу PostgreSQL (default 5432).
Database - имя базы данных PostgreSQL.
Uid и Pwd - Uid (идентификатор пользователя) и Pwd (пароль) для подключения.

10.3. Oracle Database {#oracle}

Настройка источника Oracle Database на отправку событий с помощью Oracle Audit.

Настройку источника нужно выполнять от имени учетной записи root, поддерживающей в настраиваемом экземпляре СУБД роль sysadmin с привилегиями sysdba и sysoper:

1. Подключитесь с помощью `sqlplus` локально, выполнив команду:

   `sqlplus / as sysdba`

2. Выполните команду:

   `alter session set "_ORACLE_SCRIPT"=true;`

3. Проверьте параметры аудита командой:

   `show parameter audit`

4. Выполните команду установки журнала аудита OS:

   `ALTER SYSTEM SET audit_trail=OS SCOPE=SPFILE;`

5. Выключите СУБД командой:

   `Shutdown`

6. Включите СУБД командой:

   `Startup`

7. Проверьте параметры аудита командой:

   `show parameter audit`

   Убедитесь, что audit_trail принял значение OS.
   Запишите значение `audit_file_dest`, оно понадобится при настройке отправки сообщений для параметра `File`.

8. Выполните команду:

   `ALTER SYSTEM SET audit_sys_operations=true SCOPE=SPFILE;`

9. Установите важность событий командой:

   `alter system set audit_syslog_level='local5.info' scope=spfile sid='*';`

10. Выполните команду

    `ALTER SYSTEM SET audit_trail=DB, EXTENDED SCOPE=SPFILE;`

11. Выполните последовательно команды:

    ```
    Shutdown
    Startup
    show parameter audit
    ```

    На выходе должны появиться сообщения (см. рисунок 108).

```
SQL> show parameter audit

NAME                                 TYPE        VALUE
------------------------------------ ----------- ------------------------------
audit_file_dest                      string      /u02/app/oracle/audit/ORCLCDB
audit_sys_operations                 boolean     TRUE
audit_syslog_level                   string      LOCAL5.INFO
audit_trail                          string      OS
unified_audit_sga_queue_size         integer     1048576
```

Рисунок 110 - Вывод сообщений

12. Донастройте параметры аудита:

```
AUDIT ALTER SYSTEM BY ACCESS;
AUDIT DELETE ON SYS.AUD$ BY ACCESS;
AUDIT DELETE ON SYS.FGA_LOG$ BY ACCESS;
AUDIT EXECUTE ON SYS.DBMS_FGA BY ACCESS;
AUDIT INSERT ON SYS.AUD$ BY ACCESS;
AUDIT INSERT ON SYS.FGA_LOG$ BY ACCESS;
AUDIT SELECT ON SYS.DBA_USERS BY ACCESS;
AUDIT SELECT ON SYS.LINK$ BY ACCESS;
AUDIT SELECT ON SYS.USER_DB_LINKS BY ACCESS;
AUDIT SELECT ON SYS.USER_HISTORY$ BY ACCESS;
AUDIT SYSTEM AUDIT BY ACCESS;
AUDIT TABLE BY ACCESS;
AUDIT UPDATE ON SYS.AUD$ BY ACCESS;
AUDIT UPDATE ON SYS.FGA_LOG$ BY ACCESS;
```

13. Для отправки событий через Rsyslog:
    - создайте файл с конфигурацией для rsyslog:

      `nano /etc/rsyslog.d/oracle_audit.conf`
    - настройте чтение из файла:

      ```
      input(type="imfile"  File="<значение audit_file_dest  из п.7>.xml"
      PersistStateInterval="100"
      Tag="oracle_audit_trail:"
      Severity="info"
      Facility="local5"
      startmsg.regex="<AuditRecord>"
      )
      local5.* @@<ip-address лог-коллектора>:2770
      ```
    - перезапустите сервис rsyslog

      `sudo service rsyslog restart`

10.4. Oracle MySQL {#mysql}

Для настройки источника Oracle MySQL выполните следующие шаги:

1. Установите модуль аудита MariaDB, последовательно выполнив команды:

   ```
   wget http://mirror.mephi.ru/mariadb/mariadb-10.1.45/bintar-linux-
   x86_64/mariadb-10.1.45-linux-x86_64.tar.gz
   sudo tar -xzf mariadb-10.5.5-linux-x86_64.tar.gz
   sudo install mariadb-10.1.45-linux-x86_64/lib/plugin/server_audit.so
   /usr/lib/mysql/plugin
   sudo install mariadb-10.5.5-linux-x86_64/lib/plugin/server_audit.so
   /usr/lib/mysql/plugin
   Sudo mysql
   INSTALL PLUGIN server_audit SONAME 'server_audit.so';
   SHOW PLUGINS;
   Set Global server_audit_logging=on;
   EXIT;
   ```

2. Настройте аудит, последовательно выполнив команды:

```
sudo nano /etc/mysql/mysql.conf.d/mysqld.cnf
Добавляем настройки
plugin-load=server_audit=server_audit.so
server_audit_logging=on
server_audit_events=connect,query,table,query_ddl,query_dml,query_dcl
server_audit_output_type = SYSLOG
server_audit_syslog_facility = LOG_SYSLOG
server_audit_file_path = /var/log/mysql/audit.log
```

3. Перезапустите сервис MySQL:

   `service mysql restart`

4. Запустите консоль MySQL с правами суперпользователя:

   `# mysql -u root -p`

   Вы можете просмотреть значения параметров модуля аудита, выполнив команду

   ```
   ```
   SHOW VARIABLES LIKE '%audit%';
   EXIT;
   ```
   ```

   Результат: Модуль аудита настроен.

Далее необходимо настроить rsyslogd. Для этого:

1. Создайте файл с конфигурацией для rsyslog:

   `sudo nano /etc/rsyslog.d/20-mysql.conf`

2. Запишите в созданный файл следующие значения:

   ```
   template (name="radar" type="string"
   string="<%PRI%>%TIMESTAMP:::date-rfc3339% %HOSTNAME%
   %syslogtag%%$.suffix%%msg:::sp-if-no-1st-sp%%msg%")
   :syslogtag, contains, "mysql" @@<ip-adsress Лог-коллектора>:4005;radar
   ```

3. Перезапустите службу rsyslog:

   `sudo service rsyslog restart`

10.5. Orcale NetListener {#netlistener}

Для настройки источника Oracle NetListener выполните следующие шаги:

1. Запустите LSNRCTL командой:

   `LSNRCTL`

2. Определите экземпляр используемой службы Oracle NetListener командой:

   `show current_listener`

3. После выполнения команды отобразится имя экземпляра СУБД.

4. Для смены используемого экземпляра используется команда:

   `set current_listener.`

5. Проверьте статус журналирования:

`show log_status`

6. Если для параметра log_status указано OFF, включите журналирование:

   `set log_status on save_config reload`

7. Для отправки событий через rsyslog, узнайте путь к лог-файлам командой:

   `show log_directory`

   Он понадобится для следующего этапа настройки в параметре File.

8. Создайте конфигурационный файл для rsyslog:

   `sudo nano /etc/rsyslog.d/oracle_netlistener.conf`

9. Настройте чтение из файла:

```
module(load="imfile" mode="inotify") #PollingInterval="10") #mode="inotify")
input(type="imfile"
File="/<параметр File из п.4 >/log.xml"
PersistStateInterval="100"
Tag="oracle_netlistener:"
Severity="info"
Facility="local3"
readMode="2"
)local3.* @@<Ip-address лог-коллектора>:2771
```

10. Перезапустите сервис rsyslog:

    `sudo service rsyslog restart`

11. WEB-серверы

11.1. Apache HTTP server {#apachehttp}

Для отправки событий стандартного логирования источника Apache-http-server выполните шаги:

1. Проверьте в файле, расположенном по пути `/etc/apache2/apache2.conf`, наличие записи `LogLevel info`.

   > Поддерживаемый уровень корректной обработки событий **Платформой Радар** - уровень `warn`,
   > при необходимости можно изменить на `LogLevel warn`.

2. После внесения изменений в файл `/etc/apache2/apache2.conf` обновите сервис `apache`, для этого выполните команду `systemctl reload apache2.service`
   и проверьте состояние сервиса командой `systemctl status apache2.service`.

3. Далее создайте файл `apache2.conf` с настройками отправки событий через rSyslog в **Платформу Радар** по пути `/etc/rsyslog.d/` со следующим содержимым:

```
# Apache2 logs
input(type="imfile"
      File="/var/log/apache2/access.log"
      Tag="apache2-accesslog"
      Severity="warn"
```

```
      Facility="local2")

input(type="imfile"
      File="/var/log/apache2/error.log"
      Tag="apache2-errorlog"
      Severity="warn"
      Facility="local3")

local2,local3.*  @@<collector_ip>:2830
```

4. Перезапустите rSyslog командой

   `systemctl restart rsyslog`

   и проверьте состояние rSyslog командой `systemctl status rsyslog`.

5. В конфигурационный файл лог-коллектора добавьте стандартную настройку `tcp-input/tcp-output` эквивалентную нижеуказанной настройке:

```
tcp_input: &tcp_input
  id: "tcp_input"
  host: "0.0.0.0"
  port: 2830
  sock_buf_size: 0
  format: "json"
  log_level: "INFO"

tcp_output: &tcp_output
  id: "tcp_output"
  target_host: "<Ip-адрес-Платформы-Радар>"
  port: 2830

collectors:
  tcp_receiver:
    - <<: *tcp_input

senders:
  port: 48003
  tcp:
    - <<: *tcp_output

route_1: &route_1
  collector_id:
    - "tcp_input"
  sender_id:
    - "tcp_output"

routers:
  - <<: *route_1
```

6. Перезапустите сервис лог-коллектора для принятия изменений.
7. Проверьте наличие событий в графическом интерфейсе **Платформы Радар** через вкладку *Просмотр событий*.

11.2. Apache Tomcat {#tomcat}

Apache Tomcat - это служба, которая может использоваться в следующих сценариях:

- в качестве самостоятельного веб-сервера;
- в качестве контейнера сервлетов вместе с Glassfish, JBoss;
- в качестве сервера контента, в связке с Apache HTTP Server.

По умолчанию, Apache Tomcat выполняет журналирование с помощью обработчика `java.util.logging`, параметры которого заданы в файле `${catalina.base}/conf/logging.properties` (здесь `${catalina.base}` - это директория, в которую установлен Tomcat). Если используются стандартные настройки, распределение событий Tomcat происходит по нескольким файлам, в зависимости от типа:

- `catalina.out и catalina.$(date).log` - лог контейнера сервлетов, основные события, произошедшие с ядром Tomcat;
- `localhost.$(date).log` - лог событий локального экземпляра Tomcat, в который, как правило, сохраняются основные внутренние ошибки;
- `localhost_access_log.$(date).txt` - журнал запросов (access log), эквивалентный журналу службы httpd (параметры доступа определяются в файле `${catalina.base}/conf/server.xml)`;
- журналы `manager.$(date).log и host-manager.$(date).log` - журналы работы веб-приложений, функционирующих в составе Tomcat.

Передачу журналов Apache Tomcat коллектору **Платформы Радар** можно организовать с помощью службы RSyslog. Для получения событий Tomcat необходимо настроить отправку содержимого журналов `catalina.out` и `localhost_access_log*.txt`.

Предположим, что отправка журналов будет выполнятся с использованием facility "local1","local2". Переменная `${catalina.base}` в нашем примере будет иметь значение `/opt/tomcat/`. Для настройки отправки событий, в директории `/etc/rsyslog.d/` создайте файл tomcat.conf со следующим содержимым:

```
# Apache Tomcat logs
input(type="imfile"
      File="/opt/tomcat/logs/localhost_access_log*.txt"
      Tag="catalina-access"
      Severity="info"
      Facility="local1")

input(type="imfile"
      File="/opt/tomcat/logs/catalina.out"
      Tag="catalina-out"
      Severity="info"
      Facility="local2")

local1,local2.*  @@<collector_ip>:<collector_port>
```

> **Важно:** вместо <collector_ip> необходимо указать адрес хоста с установленным коллектором **Платформы Радар**, а вместо <collector_port> - сетевой порт для приема событий от источника. Два символа '@@' означают, что отправка будет производиться по протоколу TCP.

> **Важно:** в случае, если Tomcat запущен из-под выделенной учетной записи, необходимо предоставить соответствующие права на чтение для каталога с журналами. В противном случае, служба rsyslog сообщит об ошибке чтения файла. Также, в конфигурации Tomcat для файлов журналов должно быть задано корректное значение UMASK (0022).

Разбор параметров:

- Type - задает название модуля,
- File - задает абсолютный путь до журналов Tomcat;
- Tag - идентификатор для журналов, которые будут отправляться коллектору SIEM;
- Severity - задает важность отправляемых сообщений;
- Facility - задает идентификатор для источника событий на сервере.

После настройки файла tomcat.conf, следует проверить содержимое файла rsyslog.conf. Для модуля imfile должна быть задана частота опроса журналов, а сам модуль должен быть загружен:

`module(load="imfile" PollingInterval="10")`

После внесения изменений, следует перезагрузить службу rsyslog и проверить результат:

```
# 12. systemctl stop rsyslog
# 13. systemctl start rsyslog
```

Конфигурация коллектора логов **Платформы Радар** для приема событий должна иметь настройки, эквивалентные следующим:

```
tcp_input: &tcp_input
  id: "tcp_input"
  host: "0.0.0.0"
  port: 514
  sock_buf_size: 0
  format: "json"
  log_level: "INFO"

tcp_output: &tcp_output
  id: "tcp_output"
  target_host: "<log_collector_ip>"
  port: <log_source_port>
  sock_buf_size: 0
  log_level: "INFO"
  ssl_enable: false
  require_cert: false
  ssl_compression: false
  batch_mode_enable: false

collectors:
  tcp_receiver:
    - <<: *tcp_input

senders:
  port: 48003
  tcp:
    - <<: *tcp_output
```

```
route_1: &route_1
  collector_id:
    - "tcp_input"
  sender_id:
    - "tcp_output"

routers:
  - <<: *route_1
```

Nginx {#nginx}

Для подключение веб-сервера Nginx в качестве источника событий выполните шаги:

1. Для настройки внешнего веб-сервера nginx на отправку событий в **Платформу Радар** необходимо проверить файл с настройками по пути `/etc/nginx/nginx.conf` и файл настройки по пути `/opt/pangeoradar/configs/nginx.conf`.
2. В настройке `/*/*/*nginx.conf` в поле `# Logging Setting` должны быть указаны пути для регистрации событий *Access.log* или *Error.log*. Если поле с данными параметрами отсутствует, то данную конструкцию добавьте вручную.

   ```
   ##
   ```

1. Logging Settings

```
##
      access_log /var/log/nginx/access.log;
      error_log /var/log/nginx/error.log;
  ```
```

3. Проверьте в файле `/etc/rsyslog.conf` наличие поля `# Include all config files in /etc/rsyslog.d/`. Если оно отсутствует, то добавьте следующее значение:

   ```
   ```

1. Include all config files in /etc/rsyslog.d/

```
       $IncludeConfig /etc/rsyslog.d/*.conf
```
```

4. Создайте файл `/etc/rsyslog.d/nginx.conf` со следующим содержимым:

```
input(type="imfile"
      File="/var/log/nginx/access.log"
      Tag="nginx-access"
      Severity="info"
      Facility="local0")

input(type="imfile"
      File="/var/log/nginx/error.log"
      Tag="nginx-error"
      Severity="warn"
      Facility="local1")

local0,local1.* @@<Ip-адрес  лог-коллектора>:2960
```

> через @ задается отправка по протоколу «UDP» с указанием адреса машины с лог-коллектором и необходимым портом для отправки.

> через @@ задается отправка по протоколу «TCP» с указанием адреса машины с лог-коллектором и необходимым портом для отправки

Шаблоны:

для отправки по TCP

`local0,local1.* @@<Ip-адрес  лог-коллектора>:<порт-лог-коллектора>`

для отправки по UDP

`local0,local1.* @<Ip-адрес  лог-коллектора>:<порт-лог-коллектора>`

5. На машине с лог-коллектором добавьте настройку для получения событий от источника и отправки их в **Платформу Радар**

```
#####################################################################
          Часть настройки лог-коллектора
#####################################################################
```

1. Так как в 4м пункте был выбран шаблон отправки по TCP, поэтому настройка на лог-коллекторе соответствует протоколу TCP

```
tcp_input: &tcp_input
    id: " tcp_input"
  host: "0.0.0.0"
    port: 2960
    sock_buf_size: 0
  format: "json"
    log_level: "INFO"

    tcp_output: &  tcp_output
  id: "  tcp_output"
    target_host: "<ip адрес Платформы Радар>"
```

```
  port: 2960
    sock_buf_size: 0
    log_level: "INFO"

senders:
  port: 48002
    tcp:
    - <<: *  tcp_output
collectors:
    tcp_receiver:
    - <<: *tcp_input

route_1: &route_1
    collector_id:
    - "tcp_input"
    sender_id:
    - "tcp_output"
routers:
  - <<: *route_1
```

6. В **Платформе Радар** включите «Тип Источника» «Nginx-Web-server» и нажмите кнопку «Синхронизировать».
7. Проверьте приходящие события в «Просмотр событий».

2. Системы контроля привилегированного доступа

2.1. Подключение источника Solar Dozor {#solardozor}

Solar Dozor - продукт класса DLP (Data Leak Prevention) российской компании Ростелеком-Солар. Настройки подключения источника тестировались с **Платформой Радар** версии 3.5.1 и Solar Dozor 7.9.0-760, установленным под управлением Red Hat Enterprise Linux 7.9.

Для настройки необходимо выполнить шаги из инструкции ниже.

2.1.1. Настройка отправки с использованием Rsyslog

2.1.1.1. Регистрация действий пользователей в веб-интерфейсе системы

В веб-интерфейсе продукта перейдите на вкладку "Система" > "Конфигурация" > "Расширенные настройки" > "Интерфейс" > "Вебсервер (webserver.conf)".

Установите следующие параметры:

- Запись журналов действий в syslog в формате CEF (action-cef).

Сохраните и примените настройки (см. рисунок 111).

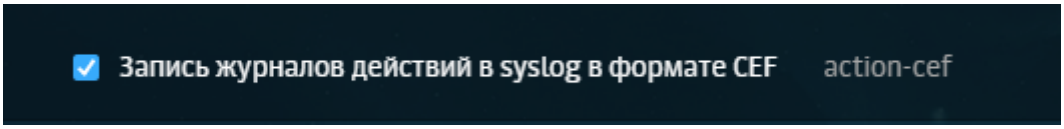

Рисунок 111 - Включение записи в журналы

Данная настройка обеспечивает запись действий пользователей в веб-интерфейсе системы в системный журнал `/var/log/messages` в формате CEF.

Как правило, веб-интерфейс Solar Dozor (служба “webserver”) размещается на узле DLP-системы с ролью “Мастер-сервер”. Для настройки пересылки лога в SIEM-систему следует добавить в `/etc/rsyslog.conf` мастер-сервера следующие строки:

```
$ActionQueueFileName SIEMForwarder
$ActionQueueMaxDiskSpace 1g
$ActionQueueSaveOnShutdown on
$ActionQueueType LinkedList
$ActionResumeRetryCount -1
if $msg contains 'CEF' then @@<pangeo-log-collector-ip>:<port>
```

2.1.1.2. Регистрация событий в журнал Rsyslog

Для включения регистрации событий перейдите в раздел веб-интерфейса "Система" > "Конфигурация" > "Расширенные настройки" > “События и инциденты” > “Сервис хранения и индексации событий и инцидентов (settings.json)”.

Установите следующие параметры (см. рисунок 112):

- Журналировать в syslog регистрацию событий (syslog-events-config) - Включено;
- Журналировать в syslog изменение статуса событий и инцидентов (syslog-events-change);
- Журналировать в syslog в формате CEF (syslog-events-cef).

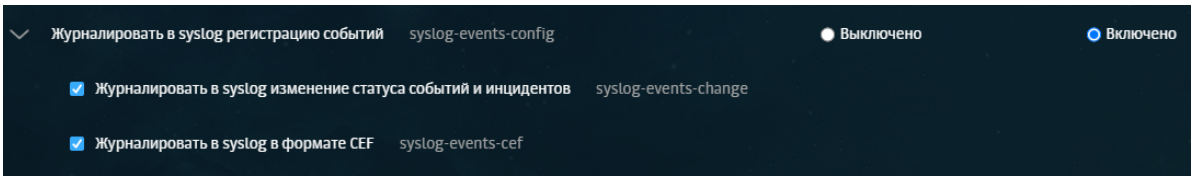

Рисунок 112 - Включение регистрации событий

Сохраните и примените настройки.

2.1.1.3. Журналирование действий над сообщениями

Для включения журналирования действий над сообщениями в веб-интерфейсе продукта перейдите на вкладку "Система" > "Конфигурация" > "Расширенные настройки" > “Обработка сообщений” > “Сервис фильтрации сообщений (mailfilter.edn)”.

Установите следующие параметры:

- Журналировать операции над сообщениями в файл (message-log-file);
- Использовать формат CEF при журналировании операций над сообщениями.

Сохраните и примените настройки (см. рисунок 113).

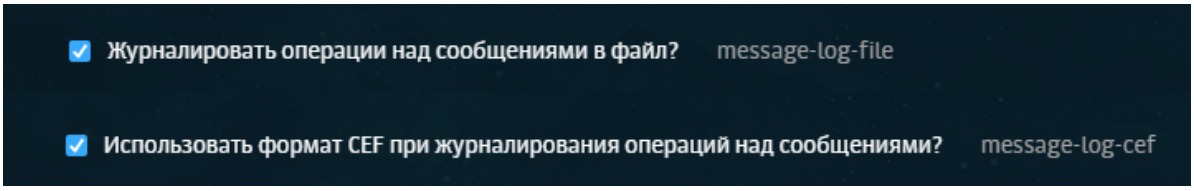

Рисунок 113 - Включение журналирования над сообщениями

В случае активации данных настроек на всех узлах с ролью “Фильтр почтового потока” (mailfilter) будет создан файл, содержащий записи действий над сообщениями - `/opt/dozor/var/log/message-stat.log`.

Необходимо обеспечить передачу содержимого файла в SIEM-систему. Для этого создайте файл конфигурации syslog (пример) - `/etc/rsyslog.d/04-send_dozor_mail.conf`:

```
module(load="imfile" PollingInterval="10")

input(type="imfile"
      reopenOnTruncate="on"
      File="/opt/dozor/var/log/message-stat.log"
      Tag="solar-dozor-mail"
      )

$template rawSmap,"<%PRI%>%TIMESTAMP% %HOSTNAME% %syslogtag%%msg%\n"
if $msg contains 'CEF' then @@<pangeo-log-collector-ip>:<port>;rawSmap
```

здесь - это адрес лог-коллектора, а - номер порта, предназначенного для приема событий. Отправка будет выполняться по протоколу TCP.

2.1.1.4. Настройка ротации журнала действий над сообщениями

Для предотвращения переполнения дискового пространства настройте конфигурацию logrotate. Настройка выполняется на всех узлах Solar Dozor с ролью “Фильтр почтового потока”.

Создайте файл `/etc/logrotate.d/smap-maillog` со следующим содержимым:

```
/opt/dozor/var/log/message-stat.log {
    weekly
    rotate 4
    missingok
    notifempty
    nomail
    compress
    create 0644 dozor dozor
    minsize 50M
}
```

Выполните проверку условия logrotate с помощью команды:

```
logrotate -df /etc/logrotate.d/smap-maillog
```

Запуск ротации вручную выполняется следующей командой:

```
logrotate -f /etc/logrotate.d/smap-maillog
```

2.1.2. Пример конфигурации PANGEO-LOG-COLLECTOR

```
tcp_input3: &dozor_input
  id: "dozor_input"
  host: "0.0.0.0"
  port: 516
```

```
  sock_buf_size: 0
  format: "json"
  buf_size: 16384
  log_level: "INFO"

tcp_output3: &dozor_output
  id: "dozor_output"
  target_host: "192.168.2.124"
  port: 2593
  sock_buf_size: 0
  log_level: "INFO"

collectors:
  tcp_receiver:
    - <<: *dozor_input

senders:
  port: 48003
  tcp:
    - <<: *dozor_output

route_1: &route_1
  collector_id:
    - "dozor_input"
  sender_id:
    - "dozor_output"

routers:
  - <<: *route_1
```

2.2. Staffcop Enterprise {#staffcop}

2.2.1. Включение системной политики Syslog-коннектор

Включение данной политики позволяет выводить информацию попавшую под политику в системный журнал - `/var/log/syslog`.

1. Перейдите во вкладку «*Фильтры - Политики - Системные политики*» (см. рисунок 114).

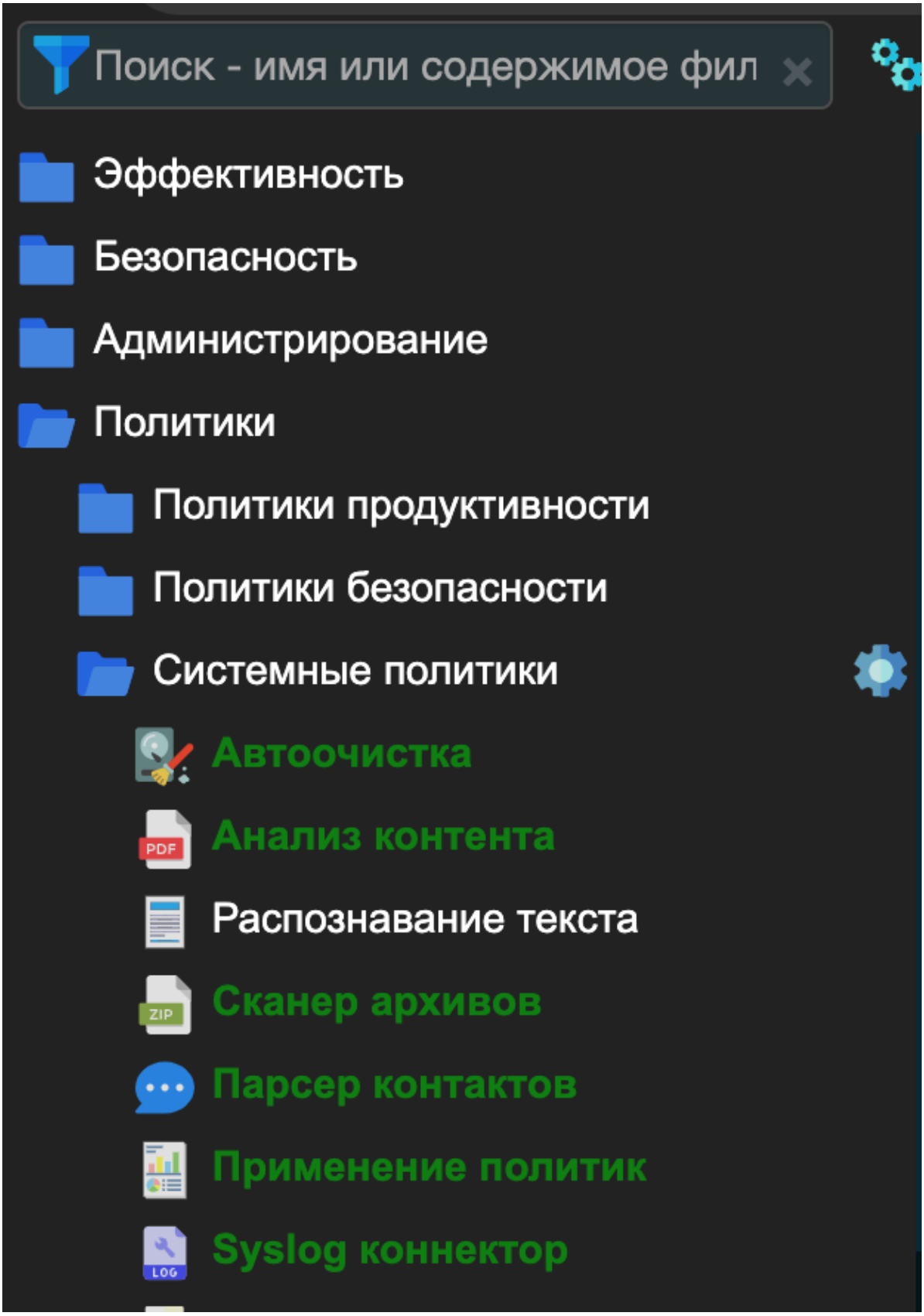

Рисунок 114 - Системные политики.

2. По левому щелчку мыши по полю *Syslog-коннектор* откройте редактирование политики.
3. Во вкладке *Фильтр* задайте необходимые параметры для событий, которые вы хотите видеть в системе (см. рисунок 115).

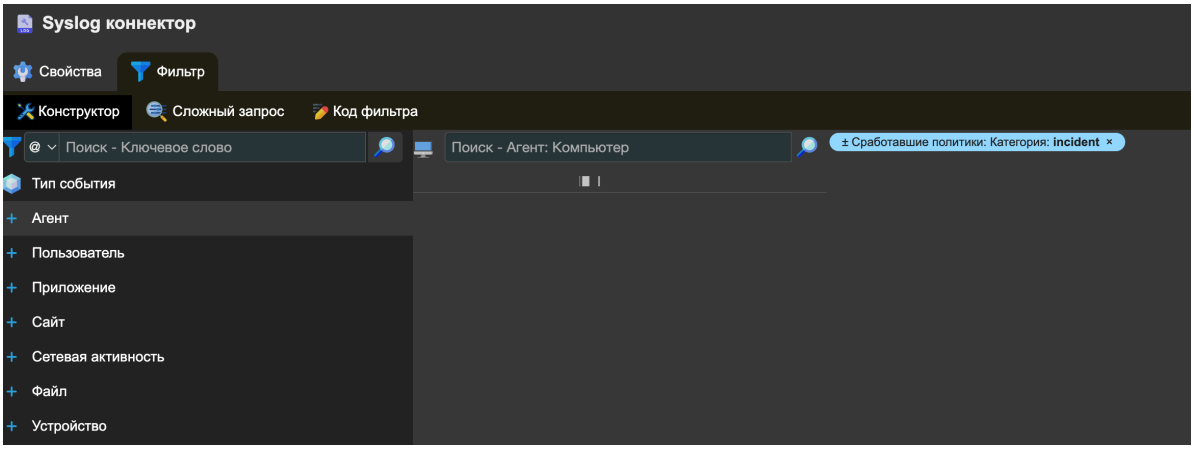

Рисунок 115 - Параметры для событий.

4. Во вкладке *Свойства* отметьте галочкой пункты *Политика активна*, *Формат логов: CEF*.
5. Примените только к новым или ко всем предыдущим событиям, сохраните изменения (см. рисунок 116).

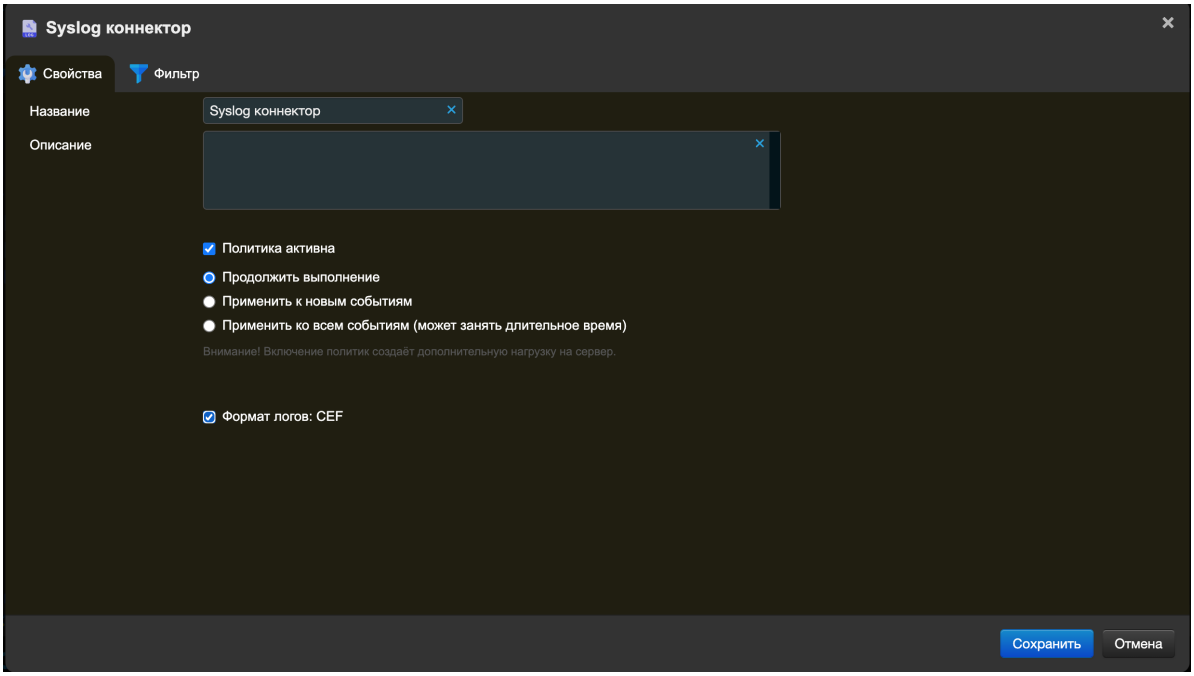

Рисунок 116 - Сохранение изменений.

Выбранные события раз в 5 минут будут помещаться в журнал /var/log/syslog.

2.2.2. Настройка rsyslog

На сервере StaffCop выполните следующие команды:

1. Проверьте наличие и активность службы rsyslog:

   `service rsyslog status`

   По умолчанию служба должна быть установлена и запущена

```
root@enterprise:~# service rsyslog status
● rsyslog.service - System Logging Service
   Loaded: loaded (/lib/systemd/system/rsyslog.service; enabled; vendor preset: enabled)
   Active: active (running) since Fri 2023-03-03 09:39:13 MSK; 1 weeks 3 days ago
     Docs: man:rsyslogd(8)
           http://www.rsyslog.com/doc/
 Main PID: 14748 (rsyslogd)
    Tasks: 4 (limit: 4659)
   CGroup: /system.slice/rsyslog.service
           └─14748 /usr/sbin/rsyslogd -n
```

Рисунок 117

2. Создайте и откройте для редактирования конфигурационный файл `50-siem.conf`

   `nano /etc/rsyslog.d/50-siem.conf`

3. Пропишите в файл следующие настройки (заменив ip-адрес из примера на адрес **Платформы Радар**):

   `If $programname=='staffcop' then @@10.10.10.10:514`

4. Перезапустите службу rsyslog

   `service rsyslog restart`

2.2.3. Добавление новой конфигурации в коллектор

Приведенные настройки с описанием для добавления в config.yaml ниже:

```
tcp_input: &tcp_input
  id: "tcp_input"
  host: "0.0.0.0"
  port: 514
  sock_buf_size: 0
  format: "json"
tcp_output: &tcp_output
  id: "tcp_output_3"
  target_host: "10.10.10.10"
  port: 2512
```

> **В поле `target host` необходимо указать ip-адрес вашей платформы.**

3. Proxy-серверы

3.1. Подключение источника Solar webProxy {#solar}

Solar webProxy - продукт класса SWG (Secure Web Gateway) российской компании Ростелеком-Солар.

Для настройки необходимо выполнить несколько шагов (дополнительно процедура настройки описана в руководстве по установке и настройке).

3.1.1. Настройка журналирования службы веб-интерфейса пользователя (smap-play-server)

Данная настройка позволяет журналировать действия администраторов в веб-интерфейсе системы Solar webProxy. События по умолчанию сохраняются в файл `/var/log/messages` на узле с ролью "Сервер управления".

Пример событий:

```
Mar 29 11:59:48 swp01-main java: webserver: admin@/192.168.11.2: get filter hosts
[swp01-filter.test.lab,swp01-reverse.test.lab]
Mar 29 12:00:06 swp01-main java: webserver: admin@/192.168.11.2: get list of all
categories
Mar 29 12:00:06 swp01-main java: webserver: admin@/192.168.11.2: Action: 'read
layer'; Layer: 'Вскрытие HTTPS'
Mar 29 12:00:10 swp01-main java: webserver: admin@/192.168.11.2: get list of all
categories
Mar 29 12:00:22 swp01-main java: webserver: admin@/192.168.11.2: get list of all
categories
```

1. В веб-интерфейсе продукта перейдите в раздел "Система" > "Основные настройки" > "Журналирование" > "Сервер веб-интерфейса" и установите флажок "Журналировать действия пользователей в syslog". Затем сохраните и примените конфигурацию (см. рисунок 118).

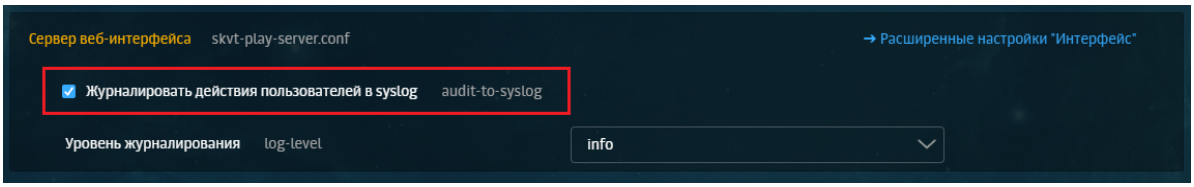

Рисунок 118 - Журналирование действий пользователей

2. Далее отредактируйте файл `/etc/rsyslog.conf`, добавив в него следующую строку (в качестве разделителя между `local0.*` и `/var/log/messages` необходимо использовать символ табуляции):

```
local0.*            /var/log/messages
```

3. Затем, для отправки событий в **Платформу Радар**, на узле с ролью "Сервер управления" создайте файл конфигурации:

   `/etc/rsyslog.d/03-send_skvt_master.conf`

   со следующим содержимым:

```
module(load="imfile" PollingInterval="10")
input(type="imfile"
      reopenOnTruncate="on"
      File="/var/log/messages"
      )

if $programname == 'java' and $msg contains 'webserver' then @@<pangeo-log-
collector>:<port>
```

   здесь - это адрес лог-коллектора, а - номер порта, предназначенного для приема событий. Отправка будет выполняться по протоколу TCP.

4. После корректировки настроек rsyslog перезагрузите службу, выполнив следующую команду:

   `# systemctl restart rsyslog`

3.1.2. Настройка журналирования веб-запросов пользователей прокси (skvt-wizor)

Для выбора формата записи журналов перейдите в раздел настроек “Система” > “Расширенные настройки” > “Фильтрация и кэширование трафика”, затем выберите секцию "Фильтрация и анализ трафика пользователей" > "Форматы записи в syslog" (см. рисунок 119).

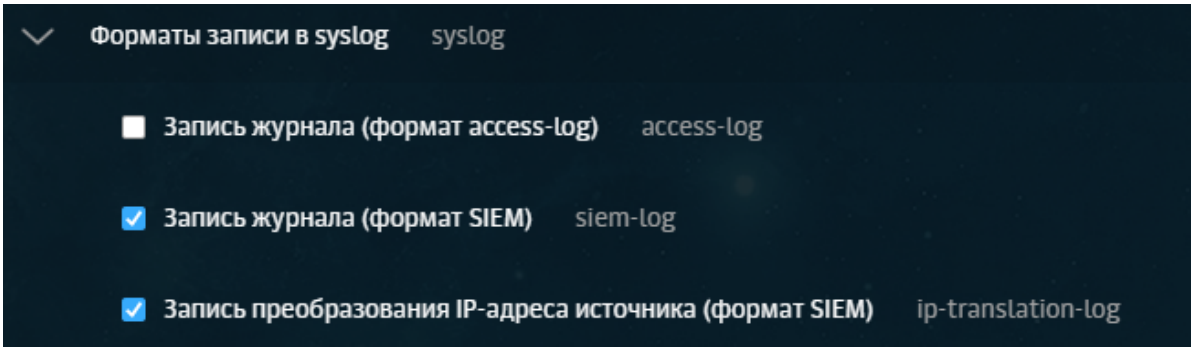

Рисунок 119 - Форматы записи в syslog

Пример событий:

```
Mar 29 15:24:11 swp01-filter java: [acc-domain:TEST.LAB] [acc-groups:] [acc-
ip:192.168.2.70] [acc-name:da] [acc-port:51380] [bytes-in:3147] [bytes-out:781]
[flt-categories:21004] [flt-codes:11,0,0,0,0] [flt-policy:Завершение обработки
политики] [flt-rules:Вскрывать HTTPS по умолчанию,Переход к слою Icap
Request,Переход к слою Запрет доступа к сайтам,Переход к слою Icap
Response,Переход к слою Завершение обработки политики] [flt-status:200] [flt-
time:125] [req-hostname:safebrowsing.googleapis.com] [req-method:GET] [req-
pathname:/v4/threatListUpdates:fetch] [req-protocol:https] [req-
query:$ct=application/x-
protobuf&key=AIzaSyC7jsptDS3am4tPx4r3nxis7IMjBc5Dovo&$httpMethod=POST&$req=ChUKE
25hdmNsaWVudC1hdXRvLWZmb3gaJwgFEAEaGwoNCAUQBhgBIgMwMDEwARC3nRAaAhgFyU6KeiICIAIoA
RonCAEQARobCg0IARAGGAEiAzAwMTABENWDDBoCGAUyxlEzIgIgAigBGicIAxABGhsKDQgDEAYYASIDM
DAxMAEQ8_oLGgIYBVB3OG4iAiACKAEaJwgHEAEaGwoNCAcQBhgBIgMwMDEwARC81AwaAhgFLhmniCICI
AIoARolCAkQARoZCg0ICRAGGAEiAzAwMTABECAaAhgF-13fQCICIAIoAQ==] [req-referer:] [req-
time:2023-03-29T12:24:11.471Z] [res-datatype:application/x-protobuf] [res-
ip:108.177.14.95] [traf-mode:forward] [x-virus-id:] [req-port:443] [flt-reason:]
Mar 28 11:35:59 swp01-filter java: [acc-domain:] [acc-groups:] [acc-
ip:192.168.2.70] [acc-name:] [acc-port:55073] [bytes-in:0] [bytes-out:0] [flt-
categories:] [flt-codes:0] [flt-policy:policy.xml] [flt-rules:] [flt-status:407]
[flt-time:1] [req-hostname:secure.eicar.org] [req-method:CONNECT] [req-pathname:]
[req-protocol:https] [req-query:] [req-referer:] [req-time:2023-03-
28T08:35:59.399Z] [res-datatype:application/skvt-unchecked] [res-ip:] [traf-
mode:forward] [x-virus-id:] [req-port:443] [flt-reason:]
```

> Для интеграции с **Платформой Радар** необходимо активировать опции "Запись журнала (формат SIEM)" и "Запись преобразования IP-адреса источника (формат SIEM)".

После выбора опций сохраните и примените конфигурацию.

По умолчанию запись событий происходит в `/var/log/messages`, однако предпочтительно настроить журналирование в отдельный файл. Для этого выполните следующие действия на узлах с ролью "Фильтр HTTP-трафика" и "Обратный прокси-сервер":

1. Создайте файл `/var/log/skvt.log`:

`# touch /var/log/skvt.log`

2. Ограничьте доступ к файлу:

`# chmod 600 /var/log/skvt.log`

3.1.3. Отключение записи событий в `/var/log/messages` и запись событий в отдельный файла журнала - `/var/log/skvt.log`

Выполните настройку перенаправления событий в файл `/var/log/skvt.log`. Для этого внесите в файл `/etc/rsyslog.conf` соответствующую конфигурацию.
Так как события будут записываться в файл `/var/log/skvt.log`, отключите дублирование в `/var/log/messages` (оператор stop):

```
$template rawSkvt,"%syslogtag% %msg%\n"

local0.*                                                /var/log/skvt.log;
rawSkvt
& stop
*.info;mail.none;authpriv.none;cron.none                /var/log/messages
```

После внесенных изменений файл конфигурации сохраните, затем перезапустите службу rsyslog:

`# systemctl restart rsyslog`

3.1.4. Настройка ротации

Ротацию файла `/var/log/skvt.log` можно настроить с помощью `logrotate`. Для этого создайте файл

`/etc/logrotate.d/skvt`

со следующим содержимым:

```
/var/log/skvt.log {
    weekly
    rotate 4
    missingok
    notifempty
    nomail
    compress
    create 0600 dozor dozor
    minsize 10M
}
```

Проверку условия `logrotate` выполните с помощью команды:

`logrotate -df /etc/logrotate.d/skvt`

Запуск ротации вручную выполняется следующей командой:

`logrotate -f /etc/logrotate.d/skvt`

3.1.5. Отправка событий в Платформу Радар

Для отправки событий в **Платформу Радар** создайте файл конфигурации

`/etc/rsyslog.d/03-send_skvt.conf`

со следующим содержимым:

```
module(load="imfile" PollingInterval="10")
input(type="imfile"
      reopenOnTruncate="on"
      File="/var/log/skvt.log"
      Tag="skvt_wizor_log"
     )

if $syslogtag == 'skvt_wizor_log' then @@<pangeo-log-collector>:<port>
& stop
```

После внесенных изменений сохраните файл конфигурации, затем перезапустите службу rsyslog:

`# systemctl restart rsyslog`

3.1.6. Пример конфигурации PANGEO-LOG-COLLECTOR

```
tcp_input: &wp_input
  id: "wp_input"
  host: "0.0.0.0"
  port: <port>
  sock_buf_size: 0
  format: "json"
  buf_size: 16384
  log_level: "INFO"

tcp_output: &wp_output
  id: "wp_output"
  target_host: "<radar-balancer-fqdn>"
  port: 2592
  sock_buf_size: 0
  log_level: "INFO"

collectors:
  tcp_receiver:
    - <<: *wp_input

senders:
  port: 48003
  tcp:
    - <<: *wp_output

route_1: &route_1
  collector_id:
    - "wp_input"
  sender_id:
    - "wp_output"
```

```
routers:
  - <<: *route_1
```

4. Другое

4.1. ОС Windows. Утилита Sysmon {#sysmon}

Об утилите

Sysmon (System Monitor) - утилита, которая позволяет получить более полные сведения о событиях Windows.

Ссылка на ресурс Microsoft для подробного изучения.

Для запуска утилиты необходимо, чтобы на машине, на которой планируется сбор событий, было расположено два файла: файл-установщик с расширением .bat или .exe и файл конфигурации с расширением .xml. Для удобства работы рекомендуется расположить эти файлы в одной папке.

Актуальную версию утилиты можно скачать с официального ресурса Microsoft

4.1.1. Настройка источника

1. Установите и настройте утилиту Sysmon:
    - Нажмите **Пуск+S** на клавиатуре
    - Введите в строке поиска **cmd** и нажмите **Enter**
    - Перейдите в папку, где лежат файл-установщик и файл конфигурации с помощью команды

      `cd <directory>`

      Пример: C:\Windows\system32> cd c:\Sysmon
    - Установите утилиту Sysmon с помощью команды `sysmon.exe -i <configfile>`

      Пример: C:\Windows>sysmon.exe -i sysmon.xml
2. После успешной установки в **Просмотре событий Windows** (Event Viewer) появится новый журнал (Channel) **Microsoft-Windows-Sysmon/Operational**.

4.1.2. Включение источника в Платформе Радар

Процесс включения источника в **Платформе Радар** не отличается от включения источника в **Платформе Радар** для Microsoft Windows

4.1.3. Настройка коллектора событий

Процесс настройки лог-коллектора отличается от настройки коллектора событий для Microsoft Windows только настройкой журналов для сбора событий.

Для отправки событий журнала Sysmon в **Платформу Радар** необходимо внести изменение в файл конфигурации лог-коллектора. В разделе **eventlog_collector** необходимо указать в строке **channel** имена всех журналов, события которых нужно отправить в **Платформу Радар**, через запятую.

```
Пример: channel: ['Security','Microsoft-Windows-Sysmon/Operational']
```

4.2. Инструкция по настройке VipNet для отправки событий в Платформу Радар

4.2.1. Отправка событий в формате syslog + CEF

Чтобы настроить передачу данных в **Платформу Радар** в формате CEF, выполните следующие действия:

1. Подключитесь к консоли VipNet Coordinator и пройдите авторизацию с полномочиями администратора.

   ```
   user: user
   password: 11111111

   Вход в режим администратора
   enabled (или en)
   password: 11111111
   ```

2. Определите идентификатор МСЭ, который содержится в приглашении командной строки в составе
   имени узла (например, xF1000-270E033A, где 270E033A — идентификатор МСЭ).
3. Данная настройка работает только в демоне iplircfg.
   - Остановите работу демона iplircfg командой: `iplir stop` (или `ip sto`)
   - Откройте файл конфигурации iplir.conf для редактирования командой: `iplir config` (или `ip co`)
   - Задайте параметры экспорта журнала в секции [misc]:

     ```
     cef_enabled=  yes.
     cef_ip = ip-адрес лог-коллектора.
     cef_port = 514 (или любой другой, который будет использоваться в
     источнике).
     ```

     Дополнительный параметр:

     ```
     cef_format  = ips, или xf
     ```

   - Задайте параметры debug в секции `[debug]`

     ```
     [debug]:
     debuglevel = 3
     debuglogfile = syslog:daemon.debug
     ```

     Секция debuglevel может иметь параметры от -1 до 4 (в старых системах VipNet Coordinator версии 3.x до 5-го).
     Чем выше уровень детализации, тем более подробная информация выводится в журнал. Значение параметра -1 выключает ведение журнала (при этом некоторые важные системные события по-прежнему будут выводиться в журнал).

Секция debuglogfile — источник информации, выводимой в журнал, в формате: `syslog:<facility.level>`, где: facility — процесс, формирующий информацию. Возможные значения: auth, authpriv, cron, daemon, ftp, kern, lpr, mail, mark, news, security, syslog, user, uucp, local0, local1, local2, local3, local4, local5, local6, local7. level — уровень важности информации.
Возможные значения: emerg, panic, alert, crit, err, error, warn, warning, notice, info, debug, none.

Значение параметра debuglogfile по умолчанию — syslog:daemon.debug

Значение параметра debuglogfile = syslog:syslog.debug при запуске перезаписывает все другие добавленные параметры debuglogfile

- Сохраните изменения и закройте конфигурационный файл iplir.conf. Для этого нажмите сочетание клавиш Ctrl+O, далее клавишу Enter и сочетание клавиш Ctrl+X.
- Должно появиться сообщение:

```
Verifying new configuration
<I_CFG> Command: iplir config – iplir.conf has been edited successfully
```

- В случае ошибки появится сообщение типа:

```
Verifying new configuration

/tmp/vipnet/user/iplir.conf/, line 151: invalid ‘/debuglogfile/’ value
error: verification of on configuration has been failed/

<I_CFG> Command:iplir config
: incorrect configuration, please try again Roll back the changes are
restore the previous version file [Yes/No]:
```

При вводе Yes (или Y) возвращает предыдущий успешно сохраненный конфиг.

При вводе No (или N) возвращает только что измененный конфиг (с ошибкой).

- Запустите демон iplircfg командой: `iplir  start` (или `ip sta`)
- Проверьте настройки сервиса iplir без его остановки:

`Iplir show config  (или ip sh co)`

- Создайте разрешающее исходящее правило лог-коллектора командой:

`firewall  local  add  src @local  dst [IP-адрес лог-коллектора] udp  dport  514 pass`

например:

`firewall  local  add  src @local  dst 192.168.0.2 udp  dport  514 pass`

- Задайте параметр отправки событий на адрес лог-коллектора командой:

`machine set loghost [IP-адрес лог-коллектора]`

Синтаксис – `machine set loghost {<IP-адрес> | local | null}`

<IP-адрес> — IP-адрес удаленного сетевого узла, на который должен отправляться системный журнал (удаленное протоколирование).

local — системный журнал хранится на самом ViPNet Coordinator HW (локальное протоколирование).

null — выключение протоколирования.

- Перезапустите службу лог-коллектора.
- Включите источник VipNet в "Источники" -> "Управлении Источниками" и нажмите кнопку "Синхронизировать".
- Проверьте результат в **Платформе Радар** / Просмотр событий. Должны появиться события.

4.2.2. Настройка для лог-коллектора на получение и отправку событий от VipNet Coordinator

```
udp_input_vipnet: &udp_input_vipnet
  id: "udp_input_vipnet"
  host: "0.0.0.0"
    port: 514
    sock_buf_size: 0
    format: "JSON"
    log_level: "INFO"

udp_output_vipnet: &udp_output_vipnet
  id: "udp_output_vipnet"
    target_host: "<ip-адрес Платформы Радар>"
  port: 2211
    batch_mode_enable: false
    batch_flush_interval: 5
    batch_flush_limit: 200
    ssl_compression: false
    require_cert: false
    ssl_enable: false
    cert_file: "client-cert.pem"
    key_file: "client-key.pem"
    cert_key_pass: ""
    ca_file: "ca.pem"
    log_level: "INFO"

senders:
  port: 48002
    udp:
    - <<: *udp_output_vipnet

collectors:
    log_level: "INFO"
    udp_receiver:
   - <<: *udp_input_vipnet

route_1_vipnet: &route_1_vipnet
    collector_id:
   - "udp_input_vipnet"
    sender_id:
   - "udp_output_vipnet"

routers:
    - <<: *route_1_vipnet
```

4.3. Подключение новых источников, не поддерживаемых Платформой Радар

1. Необходимо кликнуть на раздел “Источники”, “Управление источниками”,
2. В поле “Добавить новый источник” настроить новый источник:
    - В поля “Название”, “Тип”, “Вендор” необходимо указать соответствующие значения для добавляемой системы.
    - В поле “Порт” необходимо указать один из свободных портов, куда будут отправляться события с нового источника (+- диапазон 6000-8000).
    - В поле “input_type” необходимо указать протокол, по которому будут отправляться события.
    - В поле “template_format” необходимо выбрать один из шаблонов форматов, в которых будут приходить события.
    - В поле “message_type” необходимо указать идентификатор сообщений новой системы.
    - В поле “parsers” обязательно необходимо указать “common”.
    - В поле “normalizer” обязательно необходимо указать “passthrough”
3. После добавления нового источника его необходимо включить, после чего нажать на кнопку “Синхронизировать”.

Если все настроено правильно, то в индексе errors должны начать появляться события с добавленного источника.

4.4. Добавление UFW в качестве источника

1. Проверить статус UFW:

    ```
    $ sudo ufw status
    Status: active
    ```

2. В случае его неактивности включить:

    `$ sudo ufw enable`

3. Включить логирование и выбрать его уровень (можно также править в `/etc/ufw/ufw.conf`):

    ```
    $ sudo ufw logging on
    $ sudo ufw logging low | medium | high | full
    ```

4. Добавить в конфигурационный файл rsyslog’a строку:

    `:msg,contains,"[UFW " @<ip-адрес коллектора>:<порт>`

5. Перезапустить службу rsyslog.

    `$ sudo systemctl restart rsyslog.service`

4.5. Linux Auditd {#auditd}

Подключение Linux Auditd в качестве источника событий **Платформы Радар**. В качестве примера используется виртуальная машина на Debian 12.

1. Для установки auditd в «Командной строке Linux» (далее – Терминал) выполните следующую команду:

`apt-get install auditd audispd-plugins`

2. Далее нужно настроить конфигурационный файл auditd. Для этого необходимо:
    - Открыть файл auditd.conf, выполнив команду:

      `nano /etc/audit/auditd.conf`
    - В открывшемся файле заменить все содержимое на содержимое:

```
local_events = yes
write_logs = yes
log_file = /var/log/audit/audit.log
log_group = adm
log_format = ENRICHED
flush = INCREMENTAL_ASYNC
freq = 50
max_log_file = 8
num_logs = 5
priority_boost = 4
disp_qos = lossy
dispatcher = /sbin/audispd
name_format = NUMERIC
##name = mydomain
max_log_file_action = ROTATE
space_left = 75
space_left_action = SYSLOG
verify_email = yes
action_mail_acct = root
admin_space_left = 50
admin_space_left_action = SUSPEND
disk_full_action = SUSPEND
disk_error_action = SUSPEND
use_libwrap = yes
##tcp_listen_port = 60
tcp_listen_queue = 5
tcp_max_per_addr = 1
##tcp_client_ports = 1024-65535
tcp_client_max_idle = 0
enable_krb5 = no
krb5_principal = auditd
##krb5_key_file = /etc/audit/audit.key
distribute_network = no
```

3. Далее создайте «файл с правилами расширенного аудита» extended.rules и добавьте туда следующие правила:
    - Создание файла с правилами расширенного аудита

      `nano /etc/audit/rules.d/extended.rules`
    - Содержимое файла с правилами расширенного аудита:

```
-i
--reset-lost
-a never,exit -F arch=b64 -S execve -F exe=/usr/sbin/crond
-a never,exit -F arch=b64 -S execve -F exe=/lib/systemd/systemd-logind
-a never,filesystem -F fstype=tracefs
```

```
-a never,filesystem -F fstype=debugfs
-a exclude,never -F msgtype=BPRM_FCAPS

## kernel modules
-a always,exit -F arch=b64 -S finit_module,init_module,delete_module -F
auid!=unset
-a always,exit -F arch=b64 -S finit_module,init_module,delete_module -F
auid!=unset
-a always,exit -F arch=b64 -S socket -F a0=2
-a always,exit -F arch=b32 -S socket -F a0=2
-a always,exit -F arch=b64 -S socket -F a0=0xa
-a always,exit -F arch=b32 -S socket -F a0=0xa
-a always,exit -F arch=b64 -S socket -F a0=0x11
-a always,exit -F arch=b32 -S socket -F a0=0x11
-a always,exit -F arch=b64 -S execve,execveat -F auid=unset -F euid>=0 -
F euid<1000
-a always,exit -F arch=b32 -S execve,execveat -F auid=unset -F euid>=0 -
F euid<1000

## listen
-a always,exit -F arch=b64 -S listen
-a always,exit -F arch=b32 -S listen

## process UID/GID
-a always,exit -F arch=b64 -S setuid,setgid,setreuid,setregid
-a always,exit -F arch=b32 -S setuid,setgid,setreuid,setregid

## process tracing
-a always,exit -F arch=b64 -S ptrace
-a always,exit -F arch=b32 -S ptrace

## capabilities
#-a always,exit -F arch=b64 -S capset
#-a always,exit -F arch=b32 -S capset
```

```
## ACLs and file attributes

-a always,exit -F arch=b64 -S setxattr,fsetxattr,lsetxattr
-a always,exit -F arch=b32 -S setxattr,fsetxattr,lsetxattr

## time
-a exit,always -F arch=b64 -S adjtimex,settimeofday,clock_settime
-a exit,always -F arch=b32 -S adjtimex,settimeofday,clock_settime

## hostname
-a always,exit -F arch=b64 -S sethostname,setdomainname
-a always,exit -F arch=b32 -S sethostname,setdomainname

## pam
-a always,exit -F dir=/etc/pam.d -F perm=wa -F
-a always,exit -F dir=/etc/security -F perm=wa -F

## passwd
-a always,exit -F path=/etc/passwd -F auid!=unset -F auid>=1000 -F perm=r
-a always,exit -F path=/etc/group -F auid!=unset -F auid>=1000 -F perm=r
```

```
-a always,exit -F path=/etc/shadow -F perm=r -F auid!=unset
-a always,exit -F path=/etc/passwd -F perm=wa
-a always,exit -F path=/etc/group -F perm=wa
-a always,exit -F path=/etc/shadow -F perm=wa
-a always,exit -F path=/etc/gshadow -F perm=wa

## COD
-a always,exit -F path=/etc/sssd/sssd.conf -F perm=wa
-a always,exit -F path=/etc/nsswitch.conf -F perm=wa
-a always,exit -F path=/etc/krb5.conf -F perm=wa
-a always,exit -F path=/etc/krb5.conf.d -F perm=wa
-a always,exit -F path=/etc/krb5.keytab -F perm=wa

## pki
-a always,exit -F path=/etc/pki/ca-trust -F perm=wa

## audit
-a always,exit -F path=/etc/libaudit.conf -F perm=wa
-a always,exit -F dir=/etc/audit -F perm=wa

## init
-a always,exit -F path=/etc/fstab -F perm=wa
-a always,exit -F dir=/etc/sysconfig -F perm=wa
## network
-a always,exit -F path=/etc/issue -F perm=wa
-a always,exit -F path=/etc/issue.net -F perm=wa
-a always,exit -F path=/etc/hosts -F perm=wa
-a always,exit -F path=/etc/hostname -F perm=wa
-a always,exit -F path=/etc/resolv.conf -F perm=wa
-a always,exit -F dir=/etc/NetworkManager -F perm=wa

## login defaults
-a always,exit -F path=/etc/login.defs -F perm=wa
-a always,exit -F path=/etc/securetty -F perm=wa

## profiles
-a always,exit -F path=/etc/bashrc -F perm=wa
-a always,exit -F path=/etc/profile -F perm=wa
-a always,exit -F path=/etc/profile.d -F perm=wa
-a always,exit -F path=/etc/skel -F perm=wa

## package management
-a always,exit -F path=/etc/yum.conf -F perm=wa
-a always,exit -F dir=/etc/yum -F perm=wa
-a always,exit -F dir=/etc/yum.repos.d -F perm=wa

## mail
-a always,exit -F path=/etc/postfix -F perm=wa
-a always,exit -F path=/etc/aliases -F perm=wa

## ntp
-a always,exit -F path=/etc/ntp.conf -F perm=wa

## syslog
-a always,exit -F path=/etc/rsyslog.conf -F perm=wa
```

```
-a always,exit -F path=/etc/rsyslog.d -F perm=wa

## kernel
-a always,exit -F path=/etc/sysctl.conf -F perm=wa
-a always,exit -F path=/etc/sysctl.d -F perm=wa
-a always,exit -F path=/etc/modprobe.d -F perm=wa

## logrotate
-a always,exit -F path=/etc/logrotate.conf -F perm=wa
-a always,exit -F path=/etc/logrotate.d -F perm=wa

## mandatory access control
-a always,exit -F path=/etc/selinux/config -F perm=wa

## ssh
-a always,exit -F path=/etc/ssh -F perm=wa

## ld.so
-a always,exit -F path=/etc/ld.so.conf -F perm=wa
-a always,exit -F path=/etc/ld.so.conf.d -F perm=wa

## sudo
-a always,exit -F path=/etc/sudoers.d -F perm=r
-a always,exit -F path=/etc/sudoers -F perm=r
-a always,exit -F path=/etc/sudoers.d -F perm=wa
-a always,exit -F path=/etc/sudoers -F perm=wa
-a always,exit -F path=/etc/sudo.conf -F perm=wa
-a always,exit -F path=/etc/sudo-ldap.conf -F perm=wa

## scheduler
-a always,exit -F path=/etc/cron.allow -F perm=wa
-a always,exit -F path=/etc/cron.deny -F perm=wa
-a always,exit -F path=/etc/cron.d -F perm=wa
-a always,exit -F path=/etc/cron.daily -F perm=wa
-a always,exit -F path=/etc/cron.hourly -F perm=wa
-a always,exit -F path=/etc/cron.monthly -F perm=wa

## boot
-a always,exit -F dir=/boot -F perm=wa

## bin
-a always,exit -F dir=/bin -F perm=wa
-a always,exit -F dir=/usr/bin -F perm=wa
-a always,exit -F dir=/sbin -F perm=wa
-a always,exit -F dir=/usr/sbin -F perm=wa
-a always,exit -F dir=/usr/local/bin -F perm=wa
-a always,exit -F dir=/usr/local/sbin -F perm=wa
-a always,exit -F dir=/usr/libexec -F perm=wa

## lib
-a always,exit -F dir=/lib64 -F perm=wa
-a always,exit -F dir=/usr/lib64 -F perm=wa
-a always,exit -F dir=/lib -F perm=wa
-a always,exit -F dir=/usr/lib -F perm=wa
```

```
## log
-a always,exit -F dir=/var/log -F perm=r -F euid>=1000
-a always,exit -F dir=/var/log -F perm=wa -F auid!=unset

## spool
-a always,exit -F path=/var/spool/cron -F perm=wa
-a always,exit -F path=/var/spool/anacron -F perm=wa

## www
-a always,exit -F path=/var/www -F perm=wa

## home
-a always,exit -F dir=/home -F perm=r -F auid!=unset
-a always,exit -F dir=/home -F perm=wa -F auid!=unset

## root
-a always,exit -F dir=/root -F perm=r -F auid!=unset
-a always,exit -F dir=/root -F perm=wa -F auid!=unset

## Finalize rules
-e 1
```

4. Далее настройте плагин syslog для записи логов auditd в syslog.
    - Откройте файл syslog.conf

      `nano /etc/audisp/plugins.d/syslog.conf`
    - Замените содержимое файла содержимым:

      ```
      # This file controls the configuration of the syslog plugin.
      # It simply takes events and writes them to syslog. The
      # arguments provided can be the default priority that you
      # want the events written with. And optionally, you can give
      # a second argument indicating the facility that you want events
      # logged to. Valid options are LOG_LOCAL0 through 7, LOG_AUTH,
      # LOG_AUTHPRIV, LOG_DAEMON, LOG_SYSLOG, and LOG_USER.

      active = yes
      direction = out
      path = builtin_syslog
      type = builtin
      args = LOG_LOCAL6
      format = string
      ```

5. Перезапустите сервис auditd командой:

   `service auditd restart`

6. Далее добавьте настройку в конфигурационный файл rsyslog.conf для экспорта событий с системы на лог-коллектор:
    - Откройте файл rsyslog.conf

      `nano /etc/rsyslog.conf`
    - Вставьте следующую строку в конце файла, указывая ip-адрес и порт

`local6.* @<ip-адрес_лог-коллектора>:<выделенный_порт>`

Для отправки по UDP-соединению необходимо указывать 1 символ «commercial at»/ «собачки» @ перед Ip-адресом, для TCP-соединения необходимо указывать 2 символа @@/ Пример:

```
Для - UDP   #local6.*        @192.168.100.101:2674
Для - TCP   #local6.*        @@192.168.100.101:2674
```

7. После выполнения всех вышеуказанных пунктов перезапустите сервис rsyslog, командой:

   `service rsyslog restart`

8. Проверьте наличие поступающих событий в Web-интерфейсе **Платформы Радар** в разделе «Просмотр событий».

4.6. Confident Dallaslock {#dallas}

Настройка получения событий от DallasLock в **Платформу Радар**.

4.6.1. Включение аудита DallasLock:

1. В оболочке администратора DallasLock перейдите на вкладку Параметры безопасности → Аудит.
2. Найдите пункт Выгрузка журналов, кликните правой кнопкой мыши выбрав пункт Свойства (см. рисунок 120).

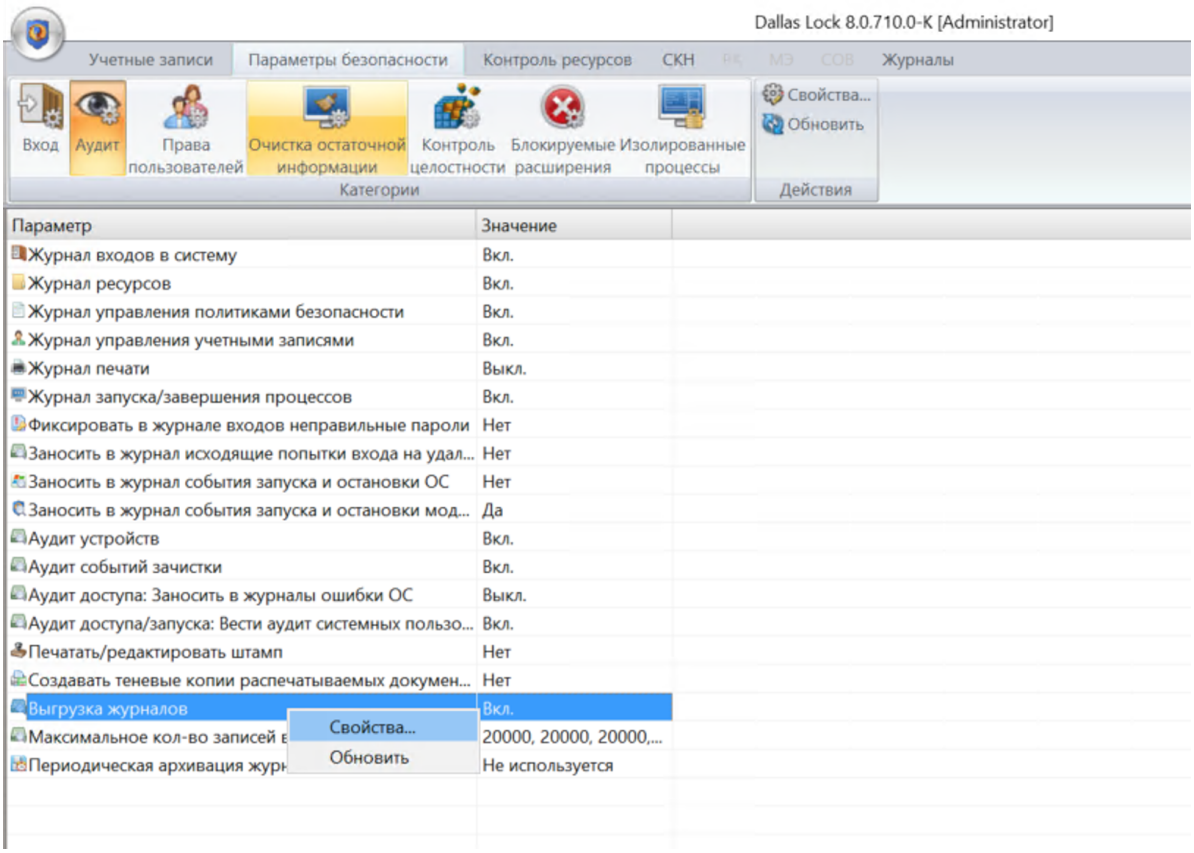

Рисунок 120 - Аудит.

3. В свойствах отметьте галочкой пункт **Экспорт журналов в SIEM систему**. Укажите адрес сервера с лог-коллектором и выберите порт для подключения. Задайте формат и кодировку

выгрузки (см. рисунок 121).

Перечень журналов для логирования выберите в соответствии с требованиями информационной безопасности.

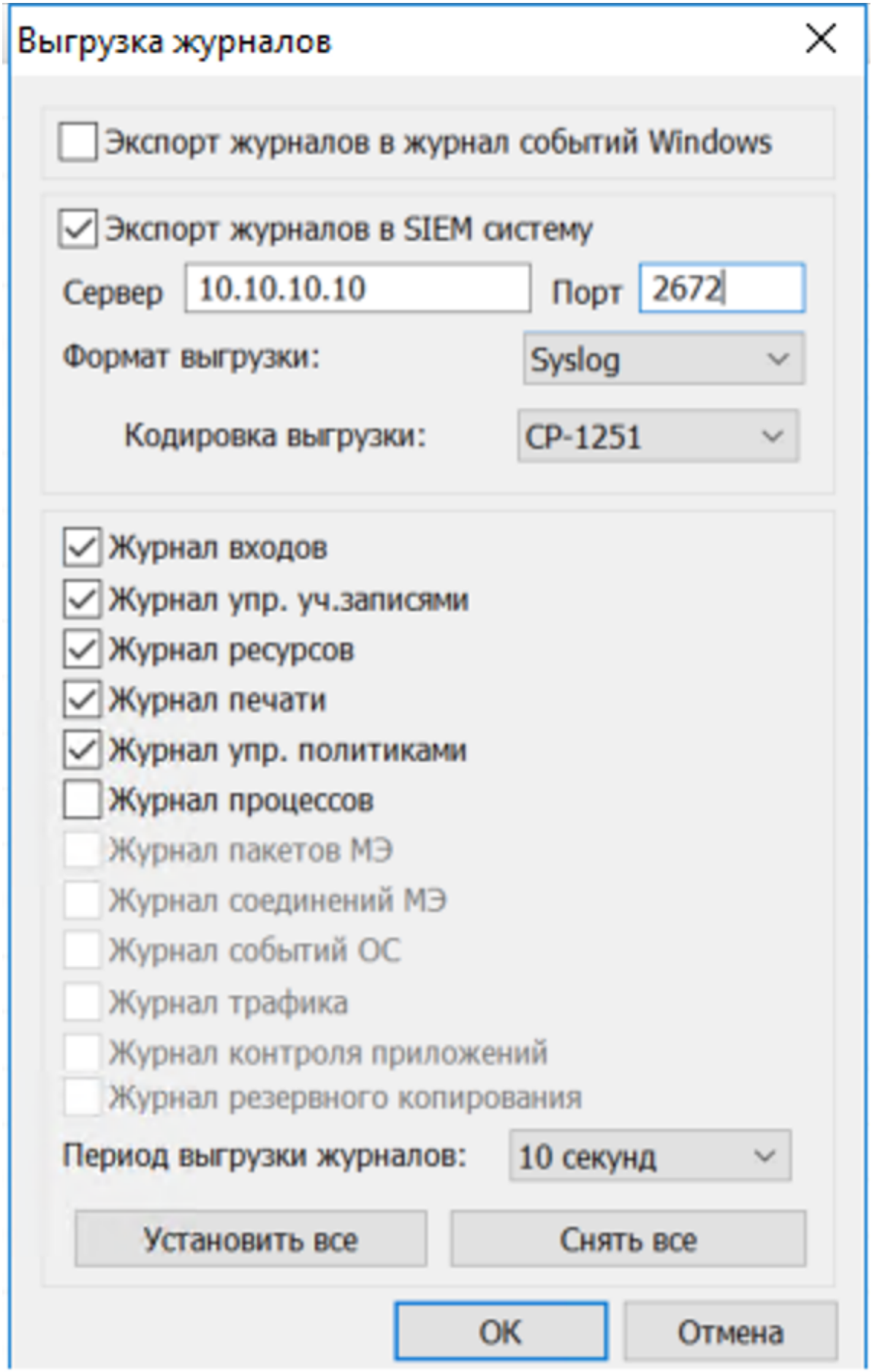

Рисунок 121 - Выбор журналов.

4.6.2. Добавление новой конфигурации в коллектор:

Приведенные настройки с описанием для добавления в config.yaml ниже:

```
tcp_input_2: &tcp_input_2
  id: "tcp_input_2"
  host: "172.30.254.69"
  port: 2672
  sock_buf_size: 0
  format: "json"
  encoding:
    change_to_utf8: true
    original_encoding: "cp1251"
```

> В качестве порта для подключения укажите выбранный ранее в свойствах выгрузки журналов.

5. Описание

Раздел «Правила обработки событий» содержит описание этапов обработки событий и рекомендации по настройке правил для их обработки. Описаны *поля нормализации*. Описаны *специальные функции для работы с полями нормализации* для дополнительной обработки событий прямо в веб-интерфейсе Платформы.

Платформа Радар позволяет как создавать новые пользовательские правила разбора и нормализации событий, так и редактировать существующие

В рамках услуги по технической поддержке могут быть разработаны правила разбора событий для источников, не входящих в стандартный пакет поставки. Срок разработки от 1 рабочего дня.

Платформа гарантирует обработку и анализ событий в режиме, близком к реальному времени.

Платформа обеспечивает обработку мультиязычных событий.

5.1. Этапы обработки события

Событие, поступившее в Платформу, проходит следующие этапы обработки:

- **Сбор** – получение события от целевой системы/лог-коллектора, сохранение на диск в raw-формате или добавление в очередь.
- **Фильтрация** – выделение событий, удовлетворяющих условиям правил фильтрации.
- **Определение типа** – определение типа системы от которой поступило событие для выбора правильных правил разбора и нормализации. Определение типа может быть статическим (задается в конфигурационном файле) и динамическим (с помощью специального правила).
- **Разбор** – разбиение необработанного текста события на фрагменты полезных данных.
- **Нормализация** – приведение всех данных, содержащихся в событии, к единой форме представления. На данном этапе также происходит категоризация событий.
- **Обогащение** – добавление в нормализованное событие дополнительной информации, полезной для выявления и расследования инцидентов.
- **Корреляция** – сопоставление данных из одного или нескольких событий с дополнительной информацией с целью выявления инцидента информационной безопасности.

6. Описание этапов разбора

6.1. Проверка этапов парсинга

В основном, все источники посылают события в формате RAW-JSON. При разборе событий в этом формате необходимо в качестве первого этапа использовать парсер JSON, а потом один из доступных в системе, в зависимости от типа данных в исходном событии.
Платформа Радар позволяет без ручной настройки разбирать следующие структурированные типы данных:

- XML
- Syslog
- CEF
- JSON
- CSV

При использовании нескольких этапов разбора событий в каждом дополнительно создаваемом парсере необходимо указывать в поле «Цель» ту переменную, значение которой необходимо разобрать.

6.1.1. JSON

Сырое событие:


```
{"rs_collector_hostname":"v-stand-
09","rs_relay_fqdn":"172.30.254.106","rs_relay_ip":"172.30.254.106","rs_collecto
r_ts":"2021-11-18T11:52:54.446148+03:00","__rs_module":"2613-Kaspersky-
SecurityCenter-db-host-
activity","fqdn":"WINSRV02.demo.local","ip_address":"192.168.100.100","last_info
_update":"2021-11-
18T08:18:49.000000+00:00","last_net_agent_connected":null,"last_update":null,"la
st_visible":"2021-09-
23T10:06:28.000000+00:00","nId":43,"nLastRtpState":0,"nStatus":0,"rs_agent_fqdn"
:"log-collector","rs_agent_ip":"172.30.254.106","rs_agent_ts":"2021-11-
18T11:52:54.4389503+03:00","wstrDisplayName":"WINSRV02","wstrDnsDomain":"demo.lo
cal"}
```


Результат обработки представлен на рисунке 122.

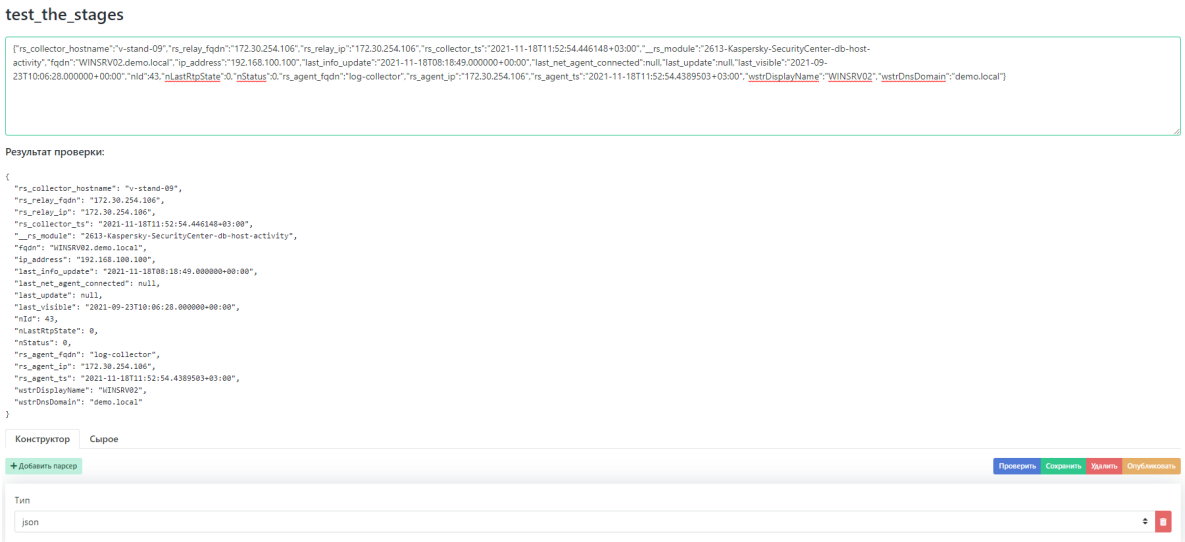

Рисунок 122 - Результат обработки этапа JSON

Результат обработки в текстовом виде:

```
{
  "rs_collector_hostname": "v-stand-09",
  "rs_relay_fqdn": "172.30.254.106",
  "rs_relay_ip": "172.30.254.106",
  "rs_collector_ts": "2021-11-18T11:52:54.446148+03:00",
  "__rs_module": "2613-Kaspersky-SecurityCenter-db-host-activity",
  "fqdn": "WINSRV02.demo.local",
  "ip_address": "192.168.100.100",
  "last_info_update": "2021-11-18T08:18:49.000000+00:00",
  "last_net_agent_connected": null,
  "last_update": null,
  "last_visible": "2021-09-23T10:06:28.000000+00:00",
  "nId": 43,
  "nLastRtpState": 0,
  "nStatus": 0,
  "rs_agent_fqdn": "log-collector",
  "rs_agent_ip": "172.30.254.106",
  "rs_agent_ts": "2021-11-18T11:52:54.4389503+03:00",
  "wstrDisplayName": "WINSRV02",
  "wstrDnsDomain": "demo.local"
}
```


6.1.2. CEF_NONSTRICT

Сырое событие:

```
CEF:1|IP Flow|IP Flow|9|flow|NetFlow Event|Unknown| eventId=13252253246
start=1623223861208 end=1623223861272 proto=TCP in=1098
categoryBehavior=/Communicate categoryDeviceGroup=/Network Equipment
catdt=Network Monitoring categoryOutcome=/Attempt categoryObject=/Host
art=1623224462176 rt=1623223873000 deviceDirection=0 src=172.0.218.2
sourceZoneURI=/All Zones/ArcSight System/Private Address Space Zones/RFC1918:
10.0.0.0-10.255.255.255 spt=8787 dst=172.0.18.108 destinationZoneURI=/All
Zones/ArcSight System/Private Address Space Zones/RFC1918: 10.0.0.0-
10.255.255.255 dpt=53445 fileType=NAT Source IPv4 Address: fileHash=NAT Source
Port: oldFileType=NAT Destination IPv4 Address: oldFileHash=NAT Destination Port:
cs1=172.0.245.1 cs4=13 cs5=26 cn1=9 cn3=0 cs1Label=nexthop cs2Label=src_as
cs3Label=dst_as cs4Label=src_mask cs5Label=dst_mask cs6Label=tcp_flags descr
cn1Label=in_pkts cn2Label=out_pkts cn3Label=tcp_flags ahost=arcsight-test
agt=172.0.6.96 agentZoneURI=/All Zones/ArcSight System/Private Address Space
Zones/RFC1918: 10.0.0.0-10.255.255.255 amac=34-B3-54-BC-66-C6 av=7.14.0.8241.0
atz=Europe/Moscow at=cisco_netflow dvchost=arcsight-test dvc=172.0.255.245
deviceZoneURI=/All Zones/ArcSight System/Private Address Space Zones/RFC1918:
10.0.0.0-10.255.255.255 dtz=Europe/Moscow geid=0 _cefVer=1.0
ad.flow__sampler__id=0 ad.vendor__51=0 ad.DevicePort=61673
ad.interface__output__snmp=312 ad.src__tos=0 ad.pkthdr__uptime=444691076
ad.pkthdr__seq=787165105 ad.pkthdr__source__id=517 ad.pkthdr__count=32
ad.interface__input__snmp=153 aid=3hughqHkBABCBSulnxz6OxA\\=\\=
```

Результат обработки в текстовом виде:


```
{
  "rs_collector_hostname": "radar-balancer-01",
  "rs_relay_fqdn": "arcsight-test",
```

```
  "rs_relay_ip": "172.0.0.96",
  "rs_collector_ts": "2021-06-09T10:41:02.253872+03:00",
  "__rs_module": "3500-Arcsight-Smartconnector-Netflow-cef",
  "cef_version": 1,
  "vendor": "IP Flow",
  "product": "IP Flow",
  "version": "9",
  "signature": "flow",
  "name": "NetFlow Event",
  "severity": "Unknown",
  "eventId": "13252253246",
  "start": "1623223861208",
  "end": "1623223861272",
  "proto": "TCP",
  "in": "1098",
  "categoryBehavior": "/Communicate",
  "categoryDeviceGroup": "/Network Equipment",
  "catdt": "Network Monitoring",
  "categoryOutcome": "/Attempt",
  "categoryObject": "/Host",
  "art": "1623224462176",
  "rt": "1623223873000",
  "deviceDirection": "0",
  "src": "172.0.218.2",
  "sourceZoneURI": "/All Zones/ArcSight System/Private Address Space
Zones/RFC1918: 10.0.0.0-10.255.255.255",
  "spt": "8787",
  "dst": "172.0.18.108",
  "destinationZoneURI": "/All Zones/ArcSight System/Private Address Space
Zones/RFC1918: 10.0.0.0-10.255.255.255",
  "dpt": "53445",
  "fileType": "NAT Source IPv4 Address:",
  "fileHash": "NAT Source Port:",
  "oldFileType": "NAT Destination IPv4 Address:",
  "oldFileHash": "NAT Destination Port:",
  "ahost": "arcsight-test",
  "agt": "172.0.6.96",
  "agentZoneURI": "/All Zones/ArcSight System/Private Address Space
Zones/RFC1918: 10.0.0.0-10.255.255.255",
  "amac": "34-B3-54-BC-66-C6",
  "av": "7.14.0.8241.0",
  "atz": "Europe/Moscow",
  "at": "cisco_netflow",
  "dvchost": "arcsight-test",
  "dvc": "172.0.255.245",
  "deviceZoneURI": "/All Zones/ArcSight System/Private Address Space
Zones/RFC1918: 10.0.0.0-10.255.255.255",
  "dtz": "Europe/Moscow",
  "geid": "0",
  "_cefVer": "1.0",
  "ad.flow__sampler__id": "0",
  "ad.vendor__51": "0",
  "ad.DevicePort": "61673",
  "ad.interface__output__snmp": "312",
  "ad.src__tos": "0",
```

```
  "ad.pkthdr__uptime": "444691076",
  "ad.pkthdr__seq": "787165105",
  "ad.pkthdr__source__id": "517",
  "ad.pkthdr__count": "32",
  "ad.interface__input__snmp": "153",
  "aid": "3hughqHkBABCBSulnxz6OxA==",
  "nexthop": "172.0.245.1",
  "src_mask": "13",
  "dst_mask": "26",
  "in_pkts": "9",
  "tcp_flags": "0"
}
```


6.1.3. CEF

Сырое событие:

```
CEF:0|InfoTeCS|IDS|2.4.3-371989|1:2023753:2|ET SCAN MS Terminal Server Traffic on
Non-standard Port|2|cat=1 cn1=348158796 cn1Label=EventID cnt=1 cs1=attempted-
recon cs1Label=IDSClass cs2=emerging-scan cs2Label=IDSGroup cs3= cs3Label=CVEID
cs4= cs4Label=ExternalRef cs5= cs5Label=IDSTags deviceExternalId=341000778
deviceFacility=Signature dmac=00:1c:58:8b:46:00 dpt=54321 dst=62.33.180.235
proto=TCP rt=May 31 2021 19:36:57.181 YEKT smac=84:78:ac:34:5e:a2 spt=2324
src=95.142.121.19
```

Результат обработки представлен на рисунке 123:

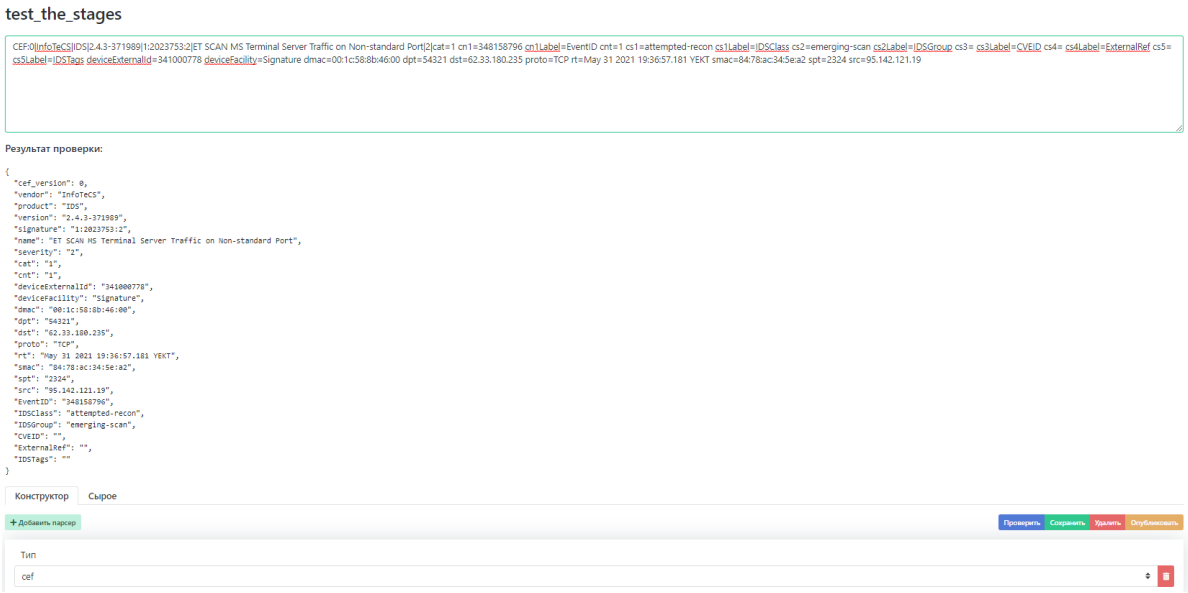

Рисунок 123 - Результат обработки этапа CEF

Результат обработки в текстовом виде:


```
{
  "cef_version": 0,
  "vendor": "InfoTeCS",
  "product": "IDS",
  "version": "2.4.3-371989",
  "signature": "1:2023753:2",
  "name": "ET SCAN MS Terminal Server Traffic on Non-standard Port",
```

```
  "severity": "2",
  "cat": "1",
  "cnt": "1",
  "deviceExternalId": "341000778",
  "deviceFacility": "Signature",
  "dmac": "00:1c:58:8b:46:00",
  "dpt": "54321",
  "dst": "62.33.180.235",
  "proto": "TCP",
  "rt": "May 31 2021 19:36:57.181 YEKT",
  "smac": "84:78:ac:34:5e:a2",
  "spt": "2324",
  "src": "95.142.121.19",
  "EventID": "348158796",
  "IDSClass": "attempted-recon",
  "IDSGroup": "emerging-scan",
  "CVEID": "",
  "ExternalRef": "",
  "IDSTags": ""
}
```


6.1.4. XML

Сырое событие:


```
<AuditRecord><Audit_Type>1</Audit_Type><Session_Id>250388</Session_Id>
<StatementId>1</StatementId><EntryId>1</EntryId><Extended_Timestamp>2020-08-
25T19:57:32.604660Z</Extended_Timestamp><DB_User>RADAR</DB_User>
<OS_User>oracle</OS_User><Userhost>805cd2dc9016</Userhost>
<OS_Process>1313</OS_Process><Terminal>pts/0</Terminal>
<Instance_Number>0</Instance_Number><Action>100</Action>
<TransactionId>12001300EE070000</TransactionId><Returncode>0</Returncode>
<Comment_Text>Authenticated by: DATABASE</Comment_Text><Priv_Used>5</Priv_Used>
<DBID>2722566360</DBID><Current_User>RADAR</Current_User>\\n</AuditRecord>
```


Результат обработки представлен на рисунке 124:

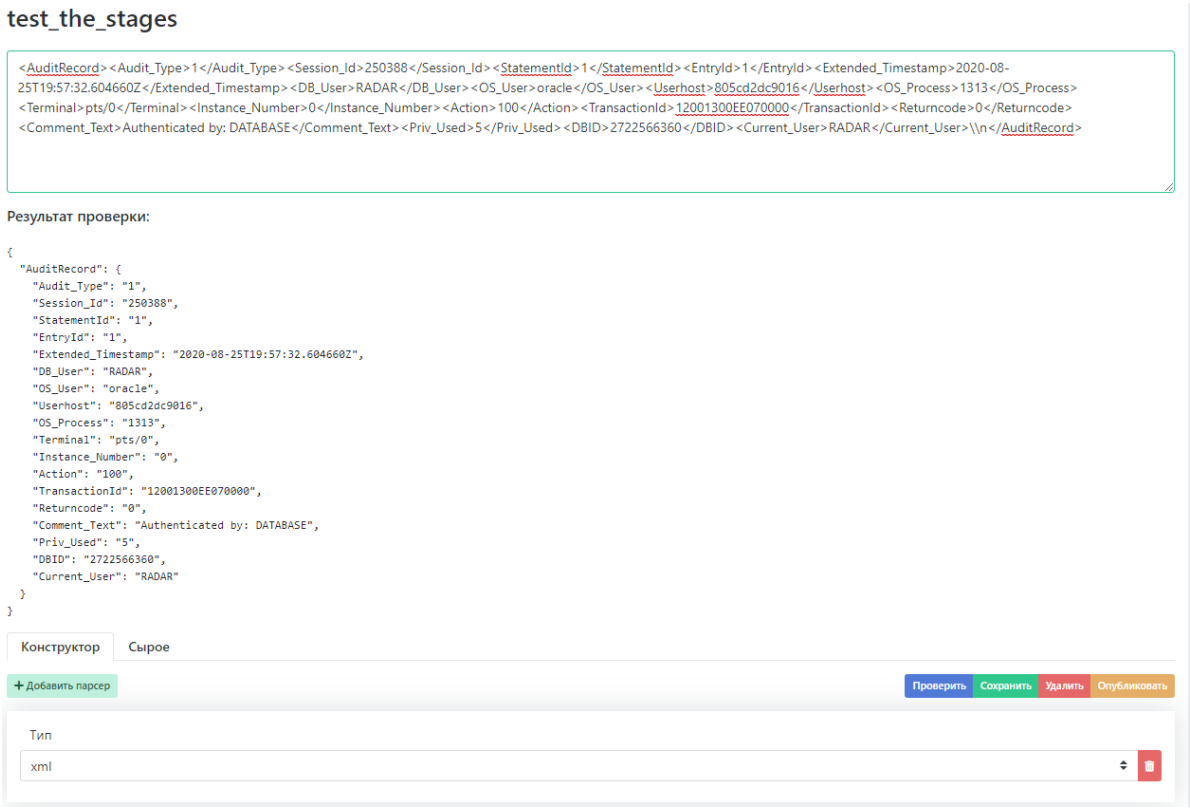

Рисунок 124 - Результат обработки этапа XML

Результат обработки в текстовом виде:


```
{
  "AuditRecord": {
    "Audit_Type": "1",
    "Session_Id": "250388",
    "StatementId": "1",
    "EntryId": "1",
    "Extended_Timestamp": "2020-08-25T19:57:32.604660Z",
    "DB_User": "RADAR",
    "OS_User": "oracle",
    "Userhost": "805cd2dc9016",
    "OS_Process": "1313",
    "Terminal": "pts/0",
    "Instance_Number": "0",
    "Action": "100",
    "TransactionId": "12001300EE070000",
    "Returncode": "0",
    "Comment_Text": "Authenticated by: DATABASE",
    "Priv_Used": "5",
    "DBID": "2722566360",
    "Current_User": "RADAR"
  }
}
```

6.1.5. CSV

Сырое событие:

```
"-1","domain618\\user286","10.10.200.10","POST","2619","500","host333.domain66.net","/path5","DENIED","","1557410124","2019-05-09 13:55:24","https","Streaming Media","","","Minimal Risk","Block URLs Whose Category Is in Category Blocklist for Default Groups","403","10.10.23.19","","Blocked by URL filtering","Other","","Google Update/1.3.33.23;winhttp\
```

Настройка этапа разбора представлена на рисунке 125 и рисунке 126:

Рисунок 125 - Настройка этапа разбора CSV

Рисунок 126 - Настройка этапа разбора CSV

Результат разбора представлен на рисунке 127:

Результат проверки:


```
{
  "user_id": "-1",
  "username": "domain618\\user286",
  "source_ip": "10.10.200.10",
  "http_action": "POST",
  "server_to_client_bytes": "2619",
  "client_to_server_bytes": "500",
  "requested_host": "host333.domain66.net",
  "requested_path": "/path5",
  "result": "DENIED",
  "virus": "",
  "request_timestamp_epoch": "1557410124",
  "request_timestamp": "2019-05-09 13:55:24",
  "uri_scheme": "https",
  "category": "Streaming Media",
  "media_type": "",
  "application_type": "",
  "reputation": "Minimal Risk",
  "last_rule": "Block URLs Whose Category Is in Category Blocklist for Default Groups",
  "http_status_code": "403",
  "client_ip": "10.10.23.19",
  "location": "",
  "block_reason": "Blocked by URL filtering",
  "user_agent_product": "Other",
  "user_agent_version": "",
  "user_agent_comment": "Google Update/1.3.33.23;winhttp\n"
}
```


Рисунок 127 - Результат обработки этапа CSV

Результат разбора в текстовом виде:


```
{
  "user_id": "-1",
  "username": "domain618\\user286",
  "source_ip": "10.10.200.10",
  "http_action": "POST",
  "server_to_client_bytes": "2619",
  "client_to_server_bytes": "500",
  "requested_host": "host333.domain66.net",
  "requested_path": "/path5",
  "result": "DENIED",
  "virus": "",
  "request_timestamp_epoch": "1557410124",
  "request_timestamp": "2019-05-09 13:55:24",
  "uri_scheme": "https",
  "category": "Streaming Media",
  "media_type": "",
  "application_type": "",
  "reputation": "Minimal Risk",
  "last_rule": "Block URLs Whose Category Is in Category Blocklist for Default Groups",
  "http_status_code": "403",
  "client_ip": "10.10.23.19",
  "location": "",
  "block_reason": "Blocked by URL filtering",
  "user_agent_product": "Other",
  "user_agent_version": "",
  "user_agent_comment": "Google Update/1.3.33.23;winhttp\n"
}
```

6.1.6. GROK

Для работы парсера необходимо завалидировать (проверить) паттерн и в случае успеха добавить его для использования, нажав на зеленый плюсик рядом с синей кнопкой «Валидировать».

Сырое событие:

```
<86>v-demo-checkpoint sshd[13236]: Accepted password for admin from 192.168.200.2
port 1091 ssh2
```

GROK-паттерн:

```
<?>%{DATA:application_name}\s+%{WORD:service}.*?\s+%{DATA:attempt}\s+for\s+%
{USERNAME:username}\s+from\s+%{IPORHOST:from_host}\s+port\s+%
{BASE10NUM:source_port}\s+%{DATA:transport}$
```

Результат обработки представлен на рисунке 128:

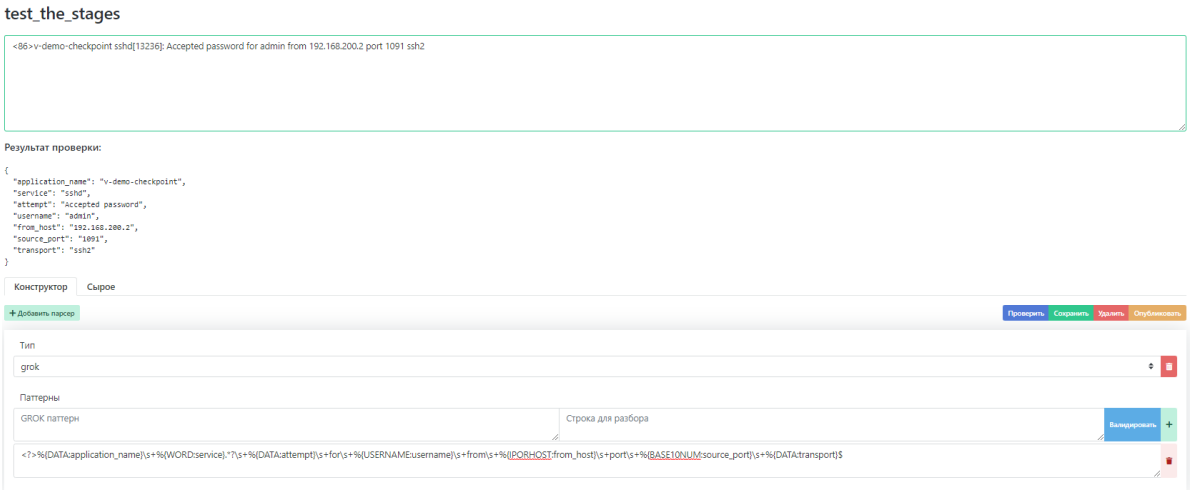

Рисунок 128 - Результат обработки этапа GROK

Результат обработки в текстовом виде:

```
{
  "application_name": "v-demo-checkpoint",
  "service": "sshd",
  "attempt": "Accepted password",
  "username": "admin",
  "from_host": "192.168.200.2",
  "source_port": "1091",
  "transport": "ssh2"
}
```

7. Разработка правил разбора и нормализации событий

7.1. Создание правил разбора {#createparser}

Управление правилами разбора осуществляется в разделе «Источники» → «Управление источниками». Далее выбрать вкладку «Правила разбора», после чего откроется страница управления правилами разбора (см. рисунок 129).

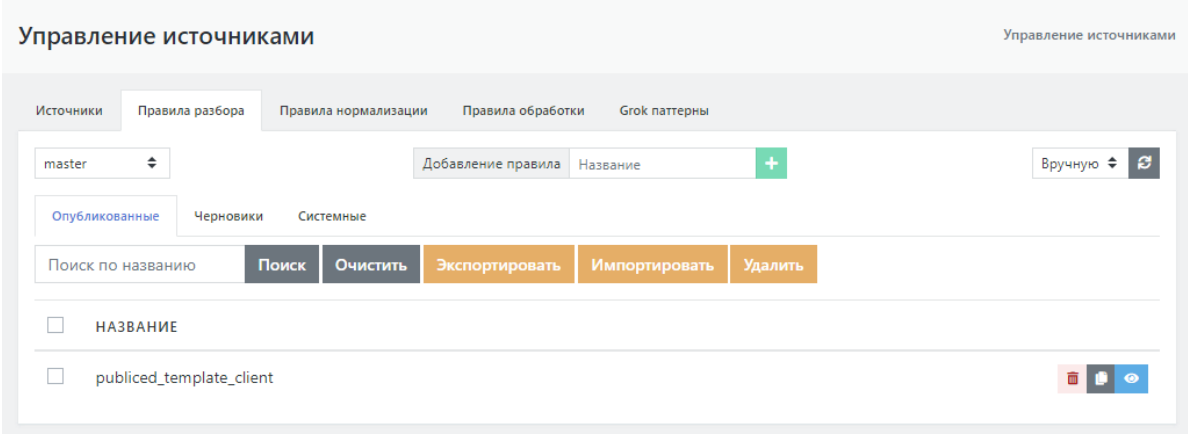

Рисунок 129 - Страница управления правилами разбора

В разделе присутствует три вкладки:

- **Опубликованные** - правила, созданные пользователем и опубликованные на **Платформе Радар**.
- **Черновики** - правила, созданные пользователем, но не опубликованные на **Платформе Радар**. Такие правила будут применяться **Платформой Радар** только после их публикации.
- **Системные** - правила, которые поставляются с **Платформой Радар** и недоступные для редактирования.

Для опубликованных и неопубликованных правил разбора доступны общие функции:

- *Экспортировать* - позволяет экспортировать выбранные правила разбора в файл архива формата ZIP.
- *Импортировать* - позволяет импортировать правила разбора из файла архива формата ZIP.
- *Удалить* - удаляет выбранные правила разбора.

На каждой вкладке с правилами разбора доступен поиск правила по его наименованию.

1. Для создания нового правила разбора необходимо указать название в поле «Добавление правила» и нажать на «+», после чего откроется форма создания правила разбора (см. рисунок 130).

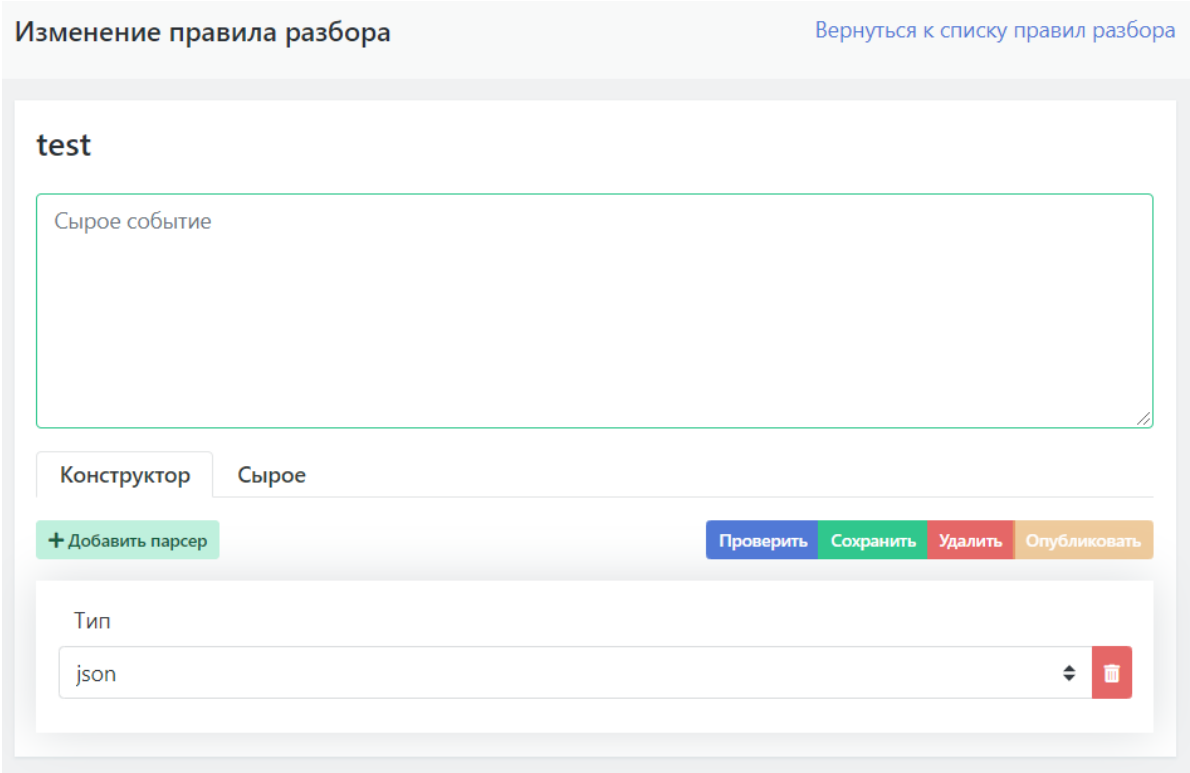

Рисунок 130 - Окно создания правила разбора

2. Рассмотрим процесс создания парсера на примере событий от продукта Micro Focus ArcSight SmartConnector.

   Тип продукта: сетевое оборудование

   Сырое событие:

{"rs_collector_hostname":"radar-balancer-01","rs_relay_fqdn":"arcsight-test","rs_relay_ip":"172.0.0.96","rs_collector_ts":"2021-06-09T10:41:02.253872+03:00","__rs_module":"3500-Arcsight-Smartconnector-Netflow-cef","message":"CEF:1|IP Flow|IP Flow|9|flow|NetFlow Event|Unknown|eventId=13252253246 start=1623223861208 end=1623223861272 proto=TCP in=1098 categoryBehavior=\/Communicate categoryDeviceGroup=\/Network Equipment catdt=Network Monitoring categoryOutcome=\/Attempt categoryObject=\/Host art=1623224462176 rt=1623223873000 deviceDirection=0 src=172.0.218.2 sourceZoneURI=\/All Zones\/ArcSight System\/Private Address Space Zones\/RFC1918: 10.0.0.0-10.255.255.255 spt=8787 dst=172.0.18.108 destinationZoneURI=\/All Zones\/ArcSight System\/Private Address Space Zones\/RFC1918: 10.0.0.0-10.255.255.255 dpt=53445 fileType=NAT Source IPv4 Address: fileHash=NAT Source Port: oldFileType=NAT Destination IPv4 Address: oldFileHash=NAT Destination Port: cs1=172.0.245.1 cs4=13 cs5=26 cn1=9 cn3=0 cs1Label=nexthop cs2Label=src_as cs3Label=dst_as cs4Label=src_mask cs5Label=dst_mask cs6Label=tcp_flags descr cn1Label=in_pkts cn2Label=out_pkts cn3Label=tcp_flags ahost=arcsight-test agt=172.0.6.96 agentZoneURI=\/All Zones\/ArcSight System\/Private Address Space Zones\/RFC1918: 10.0.0.0-10.255.255.255 amac=34-B3-54-BC-66-C6 av=7.14.0.8241.0 atz=Europe\/Moscow at=cisco_netflow dvchost=arcsight-test dvc=172.0.255.245 deviceZoneURI=\/All Zones\/ArcSight System\/Private Address Space Zones\/RFC1918: 10.0.0.0-10.255.255.255 dtz=Europe\/Moscow geid=0 _cefVer=1.0 ad.flow__sampler__id=0 ad.vendor__51=0 ad.DevicePort=61673 ad.interface__output__snmp=312 ad.src__tos=0 ad.pkthdr__uptime=444691076 ad.pkthdr__seq=787165105 ad.pkthdr__source__id=517 ad.pkthdr__count=32 ad.interface__input__snmp=153 aid=3hughqHkBABCBSulnxz6OxA\\=\\="}

> Обратите внимание, что сырое событие представлено в формате JSON.
> Платформа Радар поддерживает работу с событиями, содержащими кириллицу.

3.. Сырое событие необходимо вставить в соответствующее поле.

4. Во вкладке «Тип» нужно указать «json».
5. После нажатия кнопки «Проверить» получаем результат разобранного JSON события (см. рисунок 131).

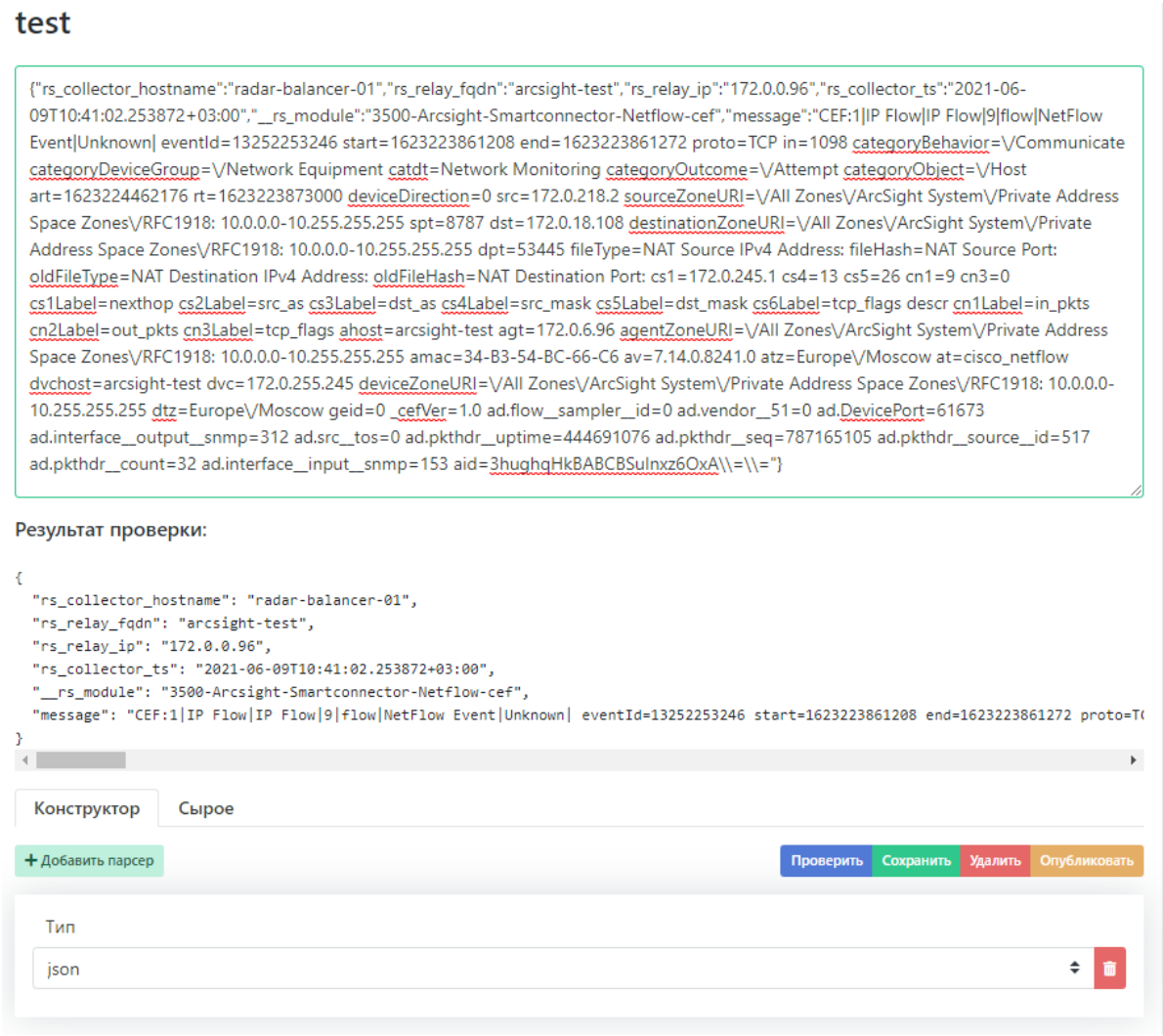

Рисунок 131 - Результат разобранного события

6. Как видно из результата, одного этапа разбора недостаточно, потому что основная информация данного события находится в поле «message».
7. В качестве второго этапа разбора необходимо использовать этап CEF. Для этого следует нажать на кнопку «Добавить парсер» и выбрать «cef_nonstrict» (этот этап используется для разбора формата CEF версии 1).
8. Далее в поле «Цель» второго этапа разбора нужно указать название поля, которое необходимо разобрать, в случае рассматриваемого примера - это поле «message» (см. рисунок 132).

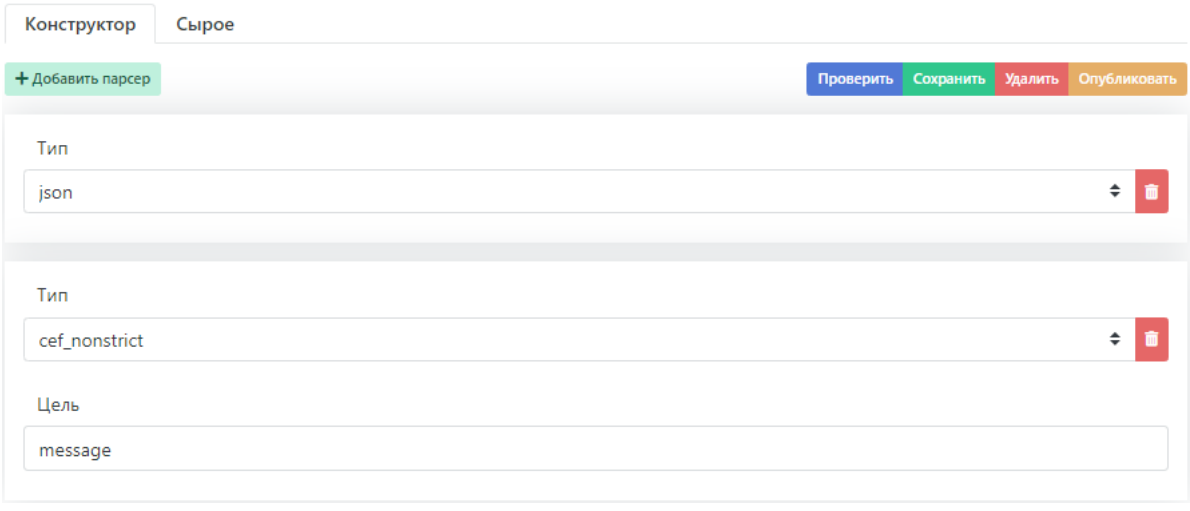

Рисунок 132 - Добавление второго этапа разбора

9. Проверяем работоспособность этапов правила разбора нажатием на кнопку «Проверить».
10. Результатом проверки правила должен быть вывод полностью разобранного события:


```
{
"rs_collector_hostname": "radar-balancer-01",
"rs_relay_fqdn": "arcsight-test",
"rs_relay_ip": "172.0.0.96",
"rs_collector_ts": "2021-06-09T10:41:02.253872+03:00",
"__rs_module": "3500-Arcsight-Smartconnector-Netflow-cef",
  "cef_version": 1,
  "vendor": "IP Flow",
  "product": "IP Flow",
  "version": "9",
  "signature": "flow",
  "name": "NetFlow Event",
  "severity": "Unknown",
  "eventId": "13252253246",
  "start": "1623223861208",
  "end": "1623223861272",
  "proto": "TCP",
  "in": "1098",
  "categoryBehavior": "/Communicate",
  "categoryDeviceGroup": "/Network Equipment",
  "catdt": "Network Monitoring",
  "categoryOutcome": "/Attempt",
  "categoryObject": "/Host",
  "art": "1623224462176",
  "rt": "1623223873000",
  "deviceDirection": "0",
  "src": "172.0.218.2",
  "sourceZoneURI": "/All Zones/ArcSight System/Private Address Space
Zones/RFC1918: 10.0.0.0-10.255.255.255",
  "spt": "8787",
  "dst": "172.0.18.108",
  "destinationZoneURI": "/All Zones/ArcSight System/Private Address Space
Zones/RFC1918: 10.0.0.0-10.255.255.255",
  "dpt": "53445",
  "fileType": "NAT Source IPv4 Address:",
  "fileHash": "NAT Source Port:",
  "oldFileType": "NAT Destination IPv4 Address:",
  "oldFileHash": "NAT Destination Port:",
  "ahost": "arcsight-test",
  "agt": "172.0.6.96",
  "agentZoneURI": "/All Zones/ArcSight System/Private Address Space
Zones/RFC1918: 10.0.0.0-10.255.255.255",
  "amac": "34-B3-54-BC-66-C6",
  "av": "7.14.0.8241.0",
  "atz": "Europe/Moscow",
  "at": "cisco_netflow",
  "dvchost": "arcsight-test",
  "dvc": "172.0.255.245",
  "deviceZoneURI": "/All Zones/ArcSight System/Private Address Space
Zones/RFC1918: 10.0.0.0-10.255.255.255",
  "dtz": "Europe/Moscow",
  "geid": "0",
```

```
  "_cefVer": "1.0",
  "ad.flow__sampler__id": "0",
  "ad.vendor__51": "0",
  "ad.DevicePort": "61673",
  "ad.interface__output__snmp": "312",
  "ad.src__tos": "0",
  "ad.pkthdr__uptime": "444691076",
  "ad.pkthdr__seq": "787165105",
  "ad.pkthdr__source__id": "517",
  "ad.pkthdr__count": "32",
  "ad.interface__input__snmp": "153",
  "aid": "3hughqHkBABCBSulnxz6OxA==",
  "nexthop": "172.0.245.1",
  "src_mask": "13",
  "dst_mask": "26",
  "in_pkts": "9",
  "tcp_flags": "0"
}
```


11. После успешной проверки этапов разбора необходимо нажать кнопку «Сохранить» для сохранения правила и следом нажать кнопку «Опубликовать» для последующего его использования.

> Описание и примеры использования возможных этапов разбора событий представлены в документации по описанию этапов разбора

7.2. Создание правил нормализации

Управление правилами нормализации осуществляется в разделе «Источники» → «Управление источниками». Далее выбрать вкладку «Правила нормализации», после чего откроется страница создания, редактирования и просмотра правил нормализации (см. рисунок 133).

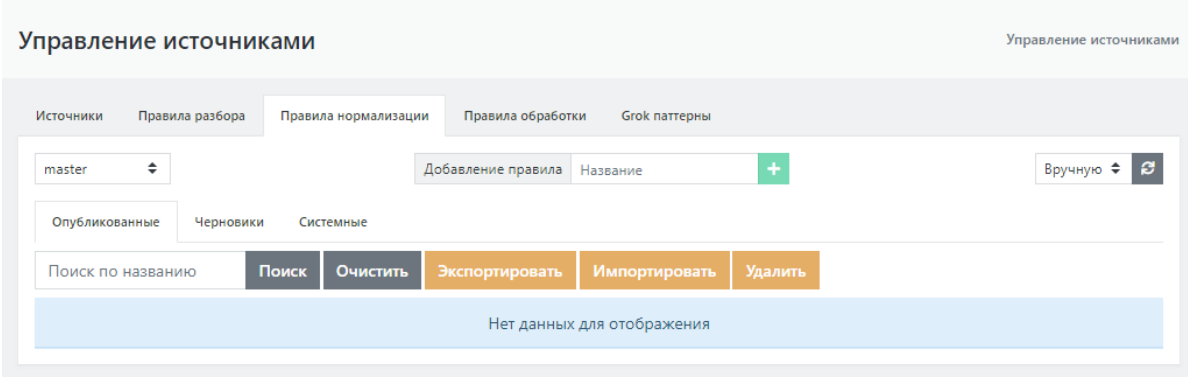

Рисунок 133 - Страница управления правилами нормализации

В разделе присутствует три вкладки:

- **Опубликованные** - правила, созданные пользователем и опубликованные на **Платформе Радар**.
- **Черновики** - правила, созданные пользователем, но не опубликованные на **Платформе Радар**. Такие правила будут применяться **Платформой Радар** только после их публикации.
- **Системные** - правила, которые поставляются с **Платформой Радар** и недоступные для редактирования.

Для опубликованных и неопубликованных правил нормализации доступны общие функции:

- *Экспортировать* - позволяет экспортировать выбранные правила нормализации в файл архива формата ZIP.
- *Импортировать* - позволяет импортировать правила нормализации из файла архива формата ZIP.
- *Удалить* - удаляет выбранные правила нормализации.

На каждой вкладке с правилами нормализации доступен поиск правила по его наименованию.

1. Для создания нового правила нормализации необходимо указать название в поле "Добавление правила" и нажать на “+”, после чего откроется окно создания правила нормализации, изображенное на рисунке 134;

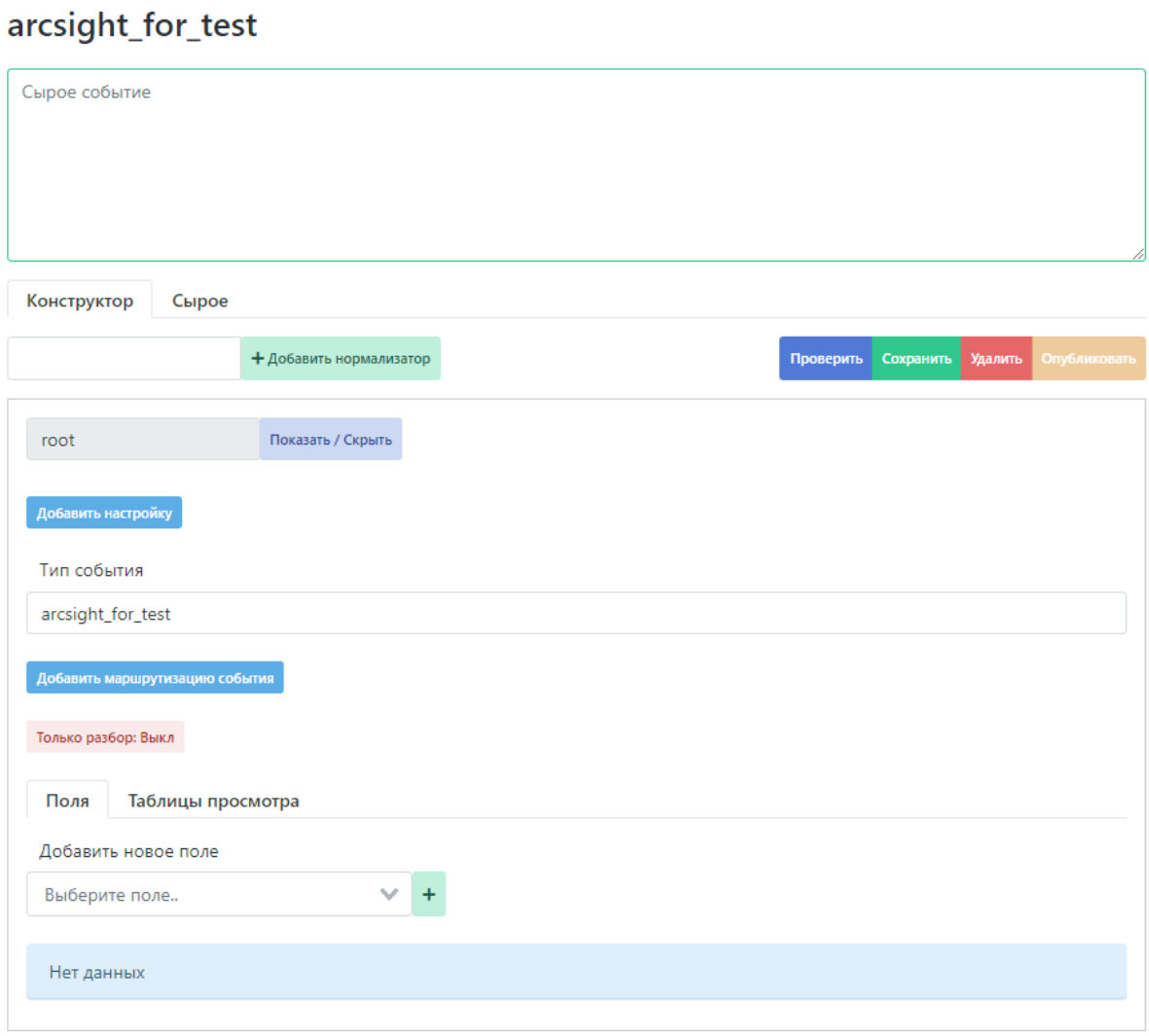

Рисунок 134 - Окно создания правила нормализации

```
> В качестве названия правила нормализации необходимо указывать уникальный идентификатор сообщения для данного источника, в случае с примером: **arcsight_for_test**
```

2. В поле для сырого события необходимо вставить разобранное событие (см. раздел результат создания правила разбора).
3. После создания правила, внутри него автоматически создается нормализатор root.yaml

> root.yaml – файл содержит декларации преобразований, общих для всех событий данной системы. Преобразования, указанные в этом файле применяются ко всем событиям системы, прошедшим стадию парсинга. Как правило они содержат классификатор источника и данные, содержащиеся в заголовке события Данный нормализатор разрабатывается один на систему.

4. Для добавления поля нормализации нужно во вкладке "Добавить новое поле" выбрать необходимое поле и нажать на "+", в результате чего в нормализатор добавится выбранное поле, как изображено на рисунке 135. Можно использовать как преднастроенные системные поля нормализации, так и добавлять пользовательские поля;

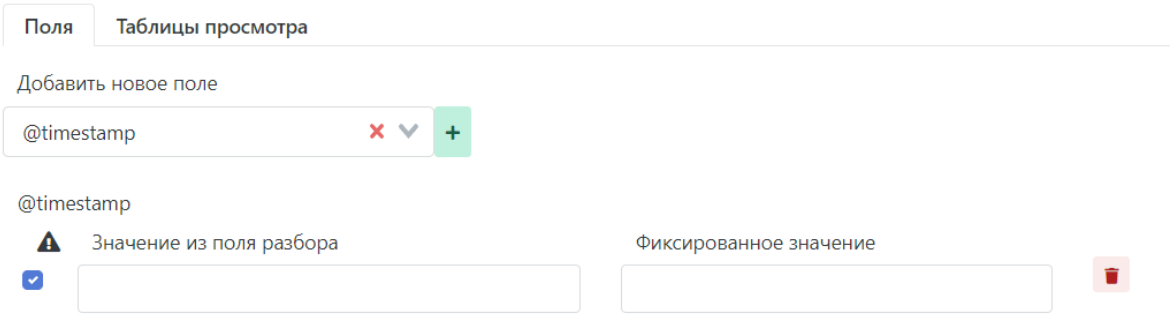

Рисунок 135 - Добавление поля нормализации

```
> В поле **"Значение из поля разбора"** нужно указывать одно из разобранных
полей, таким образом в поле нормализации будет записано значение поля из разбора.
Также в данное поле можно записывать специальные функции для работы с полями
(ниже будет представлено описание некоторых из них);
>
> В поле **"Фиксированное значение"** нужно указывать произвольный набор символов
соответствующий типу данного поля нормализации.
>
> Описание и типы полей нормализации представлены в разделе [Описание полей
нормализации](http://docs.pangeoradar.ru/events/processing_rules/scheme)
```

5. Исходя из этого, в данном файле нормализации можно использовать поля, изображенные на рисунках ниже;

root Показать / Скрыть

Добавить настройку

Тип события

arcsight_for_test

Добавить маршрутизацию события

Только разбор: Выкл

Поля | Таблицы просмотра

Добавить новое поле

Выберите поле.. +

event.logsource.vendor

Значение из поля разбора

Фиксированное значение

microfocus

@timestamp

Значение из поля разбора

epoch_to_timestamp(milliseconds_to_epoch(rt))

Фиксированное значение

event.logsource.product

Значение из поля разбора

Фиксированное значение

arcsight

event.logsource.name

Значение из поля разбора

Фиксированное значение

Microfocus ArcSight Smartconnector

Рисунок 136 - Поля нормализации для "root.yaml"

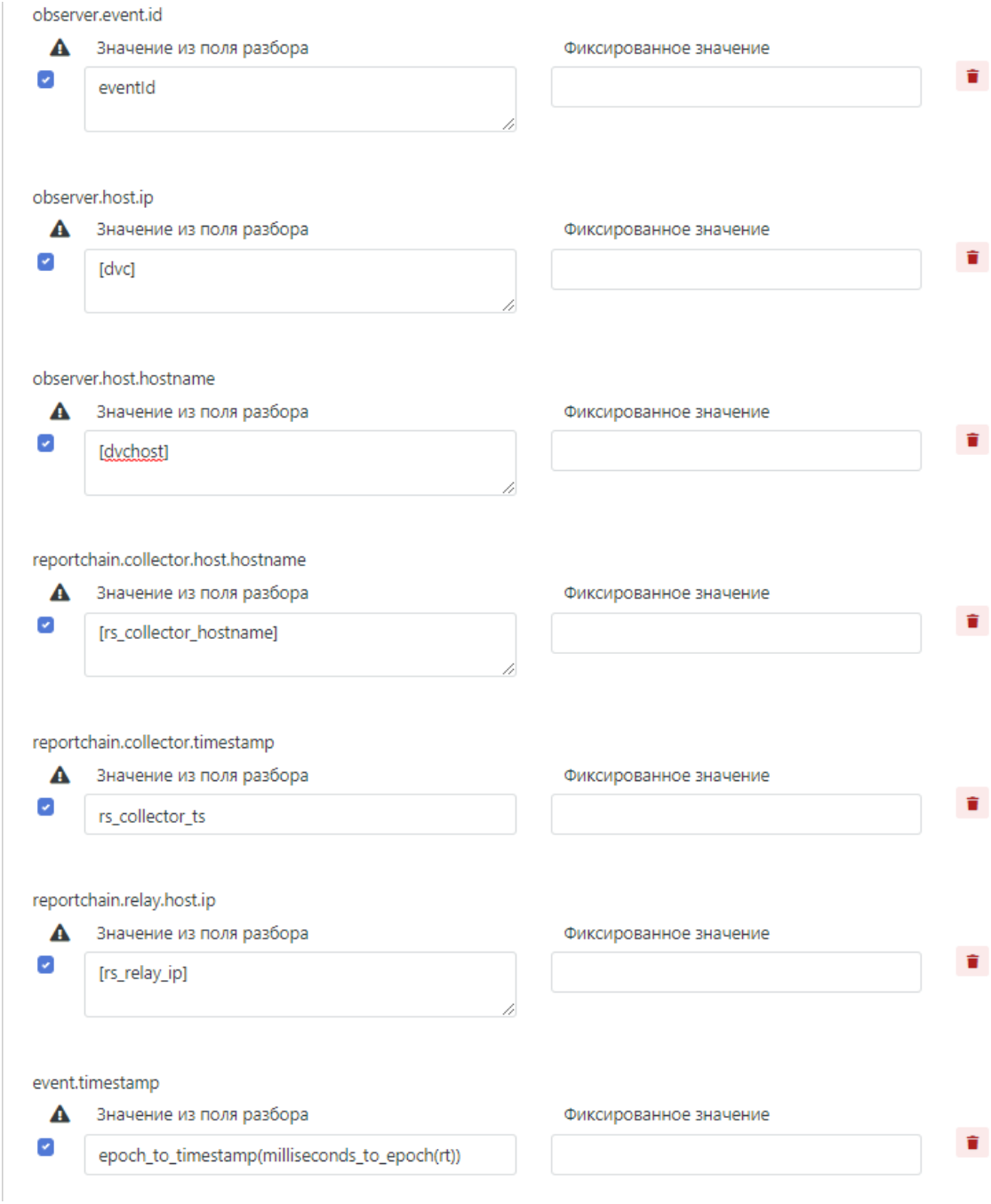

Рисунок 137 - Поля нормализации для "root.yaml"

6. После добавления необходимых для нормализации полей - нужно нажать на кнопку "Проверить" для проверки, что во все поля нормализации записались нужные данные. Результат проверки изображен на рисунке 138;

Результат проверки:


```
{
  "event": {
    "uuid": "297acb480ed2433ca67daa1a67fb63ef",
    "logsource": {
      "name": "Microfocus ArcSight Smartconnector",
      "product": "arcsight",
      "vendor": "microfocus"
    },
    "timestamp": "2021-06-09T07:31:13+00:00"
  },
  "raw": null,
  "@timestamp": "2021-06-09T07:31:13+00:00",
  "observer": {
    "event": {
      "id": "13252253246"
    },
    "host": {
      "hostname": [
        "arcsight-test"
      ],
      "ip": [
        "172.0.255.245"
      ]
    }
  },
  "reportchain": {
    "collector": {
      "host": {
        "hostname": [
          "radar-balancer-01"
        ]
      },
      "timestamp": "2021-06-09T10:41:02.253872+03:00"
    },
    "relay": {
      "host": {
        "ip": [
          "172.0.0.96"
        ]
      }
    }
  }
}
```


Рисунок 138 - Результат проверки нормализации

7. После настройки нормализатора root.yaml, необходимо перейти к созданию нормализатора для определенного типа событий от данного источника, в случае с примером - это Netflow;
8. Для этого в поле рядом с кнопкой "Добавить нормализатор" нужно ввести имя нормализатора и нажать на кнопку "+ Добавить нормализатор", как изображено на рисунке 139;

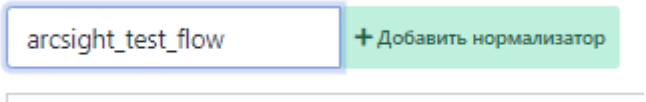

Рисунок 139 - Добавление нового нормализатора

9. Далее необходимо добавить маршрутизацию для данного нормализатора. Это делается для того, чтобы не все события от данного источника нормализовались по данному "сценарию

нормализации", а только те, которые нужны;

10. Для этого необходимо в поле "Маршрутизация события" ввести условия нормализации по данному сценарию. В качестве переменных в условии нужно использовать поля разобранного события. Заполненное поле маршрутизации изображено на рисунке 140;

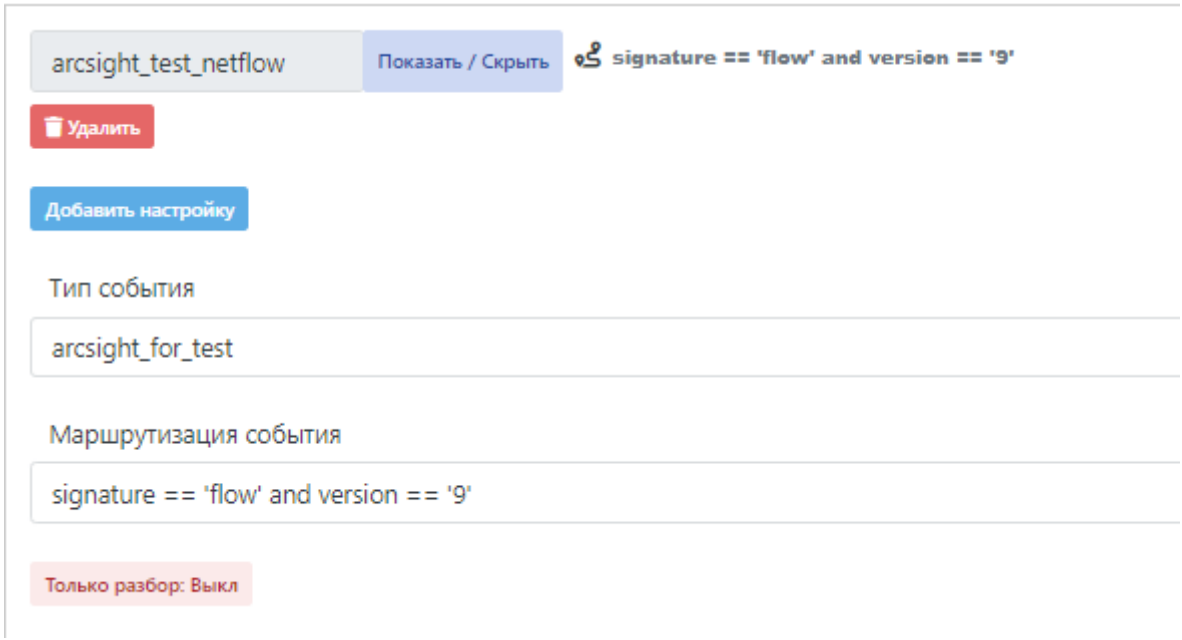

Рисунок 140 - Маршрутизация события

```
> Таким образом, все события, которые подходят под условие:
__signature == 'flow' and version == '9'__
будут нормализованы по данному файлу нормализации
```

11. Для более гибкой, понятной и правильной нормализации в данном файле нормализации используются специальные функции, которые подробно описаны в разделе *Специальные функции для работы с полями нормализации.*

12. Также в данном файле нормализации используются дополнительные настройки, описание которых представлены ниже;

> **Функция Tapping (Поле "Настройка")**
>
> К сожалению, логлайны, поступающие от клиентов, иногда могут быть непредсказуемыми.
> Таким образом, существует возможность выполнения кода Python в качестве этапа предварительной обработки.
> Событие доступно с помощью переменной *line*.
>
> Пример использования:
>
> ```
> tcp_flags = line.parsed['tcp']
>   line.parsed['flow_tags'] =
>   [
>       f"tcp_{flag}"
>       for flag in
>           ("syn", "fin", "rst", "psh", "ack", "cwr", "ecn", "urg")
>       if tcp_flags.get(flag, False)
>   ]
> ```

> В результате выполнения данной настройки, в нормализации можно использовать поле "flow_tags"
>
> > **ПРЕДУПРЕЖДЕНИЕ!**
> > Использование данного механизма может сказаться на производительности и скорости работы обработчиков событий.

13. Таким образом, таблица просмотра (функция lookup) для данного файла нормализации представлена на рисунке 141;

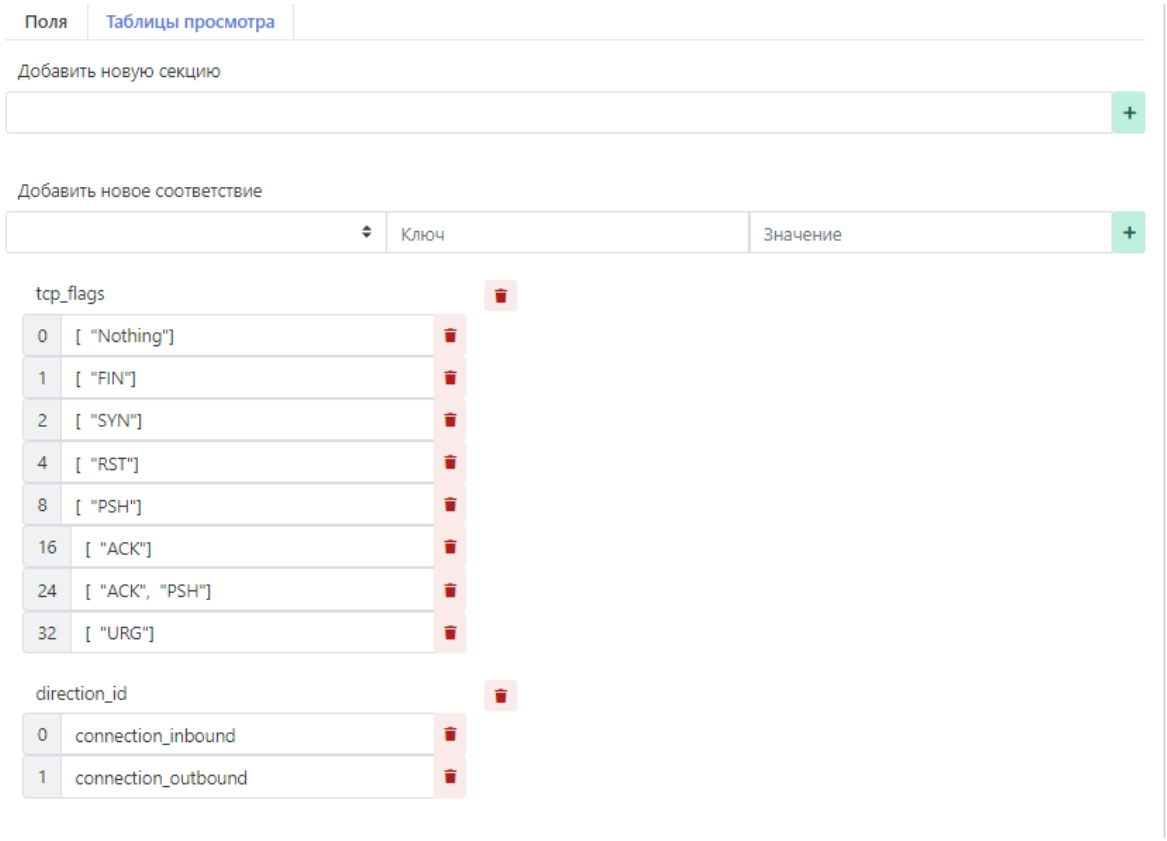

Рисунок 141 - Таблицы просмотра

14. Дополнительная настройка нормализатора изображена на рисунке 142;

Рисунок 142 - Дополнительная настройка нормализатора

> Данная настройка необходима для того чтобы в разобранном событии найти поле "in" и/или "out" и присвоить их в новые поля "in\out_bytes" в формате целых чисел.

15. Поля нормализации используемые в данном файле нормализации представлены на рисунках ниже;

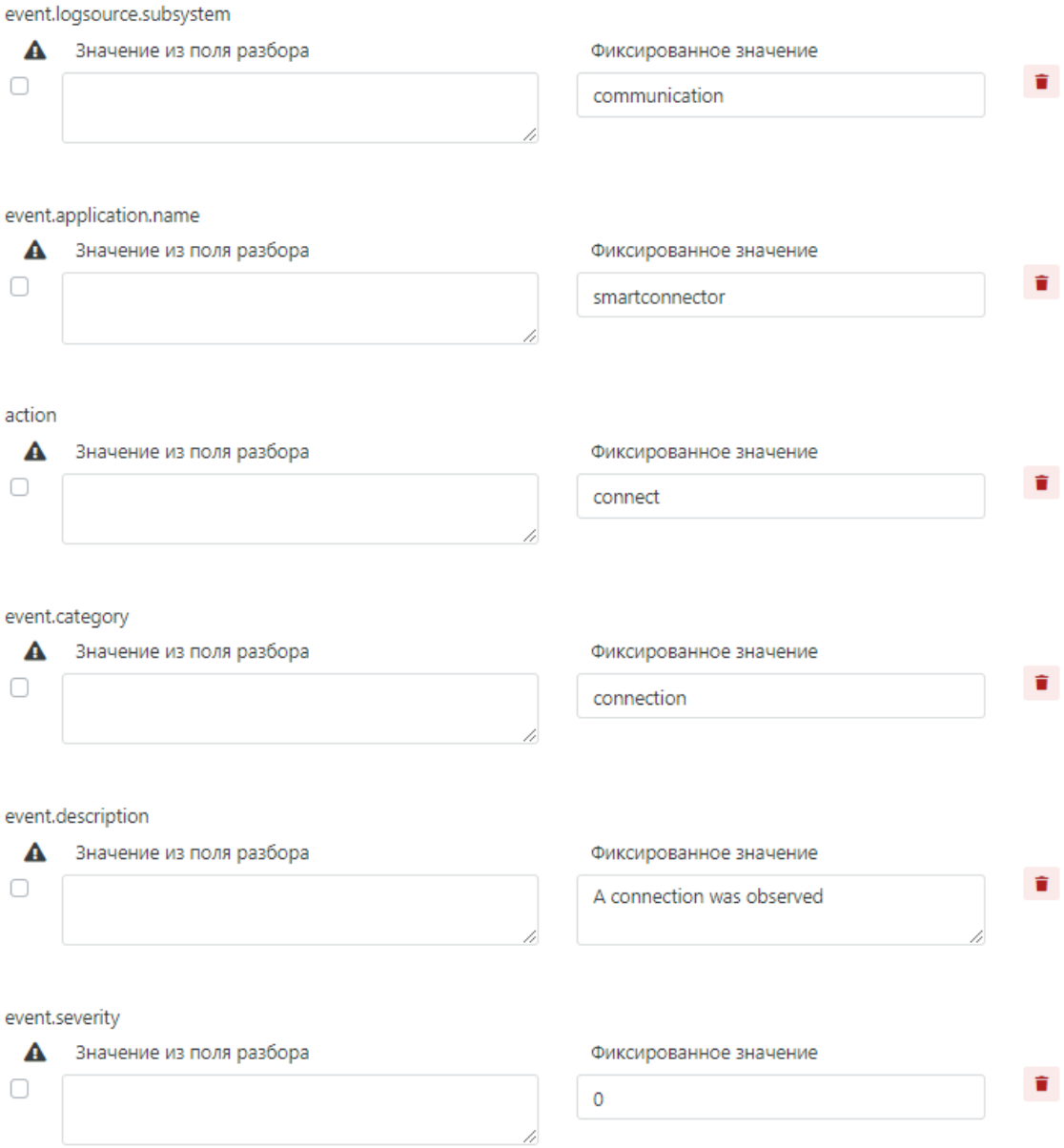

Рисунок 143 - Поля нормализации для нормализатора "arcsight_test_flow.yaml"

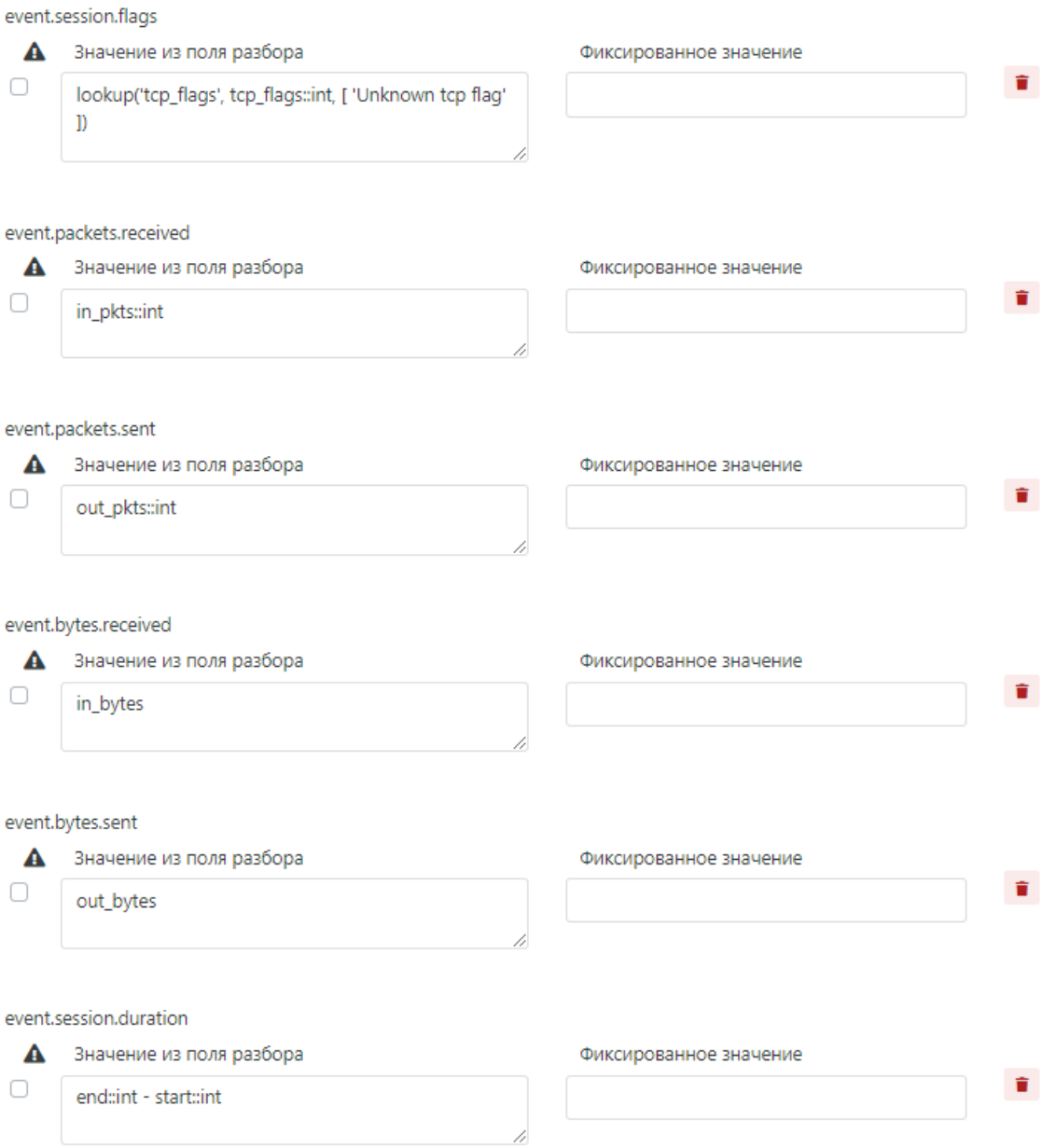

Рисунок 144 - Поля нормализации для нормализатора "arcsight_test_flow.yaml"

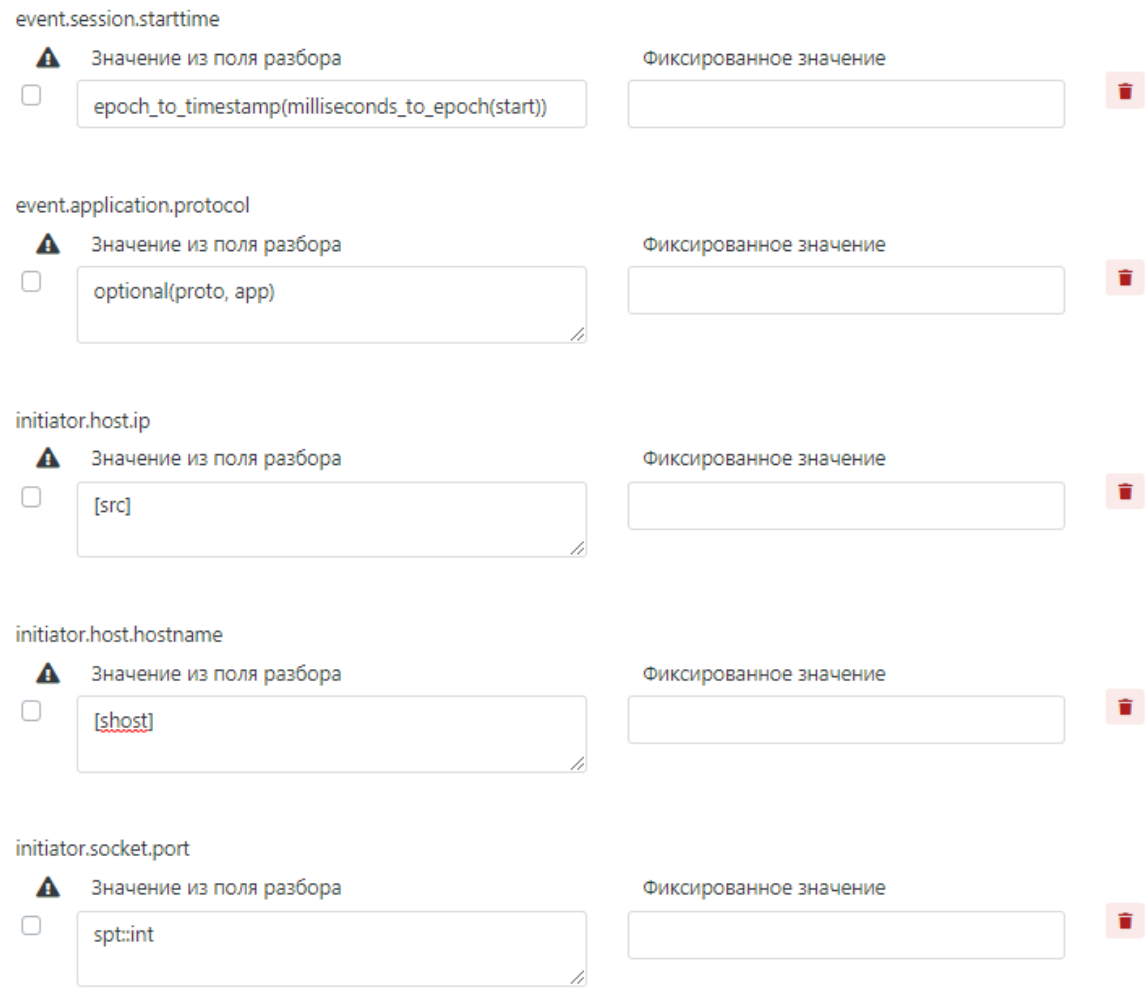

Рисунок 145 - Поля нормализации для нормализатора "arcsight_test_flow.yaml"

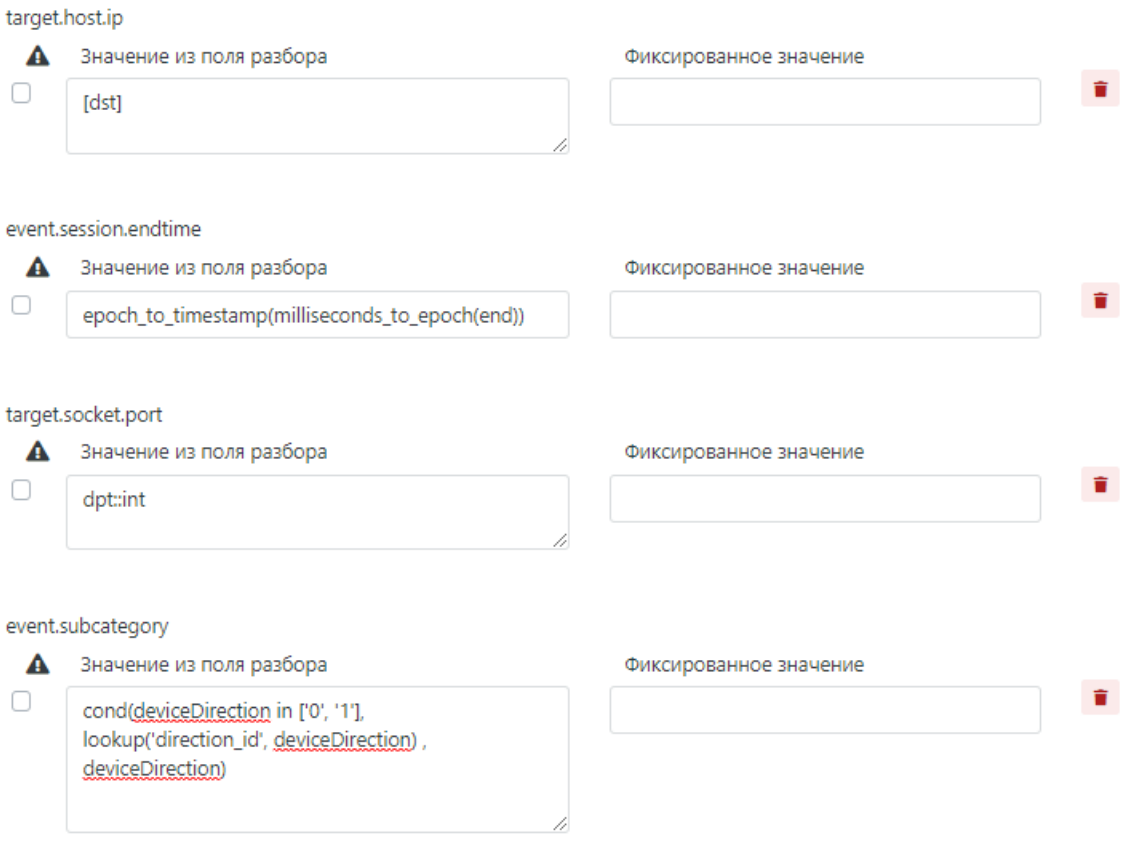

Рисунок 146 - Поля нормализации для нормализатора "arcsight_test_flow.yaml"

16. После добавления необходимых для нормализации полей, настроек и таблиц просмотра - нужно нажать на кнопку "Проверить" для проверки, что во все поля нормализации записались нужные данные;

    Результат проверки:


```
{
  "event": {
    "uuid": "283f402ba7a04a3786ca8a52e19be872",
    "application": {
      "name": "smartconnector",
      "protocol": "TCP"
    },
    "bytes": {
      "received": 1098
    },
    "category": "connection",
    "description": "A connection was observed",
    "logsource": {
      "application": "flow",
      "name": "Microfocus ArcSight Smartconnector",
      "product": "arcsight",
      "subsystem": "communication",
      "vendor": "microfocus"
    },
    "packets": {
      "received": 9
    },
    "session": {
      "duration": 64,
      "endtime": "2021-06-09T07:31:01.272000+00:00",
      "flags": [
        "Unknown tcp flag"
      ],
      "starttime": "2021-06-09T07:31:01.208000+00:00"
    },
    "severity": 0,
    "subcategory": "connection_inbound",
    "timestamp": "2021-06-09T07:31:13+00:00"
  },
  "raw": null,
  "@timestamp": "2021-06-09T07:31:13+00:00",
  "action": "connect",
  "initiator": {
    "host": {
      "ip": [
        "172.0.218.2"
      ]
    },
    "socket": {
      "port": 8787
    }
  },
  "observer": {
    "event": {
```

```
      "id": "13252253246"
    },
    "host": {
      "hostname": [
        "arcsight-test"
      ],
      "ip": [
        "172.0.255.245"
      ]
    }
  },
  "reportchain": {
    "collector": {
      "timestamp": "2021-06-09T10:41:02.253872+03:00"
    },
    "relay": {
      "host": {
        "ip": [
          "172.0.0.96"
        ]
      }
    }
  },
  "target": {
    "host": {
      "ip": [
        "172.0.18.108"
      ]
    },
    "socket": {
      "port": 53445
    }
  }
}
```


17. После настройки основного нормализатора arcsight_test_netflow.yaml, необходимо перейти к созданию нормализатора по умолчанию parsed_only.yaml;

> parsed_only.yaml – файл используется как «нормализатор по умолчанию». Для событий прошедших через этот нормализатор создается специализированных индекс в ElasticSearch, содержащий нормализованные данные. Данный нормализатор разрабатывается один на систему. В него попадают события, которые не прошли ни по одной из маршрутизаций в других нормализаторах

18. Для его создания, в поле названия нормализатора нужно ввести "parsed_only" и нажать на кнопку "+ Добавить нормализатор";
19. После чего, в маршрутизации события указать "fallback" и кликнуть на кнопку "Только разбор" чтобы она перешла в статус "Вкл", как изображено на рисунке 147;

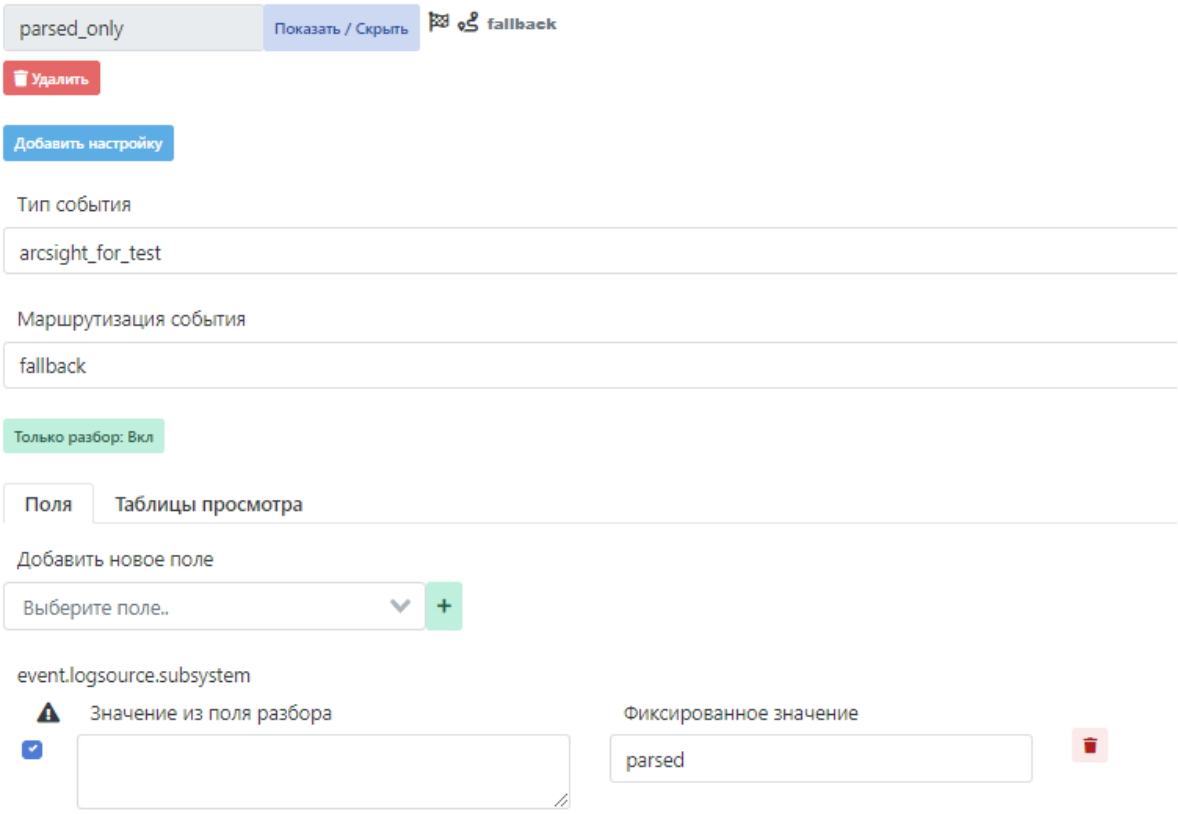

Рисунок 147 - Добавление нормализатора "parsed_only.yaml"

20. В данном нормализаторе не требуется добавления полей нормализации. В автоматически созданные поля необходимо указать "parsed" в поле "Фиксированное значение";
21. После - нужно еще раз провести проверку нормализации, нажав на кнопку "Проверить", если ошибки отсутствуют и в результате проверки ожидаемый результат можно перейти к сохранению нормализатора;
22. Для этого необходимо нажать кнопку "Сохранить" и следом кнопку "Опубликовать";
23. После чего в разделе "Правила нормализации" во вкладке "Опубликованные" должен появиться разработанный нормализатор. Это значит, что теперь его можно использовать.

7.3. Тестирование правил разбора и нормализации событий

Тестирование правил разбора и нормализации осуществляется в разделе «Источники» → «Управление источниками». Далее выбрать вкладку «Правила обработки», после чего откроется страница тестирования правил (см. рисунок 148).

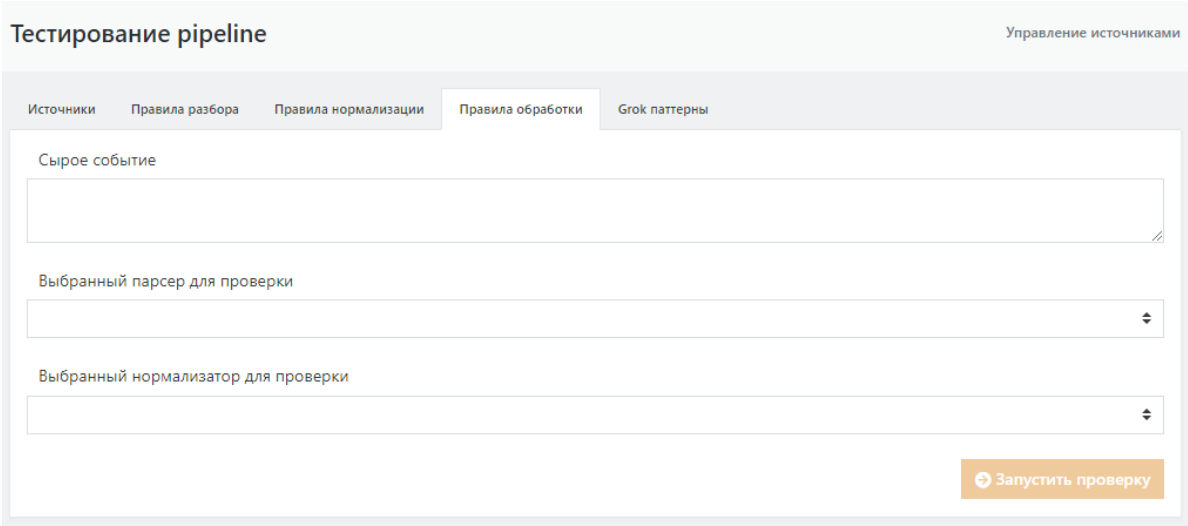

Рисунок 148 - Страница тестирования правил разбора и нормализации

В поле *Сырое событие* скопируйте сырое событие, полученное от источника.

В полях *Выбранный парсер для проверки* и *Выбранный нормализатор для проверки* выберите, соответственно, правило разбора и правило нормализации.

Далее нажмите кнопку **Запустить проверку**, после чего откроется окно с результатами проверки (см. рисунок 149).

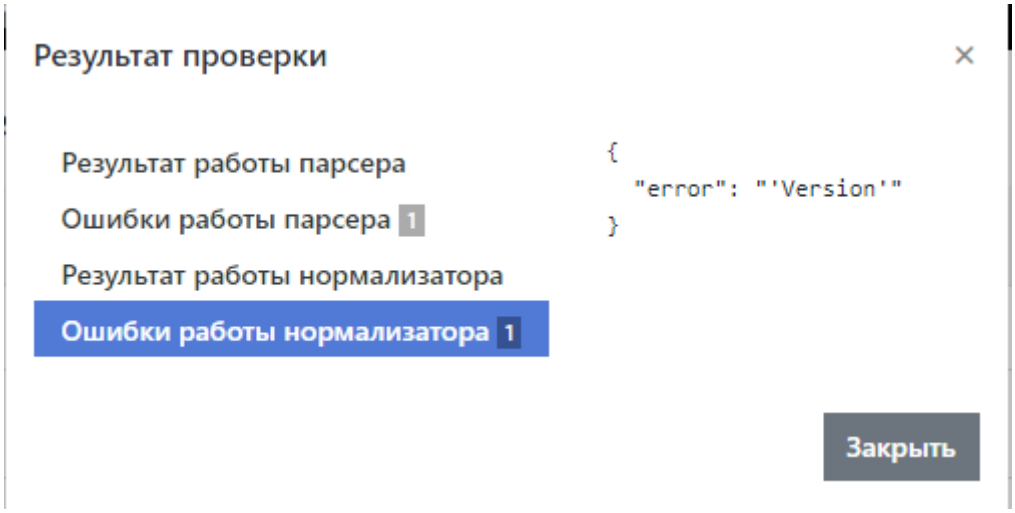

Рисунок 149 - Результат проверки правил

8. Описание полей нормализации

Поле	Тип данных	Обязательно	Описание
@timestamp	datetime_iso	Да	Временная метка
action	keyword	Да	Действия, выполненные инициатором
event.anomaly.description	text	Нет	Описание аномалии
event.anomaly.name	text	Нет	Название аномалии
event.application.category	[keyword]	Нет	Категория приложения
event.application.content-type	keyword	Нет	Тип контента, на которое ссылается приложение, например PNG
event.application.description	keyword	Нет	Дополнительное описание приложения

Поле	Тип данных	Обязательно	Описание
event.application.name	keyword	Нет	Наименование приложения например Web Browsing, Amazon Base, Microsoft Azure Base
event.application.protocol	keyword	Нет	Наименование протокола прикладного уровня, например FTP, WebDAV, Telnet
event.application.target	keyword	Нет	Тип цели, с которой работает приложение URL, Resource
event.application.vendor	keyword	Нет	Производитель приложения
event.application.version	keyword	Нет	Версия приложения
event.auth.key.length	integer	Нет	Длина ключа аутентификации
event.auth.method.description	text	Нет	Описание метода, используемого для аутентификации RDP, Network Authentication, Command Line, Web-Client
event.auth.method.id	keyword	Нет	Идентификатор метода аутентификации
event.auth.method.name	keyword	Нет	Наименование метода аутентификации, например keyboard-interactive, public key, service, batch
event.auth.protocol.name	keyword	Нет	Наименование метода аутентификации, например, SSH, NTML, Kerberos (AuthenticationPackageName in Windows)
event.auth.protocol.version	keyword	Нет	Версия протокола аутентификации
event.blacklist	blacklist	Нет	Черный список
event.bytes.received	integer	Нет	Количество полученных байт в рамках сессии, Цель -> Инициатор
event.bytes.sent	integer	Нет	Количество отправленных байт в рамках сессии, Инициатор -> Цель
event.bytes.total	integer	Нет	Общее количество байт, отправленных в рамках сессии
event.category	keyword	Да	Категория в рамках приложения/подсистемы
event.context.raw	raw_text	Нет	Контекст события
event.correlation.id	keyword	Нет	Идентификатор сессии, позволяющий связать события
event.correlation.sequence	integer	Нет	Количество последовательных сессий
event.correlation.total	integer	Нет	Количество сессий
event.description	text	Да	Текстовое описание события
event.dns.answer.host.fqdn	[domain]	Нет	FQDN-имя на которое получен DNS-ответ
event.dns.answer.host.hostname	[domain]	Нет	Hostname на который получен DNS-ответ
event.dns.answer.host.ip	[ip]	Нет	IP адрес на который получен DNS-ответ
event.dns.answer.original	keyword	Нет	Оригинальный DNS-ответ
event.dns.id	keyword	Нет	Идентификатор DNS-запроса
event.dns.query.host.fqdn	[domain]	Нет	FQDN-имя запрашиваемое в рамках DNS-запроса
event.dns.query.host.hostname	[domain]	Нет	Hostname, запрашиваемый в рамках DNS-запроса
event.dns.query.host.ip	[ip]	Нет	IP-адрес, запрашиваемый в рамках DNS-запроса
event.dns.query.original	keyword	Нет	Оригинальный DNS-запрос
event.dns.ttl	integer	Нет	DNS time to live (время жизни)

Поле	Тип данных	Обязательно	Описание
event.dns.type	keyword	Нет	Тип DNS-записи https://www.iana.org/assignments/dns-parameters/dns-parameters.xhtml#dns-parameters-4
event.endtime	datetime_iso	Нет	Время завершения события
event.eventlist	keyword	Нет	Список событий
event.file.hash.md5	keyword	Нет	MD5-хеш файла
event.file.hash.sha1	keyword	Нет	SHA1-хеш файла
event.file.hash.sha256	keyword	Нет	SHA256-хеш файла
event.flow.id	keyword	Нет	Идентификатор Flow
event.finding.id	keyword	Нет	ID артефакта
event.finding.name	keyword	Нет	Название артефакта
event.http.protocol.name	keyword	Нет	Наименование HTTP-протокола (HTTP)
event.http.protocol.version	keyword	Нет	Версия HTTP-протокола
event.logsource.application	keyword	Да	Приложение породившее событие
event.logsource.host	keyword	Нет	Хост источника события
event.logsource.input	keyword	Нет	Input источника события
event.logsource.language	keyword	Нет	Язык источника события
event.logsource.name	keyword	Да	Наименование источника события
event.logsource.product	keyword	Да	Наименование продукта источника
event.logsource.subsystem	keyword	Да	Наименование подсистемы источника
event.logsource.vendor	keyword	Да	Наименование вендора источника
event.packet.payload.printable	text	Нет	Человекочитаемые данные из полезной нагрузки
event.packet.payload.raw	raw_text	Нет	Данные полезной нагрузки
event.packet.raw	raw_text	Нет	Сырые данные из пакета
event.packets.received	integer	Нет	Количество пакетов, полученных в рамках сессии, Цель -> Инициатор
event.packets.sent	integer	Нет	Количество пакетов, отправленных в рамках сессии, Инициатор -> Цель
event.packets.total	integer	Нет	Количество пакетов, переданных в рамках сессии
event.result.analysis_output	text	Нет	Результат анализа
event.result.description	text	Нет	Описание результата
event.result.id	keyword	Нет	ID результата
event.result.incident_identifier	keyword	Нет	Идентификатор инцидента результата
event.result.mitigation	text	Нет	
event.result.name	keyword	Нет	Наименование результата
event.result.risk_impact	text	Нет	
event.result.solution	text	Нет	Решение
event.result.synopsis	text	Нет	Краткое изложение
event.service.name	keyword	Нет	Сервис, передающий событие, например HTTP
event.session.duration	integer	Нет	Длительность сессии (в секундах)
event.session.endtime	datetime_iso	Нет	Время окончания сессии

Поле	Тип данных	Обязательно	Описание
event.session.flags	[keyword]	Нет	TCP-флаги окончания сессии
event.session.id	keyword	Нет	Идентификатор сессии
event.session.starttime	datetime_iso	Нет	Время начала сессии
event.severity	float	Да	Severity события, получаемое из заголовка Syslog, по умолчанию: 0
event.socket.protocol	keyword	Нет	Протокол транспортного уровня, например, TCP, UDP
event.subcategory	keyword	Да	Подкатегория события
event.timestamp	datetime_iso	Да	Время, в которое произошло событие
event.tls.fingerprint	keyword	Нет	TLS Certificate Fingerprint
event.tls.issuerdn	text	Нет	TLS Certificate Issuer DN
event.tls.not-after	datetime_iso	Нет	TLS Certificate date validation
event.tls.not-before	datetime_iso	Нет	TLS Certificate date validation
event.tls.sni	domain	Нет	TLS Certificate SNI
event.tls.subject	text	Нет	TLS Certificate Subject
event.uuid	keyword	Нет	UUID события
event.worker.host	keyword	Нет	
event.worker.ip	keyword	Нет	
initiator.antivirus.scan.endtime	datetime_iso	Нет	Время окончания антивирусного сканирования
initiator.antivirus.scan.starttime	datetime_iso	Нет	Время начала антивирусного сканирования
initiator.antivirus.scan.type	keyword	Нет	Тип антивирусного сканирования, например: On-Access, Schedule Scan, Quick Scan, Custom Scan...
initiator.command.executed	text	Нет	Выполненная команда
initiator.command.info	keyword	Нет	Информация о команде
initiator.command.path.original	keyword	Нет	
initiator.command.type	keyword	Нет	Тип конманды
initiator.file.hash.md5	keyword	Нет	MD5-хеш файла
initiator.file.hash.sha1	keyword	Нет	SHA1-хеш файла
initiator.file.hash.sha256	keyword	Нет	SHA256-хеш файла
initiator.geoip	geo	Нет	Данные GeoIP (автоматически проставляются подсистемой обработки событий)
initiator.host.fqdn	[domain]	Нет	FQDN инициатора
initiator.host.hostname	[domain]	Нет	Hostname инициатора
initiator.host.ip	[ip]	Нет	IP инициатора
initiator.http.method	keyword	Нет	https://wiki.squid-cache.org/SquidFaq/SquidLogs#Request_methods
initiator.http.user-agent	user_agent	Нет	HTTP User Agent
initiator.interface.mac	mac	Нет	MAC-адрес инициатора
initiator.interface.name	keyword	Нет	Имя интерфейса инициатора
initiator.nat.ip	ip	Нет	NAT IP-адрес
initiator.nat.port	port	Нет	NAT порт
initiator.process.command	text	Нет	Выполненная команда

Поле	Тип данных	Обязательно	Описание
initiator.process.guid	keyword	Нет	GUID-процесса
initiator.process.id	keyword	Нет	ID-процесса
initiator.process.hash.impash	keyword	Нет	impash-процесса
initiator.process.hash.md5	keyword	Нет	md5-процесса
initiator.process.hash.sha1	keyword	Нет	sha1-процесса
initiator.process.hash.sha256	keyword	Нет	sha256-процесса
initiator.process.parent.id	keyword	Нет	
initiator.process.hash.path.original	keyword	Нет	
initiator.process.path.drive	keyword	Нет	Диск, на котором запущен процесс C:, \ (сетевой каталог)
initiator.process.path.extension	keyword	Нет	Расширение запущенного файла
initiator.process.path.full	path	Нет	Полный путь до запущенного файла e.g. C:\Windows\System32\service.exe, \radarservices\company\secret.txt
initiator.process.path.name	keyword	Нет	Имя процесса, например: service, secret
initiator.process.path.original	keyword	Нет	Оригинальное имя процесса
initiator.process.path.path	path	Нет	Каталог в котором запущен процесс
initiator.process.working-directory	path	Нет	Рабочий каталог процесса
initiator.registry.path.original	keyword	Нет	
initiator.session.id	keyword	Нет	Logon-сессия связанная с инициатором (Windows: SubjectLogonID)
initiator.shell.name	keyword	Нет	Название shell
initiator.shell.version	keyword	Нет	Версия shell
initiator.socket.port	port	Нет	Порт инициатора
initiator.user.domain	keyword	Нет	Домен, которому принадлежит пользователь-инициатор
initiator.user.group.id	keyword	Нет	ID группы пользователя-инициатора
initiator.user.group.name	keyword	Нет	Имя группы пользователя-инициатора
initiator.user.id	keyword	Нет	ID пользователя, например SID или UID
initiator.user.name	keyword	Нет	Имя пользователя-инициатора
initiator.user.privileges.code	[keyword]	Нет	
initiator.user.privileges.description	[keyword]	Нет	
initiator.user.privileges.name	[keyword]	Нет	
initiator.user.privileges.original	[keyword]	Нет	
initiator.user.subcategory	text	Нет	
initiator.vpn.host.ip	[ip]	Нет	IP адрес, назначенный VPN-сервером
observer.blacklist	blacklist	Нет	
observer.event.id	keyword	Нет	ID-события
observer.event.type	keyword	Нет	Тип события Windows Channel
observer.file.hash.md5	keyword	Нет	MD5-хеш файла
observer.file.hash.sha1	keyword	Нет	SHA1-хеш файла
observer.file.hash.sha256	keyword	Нет	SHA256-хеш файла
observer.socket.port	port	Нет	Порт обзервера
observer.host.fqdn	[domain]	Нет	FQDN обзервера

Поле	Тип данных	Обязательно	Описание
observer.host.hostname	[domain]	Нет	Hostname обзервера
observer.host.ip	[ip]	Нет	IP обзервера
observer.interface.in.mac	mac	Нет	MAC-адрес интерфейса, на который получено событие
observer.interface.in.name	keyword	Нет	Имя интерфейса, на который получено событие
observer.interface.out.mac	mac	Нет	MAC-адрес интерфейса, с которого отправлено событие
observer.interface.out.name	keyword	Нет	Имя интерфейса, с которого отправлено событие
observer.rule.category	keyword	Нет	Категория правила, например в Suricata "Potentially Bad Traffic", "Misc Attack"
observer.rule.id	keyword	Нет	ID правила, по которому сгенерировалось событие, например: Suricata SID, Firewall rule ID
observer.rule.metadata.affected-product	keyword	Нет	Приложение, подверженное атаке
observer.rule.metadata.attack-target	keyword	Нет	Тип атакуемой цели
observer.rule.metadata.deployment	[keyword]	Нет	Тип развертывания
observer.rule.metadata.malware-family	keyword	Нет	Семейство вредоносного кода, обнаруживаемое правилом
observer.rule.name	keyword	Нет	Наименование правила
observer.rule.original	text	Нет	Исходный текст правила
observer.rule.threshold.count	integer	Нет	Сработавший порог по количеству для правила
observer.rule.threshold.seconds	integer	Нет	Сработавший порог по времени для правила
observer.rule.threshold.track	keyword	Нет	
observer.rule.threshold.type	keyword	Нет	
observer.zone.in.name	keyword	Нет	Имя сетевой зоны (inbound)
observer.zone.out.name	keyword	Нет	Имя сетевой зоны (outbound)
outcome.description	text	Нет	Описание результата
outcome.name	keyword	Нет	Нормализованное представление результата
outcome.original	keyword	Нет	Вендор-специфичное представление для результата
raw	raw_text	Нет	Изначальное событие
executed.description	text	Нет	Описание реакции на событие
reaction.executed.name	keyword	Нет	Нормализованное представление реакции на событие
reaction.executed.original	keyword	Нет	Вендор-специфичное представление реакции на событие
reaction.executed.reason	keyword	Нет	Описание причины применения указанной реакции на событие
reaction.executed.user.domain	keyword	Нет	Домен пользователя
reaction.executed.user.id	keyword	Нет	ID пользователя
reaction.executed.user.name	keyword	Нет	Логин пользователя
reaction.requested.description	text	Нет	Описание требуемой реакции

Поле	Тип данных	Обязательно	Описание
reaction.requested.name	keyword	Нет	Нормализованное представление требуемой реакции
reaction.requested.original	keyword	Нет	Вендор-специфичное представление требуемой реакции
reaction.requested.reason	keyword	Нет	Описание причин требуемой реакции
reportchain.collector.host.fqdn	[domain]	Да	FQDN модуля Платформы Радар получившего событие
reportchain.collector.host.hostname	[domain]	Нет	Hostname модуля Платформы Радар, получившего событие
reportchain.collector.host.ip	[ip]	Нет	IP модуля Платформы Радар, получившего событие
reportchain.collector.timestamp	datetime_iso	Да	Время получения события Платформой Радар
reportchain.relay.host.fqdn	[domain]	Нет	FQDN хоста, отправившего событие по syslog
reportchain.relay.host.hostname	[domain]	Нет	Hostname хоста, отправившего событие по syslog
reportchain.relay.host.ip	[ip]	Нет	IP хоста, отправившего событие по syslog
reportchain.relay.timestamp	datetime_iso	Нет	Отметка времени получения события хостом с NxLog (временная отметка агента)
tags	[keyword]	Нет	Тэги
target.auth.encryption	keyword	Нет	Ticket Encryption Type
target.access_mask.original	keyword	Нет	
target.auth.options.name	keyword	Нет	Ticket Options
target.auth.options.original	keyword	Нет	
target.auth.process.name	keyword	Нет	Процесс, выполняющий аутентификацию sshd, Schannel, Advapi (LogonProcessName in Windows)
target.auth.service.domain	keyword	Нет	
target.auth.service.id	keyword	Нет	
target.auth.service.name	keyword	Нет	Наименование сервиса в Kerberos Realm, которому был отправлен TGT-запрос
target.command.executed	text	Нет	Выполненная команда
target.command.path.original	keyword	Нет	Путь до запущенного процесса
target.config.changes.description	[keyword]	Нет	Тип изменений в конфигурации
target.config.changes.id	[keyword]	Нет	ID изменений в конфигурации
target.database.name	keyword	Нет	Название БД
target.email.file.drive	[path]	Нет	Информация о email-аттаче
target.email.file.extention	[keyword]	Нет	Информация о email-аттаче
target.email.file.fullname	[keyword]	Нет	Информация о email-аттаче
target.email.file.name	[keyword]	Нет	Информация о email-аттаче
target.email.file.path	[path]	Нет	Информация о email-аттаче
target.email.receivers	[keyword]	Нет	Email-адреса получатели письма
target.email.sender	keyword	Нет	Email-адрес отправителя
target.email.subject	text	Нет	Тема письма
target.email.url.full	[keyword]	Нет	URL в письме
target.email.url.host.fqdn	[domain]	Нет	URL в письме

Поле	Тип данных	Обязательно	Описание
target.email.url.host.hostname	[domain]	Нет	URL в письме
target.email.url.host.ip	[ip]	Нет	URL в письме
target.file.content-type	text	Нет	Content-Type файла
target.file.drive	path	Нет	Диск, на котором находится файл
target.file.extension	keyword	Нет	Расширение файла
target.file.fullname	keyword	Нет	Полное имя файла
target.file.hash.md5	keyword	Нет	MD5-хеш файла
target.file.hash.sha1	keyword	Нет	SHA1-хеш файла
target.file.hash.sha256	keyword	Нет	SHA256-хеш файла
target.file.name	keyword	Нет	Имя файла
target.file.path	path	Нет	Полный путь до файла
target.file.size	integer	Нет	Размер файла (в байтах)
target.host.geoip	geo	Нет	Данные GeoIP (автоматически проставляются подсистемой обработки событий)
target.group.domain	keyword	Нет	Домен группы
target.group.id	keyword	Нет	ID группы
target.group.name	keyword	Нет	Имя группы
target.host.fqdn	[domain]	Нет	FQDN хоста
target.host.hostname	[domain]	Нет	Hostname
target.host.ip	[ip]	Нет	IP-адрес хоста
target.http.content-type	keyword	Нет	Content type ответа, например, text/html
target.http.redirect.host.fqdn	[domain]	Нет	Host part of the redirected URL
target.http.redirect.host.hostname	[domain]	Нет	Host part of the redirected URL
target.http.redirect.host.ip	[ip]	Нет	Host part of the redirected URL
target.http.redirect.path	[text]	Нет	Path of the redirected URL http://hostname.tld:port/path
target.http.redirect.port	[port]	Нет	Port of the redirected URL
target.http.redirect.protocol	[keyword]	Нет	Protocol of the redirected URL (http or https)
target.http.referer.host.fqdn	[domain]	Нет	Host part of the referer URL
target.http.referer.host.hostname	[domain]	Нет	Host part of the referer URL
target.http.referer.host.ip	[ip]	Нет	Host part of the referer URL
target.http.referer.path	[text]	Нет	Path of the referer URL http://hostname.tld:port/path
target.http.referer.port	[port]	Нет	Port of the referer URL
target.http.referer.protocol	[keyword]	Нет	Protocol of the referer URL (http or https)
target.http.status.code	integer	Нет	https://wiki.squid-cache.org/SquidFaq/SquidLogs#HTTP_status_codes
target.http.status.description	text	Нет	https://wiki.squid-cache.org/SquidFaq/SquidLogs#HTTP_status_codes
target.http.status.name	keyword	Нет	https://wiki.squid-cache.org/SquidFaq/SquidLogs#HTTP_status_codes
target.http.url.host.fqdn	[domain]	Нет	Host part of the targeted URL
target.http.url.host.hostname	[domain]	Нет	Host part of the targeted URL

Поле	Тип данных	Обязательно	Описание
target.http.url.host.ip	[ip]	Нет	Host part of the targeted URL
target.http.url.path	[text]	Нет	Path of the targeted URL http://hostname.tld:port/path
target.http.url.port	[port]	Нет	Port of the targeted URL
target.http.url.protocol	[keyword]	Нет	Protocol of the targeted URL (http or https)
target.interface.mac	mac	Нет	MAC-адрес интерфейса
target.interface.name	keyword	Нет	Имя интерфейса
target.nat.ip	ip	Нет	NAT IP-адрес
target.nat.port	port	Нет	NAT порт
target.object.attribute.name	keyword	Нет	Атрибут объекта, который был модифицирован
target.object.attribute.value	text	Нет	Значение атрибута
target.object.domain	keyword	Нет	Домен объекта, который был модифицирован
target.object.id	keyword	Нет	ID объекта, который был модифицирован
target.object.name	keyword	Нет	Имя объекта, который был модифицирован
target.object.type	keyword	Нет	Класс объекта, который был модифицирован
target.object.server	keyword	Нет	Сервер объекта, который был модифицирован
target.object.handle.id	keyword	Нет	
target.permissons.granted.name	[keyword]	Нет	
target.permissons.granted.original	[keyword]	Нет	
target.permissons.requested.description	[keyword]	Нет	
target.permissons.requested.name	[keyword]	Нет	
target.permissons.requested.original	[keyword]	Нет	
target.policy.category.description	text	Нет	https://docs.microsoft.com/en-us/windows/security/threat-protection/auditing/event-4719
target.policy.category.id	keyword	Нет	https://docs.microsoft.com/en-us/windows/security/threat-protection/auditing/event-4719
target.policy.changes.description	[text]	Нет	https://docs.microsoft.com/en-us/windows/security/threat-protection/auditing/event-4719
target.policy.changes.id	[keyword]	Нет	https://docs.microsoft.com/en-us/windows/security/threat-protection/auditing/event-4719
target.policy.subcategory.description	text	Нет	https://docs.microsoft.com/en-us/windows/security/threat-protection/auditing/event-4719
target.policy.subcategory.id	keyword	Нет	https://docs.microsoft.com/en-us/windows/security/threat-protection/auditing/event-4719
target.process.args	keyword	Нет	Аргументы
target.process.command	text	Нет	Выполненная команда
target.process.guid	keyword	Нет	GUID процесса
target.process.hash.impash	keyword	Нет	impash-хеш файла
target.process.hash.md5	keyword	Нет	MD5-хеш файла
target.process.hash.sha1	keyword	Нет	SHA1-хеш файла
target.process.hash.sha256	keyword	Нет	SHA256-хеш файла
target.process.id	keyword	Нет	ID процесса

Поле	Тип данных	Обязательно	Описание
target.process.integrity.description	keyword	Нет	https://docs.microsoft.com/en-us/windows/security/threat-protection/auditing/event-4688
target.process.integrity.id	keyword	Нет	https://docs.microsoft.com/en-us/windows/security/threat-protection/auditing/event-4688
target.process.integrity.id-hex	keyword	Нет	https://docs.microsoft.com/en-us/windows/security/threat-protection/auditing/event-4688
target.process.integrity.name	keyword	Нет	https://docs.microsoft.com/en-us/windows/security/threat-protection/auditing/event-4688
target.process.path.drive	keyword	Нет	Диск на котором запущен процесс C:, \ (сетевой каталог)
target.process.path.extension	keyword	Нет	Расширение запущенного файла
target.process.path.full	path	Нет	Полный путь до запущенного файла e.g. C:\Windows\System32\service.exe, \radarservices\company\secret.txt
target.process.path.name	keyword	Нет	Имя процесса, например: service, secret
target.process.path.original	text	Нет	Оригинальное имя процесса
target.process.path.path	path	Нет	Каталог, в котором запущен процесс
target.process.path.file.internal.name	keyword	Нет	Имя файла
target.process.privileges.code	keyword	Нет	https://docs.microsoft.com/en-us/windows/security/threat-protection/auditing/event-4688
target.process.privileges.description	[keyword]	Нет	https://docs.microsoft.com/en-us/windows/security/threat-protection/auditing/event-4688
target.process.privileges.original	[keyword]	Нет	https://docs.microsoft.com/en-us/windows/security/threat-protection/auditing/event-4688
target.process.privileges.description	[keyword]	Нет	https://docs.microsoft.com/en-us/windows/security/threat-protection/auditing/event-4688
target.process.working-directory	path	Нет	
target.registry.path.original	keyword	Нет	
target.instance.name	keyword	Нет	https://docs.microsoft.com/en-us/windows/security/threat-protection/auditing/event-4688
target.rule.action	keyword	Нет	Действие, выполненное на межсетевом экране: allow, bypass, deny, log only, discard/reject
target.rule.chain	keyword	Нет	Windows: inbound или outbound, Linux: цепочка iptables
target.rule.dst-addresses	[keyword]	Нет	IP-адрес в правиле межсетевого экрана
target.rule.dst-ports	[keyword]	Нет	Порт в правиле межсетевого экрана
target.rule.id	keyword	Нет	https://docs.microsoft.com/en-us/windows/security/threat-protection/auditing/event-4946
target.rule.name	keyword	Нет	https://docs.microsoft.com/en-us/windows/security/threat-protection/auditing/event-4946
target.rule.profiles	[keyword]	Нет	https://docs.microsoft.com/en-us/windows/security/threat-protection/auditing/event-4946
target.rule.src-addresses	[keyword]	Нет	IP-адрес источника в правиле межсетевого экрана
target.rule.src-ports	[keyword]	Нет	Порт источника в правиле межсетевого экрана
target.rule.status	keyword	Нет	Индикатор активности правила
target.service.name	keyword	Нет	Название сервиса

Поле	Тип данных	Обязательно	Описание
target.path.original	keyword	Нет	
target.service.start_type.new.description	keyword	Нет	
target.service.start_type.new.name	keyword	Нет	
target.service.start_type.new.original	keyword	Нет	
target.service.start_type.old.description	keyword	Нет	
target.service.start_type.old.name	keyword	Нет	
target.service.start_type.old.original	keyword	Нет	
target.service.status.name	keyword	Нет	
target.service.status.original	keyword	Нет	
target.service.type.description	keyword	Нет	
target.service.type.name	keyword	Нет	
target.service.type.original	keyword	Нет	
target.session.id	keyword	Нет	ID сессии (Windows: TargetLogonID)
target.share.local_path.original	keyword	Нет	
target.share.relative_name.original	keyword	Нет	
target.share.remote_path.original	keyword	Нет	
target.shell.name	keyword	Нет	
target.shell.version	keyword	Нет	Версия shell
target.syscall.id	keyword	Нет	syscall id
target.syscall.name	keyword	Нет	Системный вызов
target.task.args	keyword	Нет	Аргументы выполнения
target.task.auth.method.name	keyword	Нет	Метод аутендификации
target.task.command	keyword	Нет	
target.task.description	text	Нет	Описание
target.task.name	keyword	Нет	Имя системного вызова
target.task.privileges.name	keyword	Нет	
target.task.privileges.original	keyword	Нет	
target.task.status.name	keyword	Нет	
target.task.status.original	keyword	Нет	
target.task.status.visibility.original	keyword	Нет	
target.task.status.working-directory	keyword	Нет	Рабочая директория
target.socket.port	port	Нет	Порт
target.threat.category	keyword	Нет	Категория угрозы, например: Potentially Unwanted Software
target.threat.confidence	keyword	Нет	Уровень доверия результату детектирования
target.threat.content-type	keyword	Нет	Content type угрозы: data, file, packet, url
target.threat.description	text	Нет	Описание угрозы
target.threat.detection_delta	integer	Нет	Окно реагирования
target.threat.origin.name	keyword	Нет	
target.threat.origin.original	keyword	Нет	
target.threat.severity	keyword	Нет	Уровень угрозы
target.threat.status.original	keyword	Нет	

Поле	Тип данных	Обязательно	Описание
target.threat.name	keyword	Нет	Наименование угрозы PUA:Win32/FustionCore
target.user.category	text	Нет	Категория пользователься
target.user.delegations	[keyword]	Нет	Windows: AllowedToDelegateTo
target.user.description	keyword	Нет	Описание пользователя
target.user.domain	keyword	Нет	Домен пользователя
target.user.group.id	keyword	Нет	ID группы пользователя
target.user.group.name	keyword	Нет	Имя группы пользователя
target.home.path.original	keyword	Нет	Путь к домашней дирректории
target.user.id	keyword	Нет	ID пользователя, например, SID или UID
target.user.id-history	[keyword]	Нет	Windows: SidHistory
target.user.name	keyword	Нет	Имя пользователя
target.user.primary-group	keyword	Нет	Windows: PrimaryGroupId
target.user.privileges.code	[keyword]	Нет	
target.user.privileges.description	[keyword]	Нет	
target.user.privileges.name	[keyword]	Нет	
target.user.privileges.original	[keyword]	Нет	
target.user.spn.delegators	[keyword]	Нет	
target.user.spn.names	[keyword]	Нет	
target.user.subcategory	text	Нет	
target.user.uac.attribute.new-value	keyword	Нет	
target.user.uac.attribute.old-value	keyword	Нет	
target.user.uac.status	[keyword]	Нет	

9. Описание специальных функций

В случае необходимости дополнительной обработки данных перед процессом нормализации можно воспользоваться функциями и операторами, позволяющими выполнять сложные операции прямо на странице настройки правила нормализации. Эти операции компилируются непосредственно в байт-коде Python.

Для корректного распознавания логического выражения используйте перенос | и описывайте выражение с новой строки.

По умолчанию все поля, которые указывают при работе с функциями и операторами в рамках дополнительной обработки данных, являются обязательными. Однако в поступающих данных указанные поля иногда могут не присутствовать. И чтобы не возникало ошибки, можно пометить эти поля как необязательные, отметив это в настройке правила нормализации или добавив в описание функции строку "**required: false**". В таком случае, поле будет обработано и выведено далее, если оно присутствует во входящих данных.

9.1. Строковые функции

9.1.1. Преобразование к нижнему регистру (lower)

Преобразование поля или определенной строки к нижнему регистру.

Использование функции: **lower**(**string**), где **string** — строка, преобразуемая к нижнему регистру.
Пример:

```
my_section.my_field:
  field: lower(hostname)
```

9.1.2. Преобразование к верхнему регистру (upper)

Преобразование поля или определенной строки к верхнему регистру.

Использование функции: **upper**(**string**), где **string** — строка, преобразуемая к верхнему регистру.
Пример:

```
my_section.my_field:
  field: upper(software_name)
```

9.1.3. Удаление элементов из строки (strip)

Функция убирает из строки все элементы, указанные перечислением в необязательном первом аргументе. Если указан только один (второй) аргумент, то будут удалены только пробелы.

Использование функции: **strip**("**strip_chars**", **string**), где **string** — строка, из которой необходимо убрать перечисленные символы **strip_chars**.
Пример использования функции для удаления пробелов:

```
section.stripped_field:
  field: strip(messy_string)
```

Пример использования функции для удаления запятой:

```
section.stripped_field_comma:
  field: strip(",", messy_comma_string)
```

Пример использования функции для удаления различных знаков препинания:

```
section.stripped_field_multiple_possible:
  field: strip(",\'.", messy_multiple_possible_string)
```

9.1.4. Разбиение строки (split)

Функция разделяет строку по указанному разделителю и возвращает её в виде списка.

Использование функции: **split**(**string**, **separator**)[**index**], где **string** — строка, которую необходимо преобразовать, **separator** — разделитель, а **index** — индекс требуемого элемента (допускается использование отрицательного индекса как в Python). [0], [1] и т.д. — первый, второй и т.д. элементы с начала строки, [-1] — первый элемент с конца строки.
Пример:

```
section.proto_name:
  field: split(http.protocol, '/')[0]
```

Пример использования функции для вывода элемента первого с конца:

```
section.other_proto_name:
  field: split(http.other_name, ',')[-1]
```

9.1.5. Проверка по регулярному выражению (match)

Функция возвращает **true**, если строка соответствует заданному регулярному выражению.

Использование функции: **match**(‘**regular_expression**’, **string**), где **string** — строка, которую необходимо проверить на соответствие, **regular_expression** — регулярное выражение.
Пример:

```
section.is_expected_code:
  required: false
  field: match('(1..|2..|418)', str(http.status))
```

Подробнее про **required: false** можно прочитать в начале раздела.

9.1.6. Замена строки (replace)

Функция выполняет замену в строке, возвращая новую строку с проведенной заменой.

Использование функции: **replace**(**string**, **old_value**, **new_value**), где **string** — строка, в которой необходимо произвести замену, **old_value** — заменяемое значение, **new_value** — новое значение.
Пример замены немецкого написания слова "benutzer" на английский "user":

```
section.user_info:
  field: replace(line.full_user_name, 'benutzer', 'user')
```

9.2. Логические операторы

Инфиксные операторы также доступны внутри нормализаторов YAML. В этом разделе доступны почти все операторы Python.

Логические операторы возвращают **true** или **false** в зависимости от выражения.

9.2.1. Логическое НЕ (not)

Оператор **not** возвращает **true**, если поле не соответствует заданному значению, иначе **false**.

Пример:

```
section.is_not_using_firefox:
  field: not browser_name == 'Firefox'
```

9.2.2. Равенство (==)

Оператор **==** возвращает **true**, если оба операнда равны, иначе **false**.

Пример:

```
section.is_using_firefox:
  field: software_name == 'Firefox'
```

9.2.3. Неравенство (!=)

Оператор **!=** возвращает **true**, если оба операнда различны, иначе **false**.

Пример:

```
section.is_not_1_3:
  field: version != 1.3
```

9.2.4. Больше (>)

Оператор **>** возвращает **true**, если один операнд больше другого, иначе **false**.

Пример:

```
section.is_newer_than_1_0:
  field: version > 1.0
```

9.2.5. Больше или равно (>=)

Оператор **>=** возвращает **true**, если один операнд больше или равен другому, иначе **false**.

Пример:

```
section.is_at_least_1_0:
  field: version >= 1.0
```

9.2.6. Меньше

Оператор **<** возвращает **true**, если один операнд меньше другого, иначе **false**.

Пример:

```
section.is_prior_to_1_0:
  field: version < 1.0
```

9.2.7. Меньше или равно

Оператор **<=** возвращает **true**, если один операнд меньше или равен другому, иначе **false**.

Пример:

```
section.is_prior_or_1_0:
  field: version <= 1.0
```

9.2.8. Логическое И (and)

Оператор **and** объединяет условия между собой. Если все выражения оцениваются как **true**, то возвращается **true**, если хотя бы одно — **false**, то возвращается **false**.

Использование оператора: **bool_expr_1 and bool_expr_2**, где **bool_expr** — логическое выражение. Допускается использование более двух логических выражений.
Пример:

```
section.is_firefox_and_windows:
  field: browser_name == 'Firefox' and os_name == 'Windows'
```

9.2.9. Логическое ИЛИ (or)

Оператор **or** возвращает значение **true**, если хотя бы одно из выражений оценивается как **true**, в ином случае — **false**.

Использование оператора: **bool_expr_1 or bool_expr_2**, где **bool_expr** — логическое выражение. Допускается использование более двух логических выражений.
Пример:

```
section.is_firefox_or_windows:
  field: browser_name == 'Firefox' or os_name == 'Windows'
```

9.2.10. Проверка наличия элемента (in)

Оператор **in** проверяет вхождение элемента в массив значений. Функция также работает для проверки вхождения подстроки в строку.

Использование оператора: **variable in (value_1, value_2, value_3)**, где **variable** — переменная, **value** — значение.
Пример:

```
section.is_firefox:
  field: |
     'Firefox' in http.user_agent
```

> **Важно!** Используйте перенос | для корректного распознавания логического выражения

9.3. Арифметические операторы

9.3.1. Умножение (*)

Оператор * умножает два операнда.

Пример:

```
section.total_cpu_freq:
  field: cpu_number * frequency
```

9.3.2. Деление (/)

Оператор **/** делит первый операнд на второй.

Пример:

```
section.division:
  field: first_value / second_value
```

9.3.3. Сложение (+)

Оператор **+** суммирует два операнда.

Пример:

```
section.sum:
  field: first_value + second_value
```

9.3.4. Вычитание (-)

Оператор **-** вычитает из первого операнда второй операнд.

Пример:

```
section.difference:
  field: first_value - second_value
```

9.4. Условные конструкции

9.4.1. cond

Функция **cond** работает как оператор **if/else**. Если указанное в первом аргументе логическое выражение оценивается как **true**, то выводится второй аргумент; если **false** - третий.

Использование функции: **cond**(**bool_expr**, '**Значение, если истина**', '**Значение, если ложь**'), где **bool_expr** — логическое выражение.
Пример:

```
section.browser_hint:
  field: |
    cond(browser_name == 'Firefox', 'Firefox detected',
         'Other browser detected')
```

> **Важно!** Используйте перенос | для корректного распознавания логического выражения

Функцию **cond** можно использовать как переключатель и описывать более сложные случаи.

Использование функции: **cond**(**bool_expr**, '**Значение, если истина**', **another_bool_expr**, '**Значение, если истина**', '**Значение по умолчанию**'), где **bool_expr** — логическое выражение.
Пример:

```
section.firewall_status: |
  cond(type == 'utm', 'Suspicious activity was detected',
       action == 'close', 'A connection was closed',
       action == 'start', 'A connection was started',
       'A connection was allowed')
```

> **Важно!** Используйте перенос | для корректного распознавания логического выражения

Пример без указания значения по умолчанию:

```
reason.type: |
  cond(action == 'reset', 'flow/reset',
       action == 'deny', 'flow/deny')
  required: false
```

> **Важно!** Используйте перенос | для корректного распознавания логического выражения

Если значение по умолчанию отсутствует, поле пропускается. В этом случае необходимо отметить в форме настройки правила нормализации рядом с соответствующим полем, что оно необязательное или добавить в поле ввода "**required: false**", иначе будет ошибка.
Если поле является необязательным, а условие ссылается на входную переменную, которая отсутствует, условие будет считаться ложным.
Если условие истинно, но в значении отсутствует поле ввода, это поле будет удалено из вывода.

Подробнее про используемый в примере **required: false** можно прочитать в начале раздела.

9.4.2. optional

Функция проверяет, присутствуют ли все указанные поля, если нет — возвращает значение по умолчанию (или **false**).

Пример:

```
outcome:
  field: |
    cond(optional(tcp.rst, false), 'failed',
         optional(tcp.fin, false), 'success',
         'pending')
```

> **Важно!** Используйте перенос | для корректного распознавания логического выражения

Если выражения относятся к нескольким полям, все они должны присутствовать. Следующий пример вернёт **NaN**, если a, b или c не присутствуют в проанализированных данных. "**NaN**" указывается, если данные отсутствуют, не существуют.
Пример:

```
sum:
  field: optional(a + b + c, float('nan'))
```

9.5. Поиск данных

Массивы, которые используются в нескольких нормализаторах, размещаются в **lookups.yaml**. Это специальный файл, содержащий только глобальные поисковые запросы, доступные в каждом нормализаторе.

Необходимо убедиться, что каждый массив имеет уникальное имя.

9.5.1. lookup {#lookup}

Функция **lookup** работает как поиск значений по ключу. Значения, содержащиеся в массивах, доступны только с помощью этой функции.
Допустим, в "Таблицах просмотра" определен следующий массив с названием "protos":

```
lookup:
  protos:
    0: NotSecureProtocol
    1: SecureProtocol
    2: VerySecureProtocol
    3: Telnet
```

Тогда есть возможность получить доступ к этим значениям следующим образом, где **protocol_id** является полем события:

```
section.field:
  field: lookup('protos', proto_id)
```

Если ключ не содержится в словаре, анализ завершится неудачей. Чтобы избежать этого, можно указать возвращаемое значение по умолчанию на случай, если ключ не найден.
В примере, если **proto_id** не является допустимым ключом в **protos**, будет возвращено значение "**Unknown protocol**":

```
section.field:
  field: lookup('protos', proto_id, 'Unknown protocol')
```

9.5.2. exists

Функция **exists** проверяет, имеет ли поле полезное значение: не null, не пустую строку и не "-".

Возвращает **true** или **false**.

Пример:

```
section.user_data.is_user_set:
  field: exists(line.app.data.user)
```

Пример использования функции **exists** в сочетании с функцией **cond**:

```
section.system.app_name:
  field: cond(exists(line.app.name), line.app.name, "unknown application")
```

9.6. Преобразование типа данных

9.6.1. Строковый формат (str)

Преобразование значения поля в строковый формат.

Использование функции: **variable::str**, где **variable** — переменная.
Пример:

```
section.field_that_is_a_string:
  field: field_that_is_a_int::str
```

9.6.2. Формат целого числа (int)

Преобразование значения поля в формат целого числа.

Использование функции: **variable::int**, где **variable** — переменная.
Пример:

```
section.field_that_is_a_int:
  field: field_that_is_a_string::int
```

9.6.3. Формат числа с плавающей точкой (float)

Преобразование значения поля в формат числа с плавающей точкой.

Использование функции: **variable::float**, где **variable** — переменная.
Пример:

```
section.field_that_is_a_float:
  field: field_that_is_a_int::float
```

9.7. Функции проверки корректного представления данных

9.7.1. Проверка IP-адреса (is_ip)

Функция для определения, является ли предоставленная строка допустимым адресом IPv4 или IPv6.

Пример:

```
section.is_valid_ip:
  field: is_ip(string)
```

9.7.2. Проверка имени хоста (is_hostname)

Функция для определения, является ли предоставленная строка допустимым именем хоста. Она не должна быть пустой и содержать точки.

Пример:

```
section.is_valid_hostname:
  field: is_hostname(string)
```

9.7.3. Проверка доменного имени (is_fqdn)

Функция для определения, является ли предоставленная строка корректным доменным именем. Доменное имя должно содержать хотя бы одну точку, метки между точками не должны быть пустыми. Это не должен быть IP-адрес. Доменное имя может заканчиваться точкой.

Пример:

```
section.is_valid_fqdn:
  field: is_fqdn(string)
```

9.8. Функции для работы со временными отметками

9.8.1. Приведение к ISO 8601 (parse_timestamp)

Функция выполняет перебор всех указанных в качестве аргументов форматов временной отметки и пытается разобрать строку **my_ts**. Функция перебирает форматы временной отметки до тех пор, пока метка времени не будет проанализирована и возвращена в виде строки в формате ISO 8601.

Форматы должны быть строковыми константами. Допустимые форматы: «iso8601» и все директивы синтаксического анализа, поддерживаемые функцией Python strptime.

Использование функции: **parse_timestamp**(**my_ts**, **format1**[, **format2**, **format3**...]), где **my_ts** — отметка времени, **format** — формат временной отметки.
Пример:

```
"@timestamp":
  field: parse_timestamp(
                        date + ' ' + time,
                        '%m/%d/%Y %I:%M:%S %p',
                        '%Y/%m/%d',
                        'iso8601'
                        )
```

> **Важно!** Синтаксический анализ временных меток с помощью функции **parse_timestamp** довольно медленный. Рекомендуется для создания временной метки ISO 8601 в первую очередь использовать простые строковые операции, и, только в случае невозможности этого, использовать функцию **parse_timestamp**.

9.8.2. Приведение к Unix time (timestamp_to_epoch)

Функция принимает временную метку ISO и преобразует ее в секунды, начиная с временной метки эпохи, в виде числа с плавающей точкой. Если в необработанной строке журнала присутствует **tzinfo** (информация о смещении времени от времени UTC, о переходе на летнее время и проч), то это значение будет использоваться для локализации отметки времени перед преобразованием.

Использование функции: **timestamp_to_epoch**(**my_ts**), где **my_ts** — отметка времени.
Пример:

```
section.since_epoch:
  field: timestamp_to_epoch(ts)
```

9.8.3. Приведение к UTC (epoch_to_timestamp)

Функция принимает временную метку эпохи в секундах и преобразует ее во временную метку UTC.

Использование функции: **epoch_to_timestamp**(**my_epoch**)
Пример:

```
section.date:
  field: epoch_to_timestamp(epoch)
```

9.9. Функции для дополнительной нормализации

9.9.1. Нормализация User Agent (normalize_http_user_agent)

Функция обращается к строке User agent и производит её дополнительный разбор по следующим полям:

- **full** — содержимое строки User Agent
- **name** — название браузера
- **os** — семейство и версия операционной системы
- **device** — устройство
- **major** — мажорная версия браузера
- **minor** — минорная версия браузера

Использование функции: **normalize_http_user_agent**(**string**), где **string** — строка, которую необходимо преобразовать.
Пример использования функции.
Событие:

```
{"src_ip":"10.10.10.10", "dst_ip":"20.20.20.20", "cs_user_agent":"Mozilla/5.0
(X11; Linux x86_64;rv:60.0) Gecko/20100101 Firefox/60.0"}
```

Нормализатор:

```
section.user_agent:
  field: normalize_http_user_agent(cs_user_agent)
```

Результат:

```
"section": {
"user-agent": {
"device": "Other",
"full": "Mozilla/5.0 (X11; Linux x86_64;rv:60.0) Gecko/20100101 Firefox/60.0",
"major": 60,
"minor": 0,
"name": "Firefox",
"os": "Linux"
}
```

9.9.2. Нормализация MAC-адреса (normalize_mac_address)

Функция имеет один обязательный аргумент (MAC-адрес) и необязательный — второй — аргумент. Второй аргумент имеет логический тип данных и по умолчанию **true**. Этот аргумент определяет поведение в случае неверного MAC-адреса (по умолчанию строка журнала отправляется в Index Error).

Поддерживаются следующие форматы:

- AA-BB-CC-DD-EE-FF
- AAA.BBB.CCC.DDD
- AAA:BBB:CCC:DDD
- AAA-BBB-CCC-DDD
- AAABBBCCCDDD

Если MAC-адрес действителен, функция преобразует его в стандартный формат **AA:BB:CC:DD:11:22**.
Например, MAC-адрес формата **FF-BA-CD-1D-32-11** функция преобразует в формат **FF:BA:CD:1D:32:11**.

Если MAC-адрес недействителен, а второй аргумент **true** (по умолчанию), строка будет отправлена в Index Error. Если второй аргумент **False**, то будет возвращена пустая строка, а для события **event.anomaly.malformed_mac_address** будет задана нормализованная строка журнала.

Использование функции: **normalize_mac_address**(**mac_address**)

Пример обработки события с действительным MAC-адресом и без указания второго аргумента.
Событие:

```
{"src_ip":"10.10.10.10", "mac_address":"AA-BB-CC-DD-EE-FF"}
```

Нормализатор:

```
section.client_mac:
  field: normalize_mac_address(mac_address)
```

Результат:

```
"section": {
   "client_mac": "AA:BB:CC:DD:EE:FF"
}
```

Пример обработки события с недействительным MAC-адресом и **false** в качестве второго аргумента.

```
section.client_mac:
  field: normalize_mac_address("AA:BB:CC", false)
```

Результат:

```
{
    "event": {
        "anomaly": {
            "malformed_mac_address": [
                "AA:BB:CC"
            ]
        }
    },
    "section": {
        "client_mac": ""
    }
}
```

9.9.3. Нормализация данных по хосту (normalize_host)

Функция предназначена для корректного формирования информации о хосте. Принимает на вход ряд полей и возвращает в виде словаря с тремя ключами: **IP**, **FQDN**, **Hostname**, где **IP** — массив IP-адресов, **FQDN** — массив доменных имен, **Hostname** — массив имен хостов.

Использование функции: **normalize_host**(**field1** [, **field2**, **field3**, ... , **fieldN**]), где **field** — поле. Пример:

```
target.host:
  field: normalize_host('127.0.0.1','lt-mail','lt-mail.domain','','10.0.0.2')
```

Результат:

```
"target": {
"host": {
"fqdn": ["lt-mail.domain"],
"hostname": ["lt-mail"],
"ip": ["10.0.0.2","127.0.0.1"]
}
}
```

9.9.4. Нормализация данных URL (normalize_url)

Функция разбивает URL-адрес на составляющие и возвращает в виде словаря. Второй аргумент является необязательным, в случае его отсутствия значением по умолчанию является пустая строка.

Пример использования функции:

```
url:
  field: normalize_url(field, type)
```

Пример результата:

```
"url": {
    'protocol': 'http',
    'host': {'hostname': ['pangeoradar.ru'], 'ip': [], 'fqdn': []},
    'path': '/',
    'params': '',
    'username': '',
    'password': '',
    'port': 80,
    'query': '',
    'fragment': '',
    'original': 'https://pangeoradar.ru/',
    'source-type': 'something',
}
```

Где:

- **protocol** — протокол
- **host** — структура {'hostname': [], 'ip': [], 'fqdn': []}, в которую передается Hostname, IP или FQDN
- **path** — путь
- **params** — параметры
- **username** — имя пользователя
- **password** — пароль
- **port** — порт
- **query** — запрос
- **fragment** — фрагмент страницы
- **original** — оригинальный URL, переданный в функцию
- **source-type** — тип источника

Событие:

```
{"src_ip":"10.10.10.10", "url":"http://user:pass@NetLoc:80/path;parameters?
query=argument#fragment"}
```

Нормализатор:

```
field: normalize_url(data['url'], 'url')
```

Результат:

```
"target": {
"http": {
"url": {
"fragment": "fragment",
"host": {"fqdn": [], "hostname": ["netloc"], "ip": []},
"original": "http://user:pass@NetLoc:80/path;parameters?
query=argument#fragment",
```

```
"params": "parameters",
"password": "pass",
"path": "/path",
"port": 80,
"protocol": "http",
"query": "query=argument",
"source-type": "",
"username": "user"
}
}
}
```

9.9.5. Нормализация данных Windows SID (normalize_windows_sid)

Функция принимает одно поле (Windows SID) в качестве входных данных и возвращает словарь с тремя ключами: **category**, **subcategory** и **desc**, где **category** — категория, **subcategory** — подкатегория и **desc** — описание.

Пример использования функции:

```
initiator.user.id_details:
    field: normalize_windows_sid(SubjectUserSid)

target.user.id_details:
    field: normalize_windows_sid(TargetUserSid)
```

Пример результата:

```
"initiator" : {
    "user" : {
        "id_details" : {
            "category" : "builtin_system_account",
            "subcategory" : "builtin_anonymous_account",
            "desc" : "ANONYMOUS LOGON"
        }
    }
}
```

Перед использованием этой функции в "Таблицах просмотра" необходимо описать массив «windows_sids». Он должен предоставить записи для случаев:

1. **sid** равен строке,
2. **sid** начинается с подстроки,
3. **sid** начинается с подстроки и заканчивается другой подстрокой,
4. **sid** начинается с подстроки и не заканчивается другой подстрокой.

Пример lookup, который охватывает все 4 варианта случаев. Будет взято первое совпадение:

```
lookup:
    windows_sids:
        - "sid": "S-1-5-7"
          "match_type": equal
```

```
          "category": builtin_system_account
          "subcategory": builtin_anonymous_account
          "desc": ANONYMOUS LOGON
        - "sid": "S-1-5-111-"
          "match_type": start
          "category": builtin_system_account
          "subcategory": builtin_virtual_sshd_account
          "desc": TBD
        - "sid": "S-1-5-21-"
          "match_type": start_end
          "ends":
            - "end": "-500"
              "category": standard_account
              "subcategory": builtin_virtual_sshd_account
              "desc": TBD
            - "end": "-501"
              "category": standard_account
              "subcategory": builtin_guest_account
              "desc": TBD
            - "not_end": "$"
              "category": standard_account
              "subcategory": standard_account
              "desc": TBD
```

Пример результата, если lookup без записей:

```
"initiator" : {
    "user" : {
        "id_details" : {
            "category" : "undefined_account_type",
            "subcategory" : "undefined_account_type",
            "desc" : "undefined_account_type"
        }
    }
}
```

9.10. Дополнительные функции

9.10.1. Tapping

Функция, которая помогает обрабатывать сложные непрогнозируемые данные на этапе предварительной обработки.

Пример использования функции:

```
tap: |
    tcp_flags = line.parsed['tcp']
    line.parsed['flow_tags'] = [
        f"tcp_{flag}"
        for flag in
            ("syn", "fin", "rst", "psh", "ack", "cwr", "ecn", "urg")
        if tcp_flags.get(flag, False)
    ]
```

10. Обогащение событий

В качестве источников обогащения событий в Платформе используются следующие типы обогащений:

- GeoIP
- DNS
- Threat Intelligence
- RVS
- Lookups

10.1. Настройка GeoIP обогащения

GeoIP обогащение работает на основе базы IP-адресов GeoLite от MaxMind’s.

1. Для работы необходимо получить базу GeoLite2-City.mmdb и положить ее на экземпляр модуля обработки событий.(https://dev.maxmind.com/geoip/geolite2-free-geolocation-data?lang=en)
2. Далее в конфигурации Termite в разделе Кластер - Управление конфигурацией - Termite - GeoIP (см. рисунок 150) включить GeoIP (True) и указать путь к файлу GeoIP обогащения:

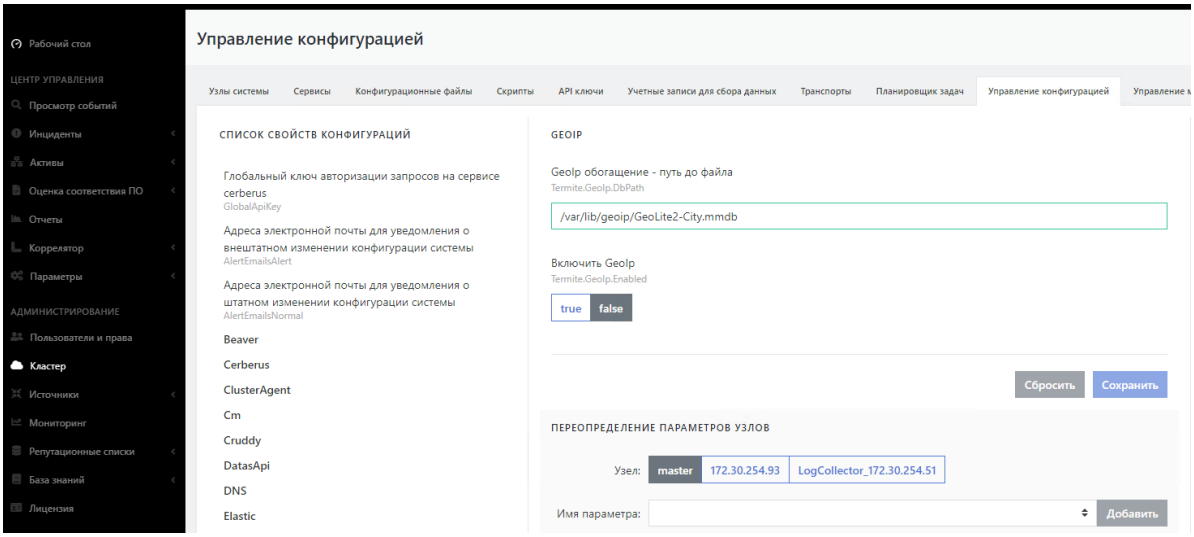

Рисунок 150 - Настройка GeoIP обогащения

3. Далее необходимо перезапустить сервис **pangeoradar-termite**.

В результате, события должны обогащаться GeoIP информацией, как изображено на рисунке 151.

initiator.host.geoip.city	[null]
initiator.host.geoip.continent	["North America"]
initiator.host.geoip.country	["United States"]
initiator.host.geoip.iso	["US"]
initiator.host.geoip.key	["66.249.64.137"]
initiator.host.geoip.location	[[-97.822, 37.751]]
initiator.host.geoip.timezone	["America/Chicago"]
initiator.host.hostname	[]
initiator.host.internal	false
initiator.host.ip	["66.249.64.137"]
initiator.network.node.host.hostname	[]
initiator.process.id	13353
initiator.socket.port	22758

Рисунок 151 - Обогащенное GeoIP событие

10.2. Настройка DNS обогащения

DNS обогащение может работать как от .csv файла с базой FQDN и IP-адресов, так и получая информацию от DNS сервера. Можно использовать оба способа единовременно.

10.2.1. DNS обогащение по сети

Для организации работы DNS обогащения в конфигурации Termite в разделе Кластер - Управление конфигурацией - Termite - DNS (см. рисунок 152) настроить перечень доменов и включить обогащение DNS (True). Для обогащения в локальной сети включить DNS локально (True) и указать перечень локальных сетей и DNS серверов.

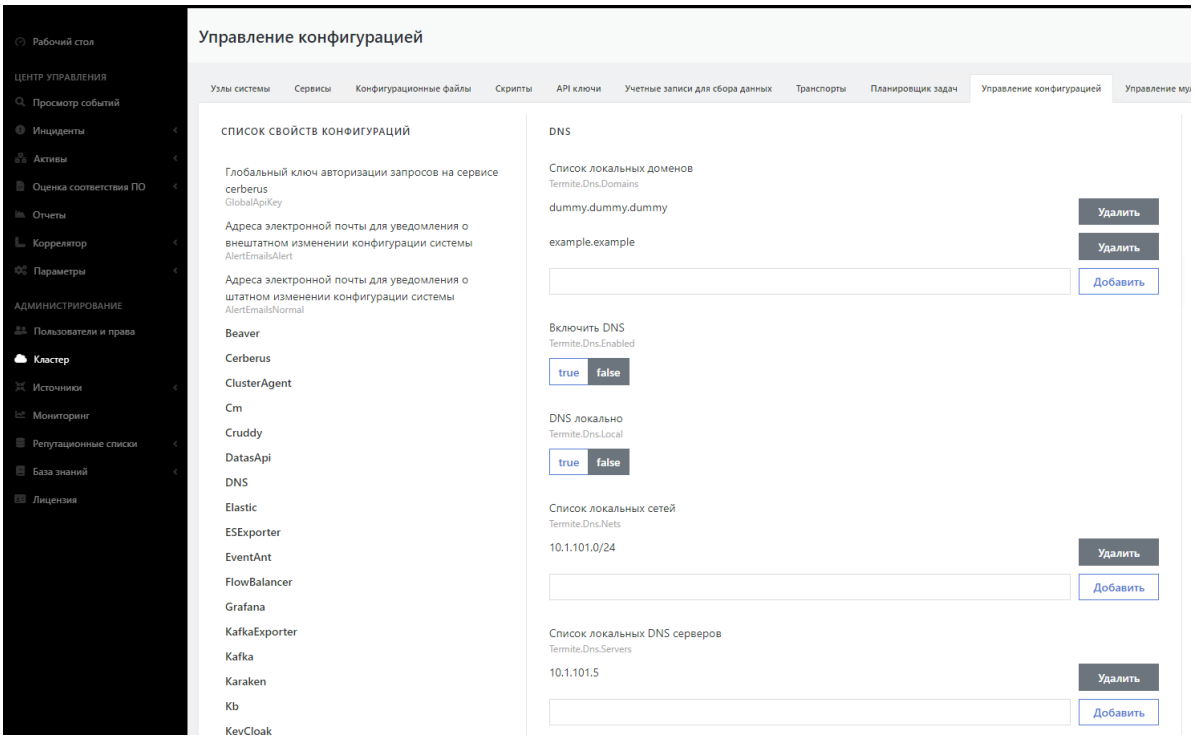

Рисунок 152 - Настройка DNS обогащения

В результате, события должны обогащаться DNS информацией, как изображено на рисунке 153.

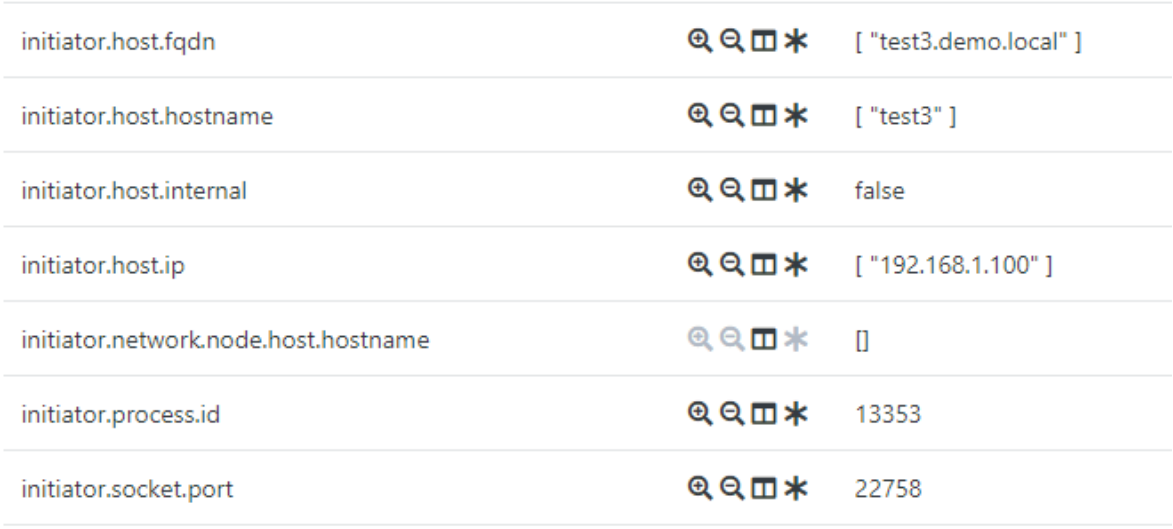

Рисунок 153 - Обогащенное DNS событие

10.3. Настройка Threat Intelligence обогащения

Threat Intelligence обогащение работает на основе баз угроз безопасности, получаемых Платформой различных поставщиков.

Для просмотра базы Threat Intelligence необходимо в интерфейсе Платформы перейти в раздел “Репутационные списки”. Раздел изображен на рисунке 154.

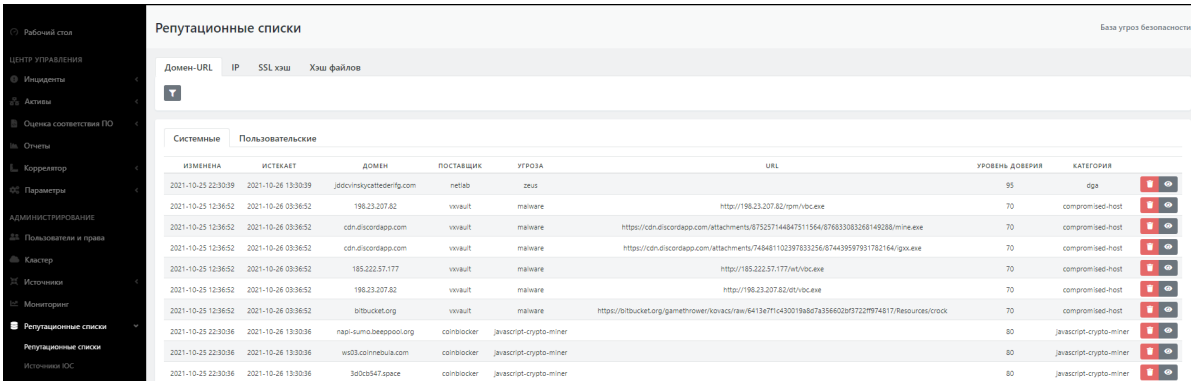

Рисунок 154 - Репутационные списки

TI обогащение позволяет наполнять дополнительной информацией события, содержащие: Домен-URL, IP - адрес, SSL хэш, Хэш файлов из базы угроз.

Для работы TI - обогащения необходимо в конфигурации Termite в разделе Кластер - Управление конфигурацией - Termite - Threatintel (см. рисунок 155) включить обогащение Threatintel (True) и указать путь к файлу Treatintel.

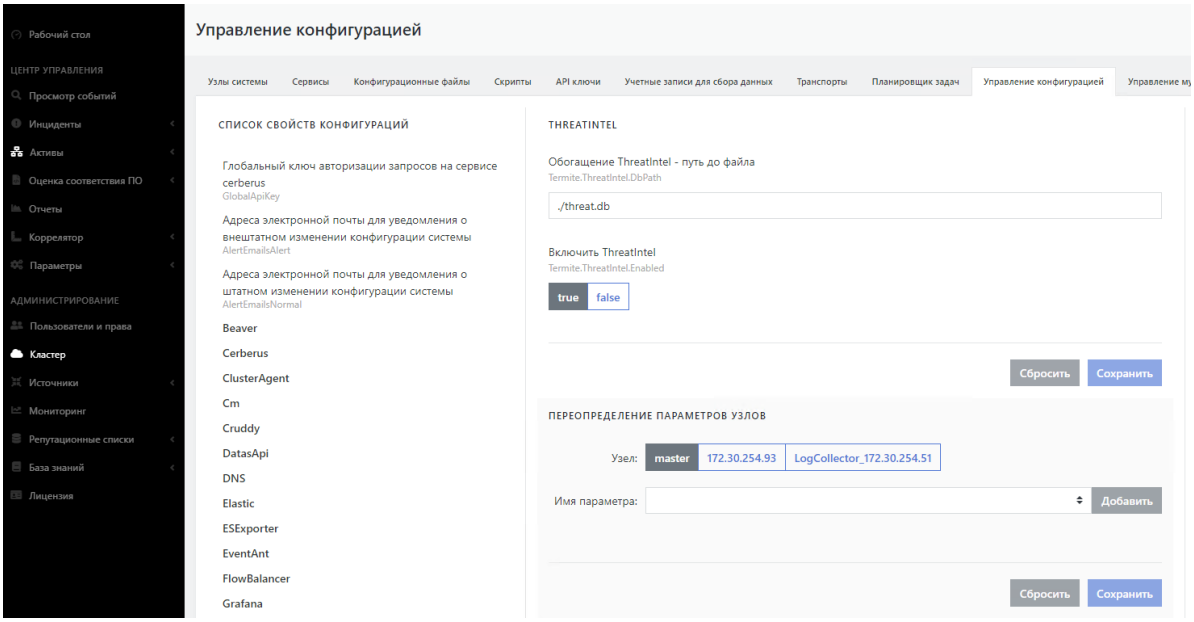

Рисунок 155 - Настройка Threatintel обогащения

В результате, события должны обогащаться TI информацией, как изображено на рисунке 156.

initiator.blacklist.ip.category	["compromised-host"]
initiator.blacklist.ip.confidence	[33]
initiator.blacklist.ip.ip	["119.236.128.231"]
initiator.blacklist.ip.port	[null]
initiator.blacklist.ip.protocol	["tcp"]
initiator.blacklist.ip.provider	["alienvault"]
initiator.blacklist.ip.threat	["compromised-host"]
initiator.host.fqdn	[]
initiator.host.hostname	[]
initiator.host.internal	false
initiator.host.ip	["119.236.128.231"]
initiator.network.node.host.hostname	[]
initiator.process.id	13353
initiator.socket.port	22758

Рисунок 156 - Обогащенное TI событие

10.4. Настройка RVS обогащения

RVS обогащение работает на основе табличных списков.

1. Для настройки RVS обогащения необходимо в табличном списке создать коллекцию (вручную или специальными средствами для обогащения).

   Работа с интерфейсом табличных списков представлена в *руководстве по работе с RVS (табличные списки)* ;

2. Далее, в созданной коллекции, необходимо добавить документ, пример которого изображен на рисунке 157.

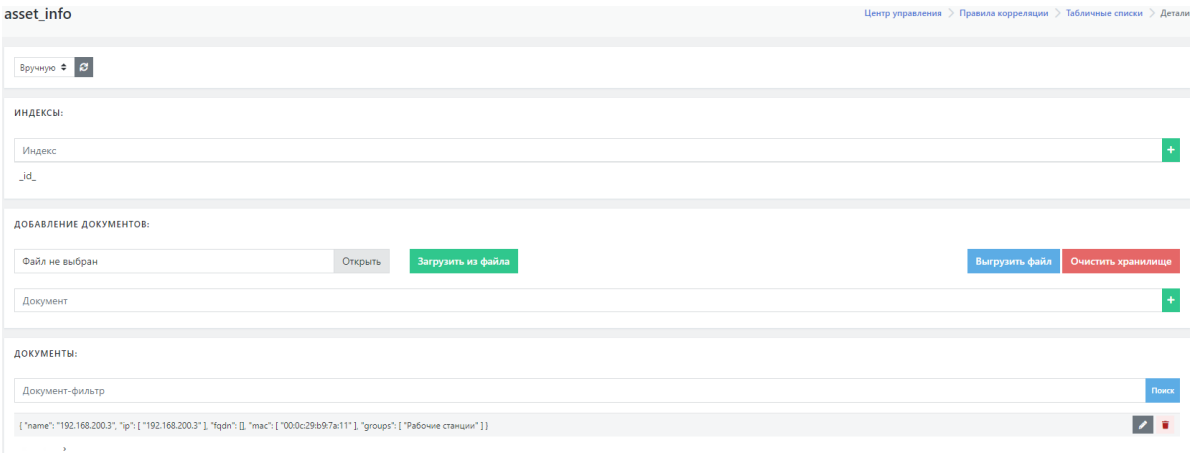

Рисунок 157 - Табличные списки

```
Созданный документ в json формате:

```

{

```
"name": "192.168.200.3",
"ip": [
    "192.168.200.3"
],
"fqdn": [],
"mac": [
    "00:0c:29:b9:7a:11"
],
"groups": [
    "Рабочие станции"
]
}
```

3. В конфигурации Termite в разделе Кластер - Управление конфигурацией - Termite - RVS (см. рисунок 158) включить обогащение RVS (True).

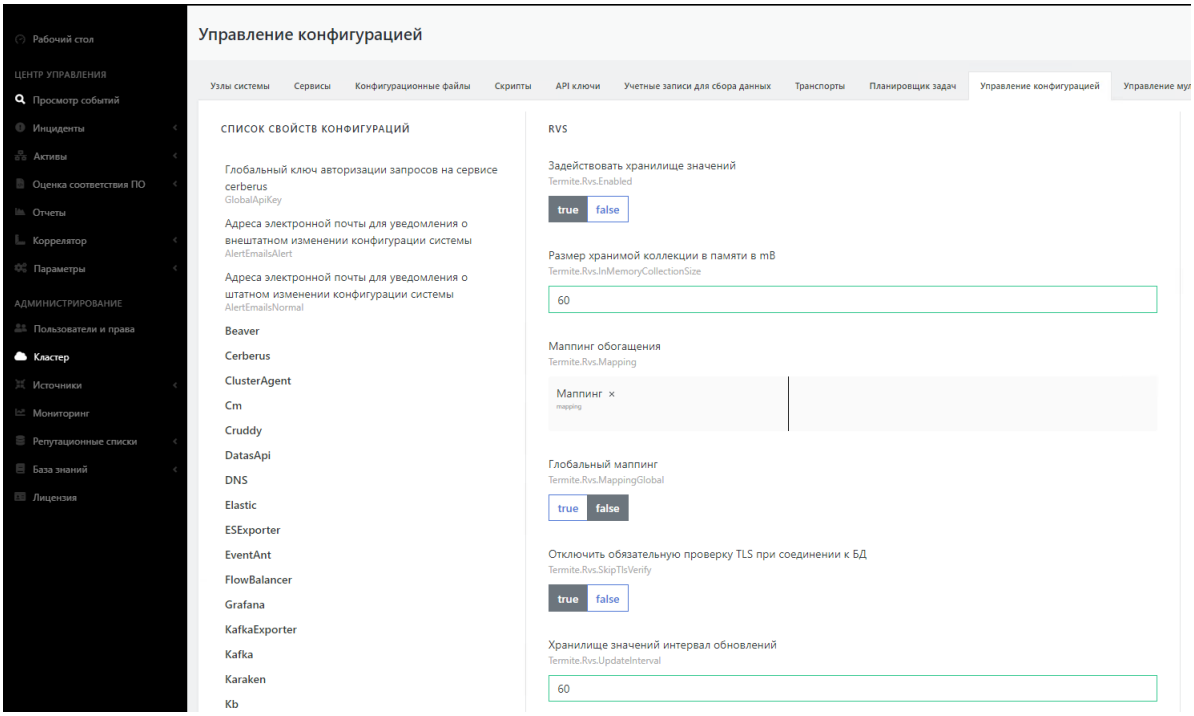

Рисунок 158 - Настройка RVS обогащения

```
Для обогащения заполнить поля:

- Задействовать хранилище значений: true
- маппинг (пример):

```
2162-Cisco-NetFlow: # название источника (event.logsource.input)
    asset_info: # название коллекции
        - enrich_from: # настройка проверки совпадения полей коллекции и
нормализованного события
            collection_field: ip # поле в коллекции, сопоставляемое с полем в
нормализованном событии
            normalized_field: initiator.host.ip # поле в нормализованном
событии, сопоставляемое с полем коллекции
        - enrich_to: # настройка обогащения полей из коллекции в поля
нормализации
```
```

```
            collection_field: mac # содержимое поля в коллекции
            normalized_field: initiator.interface.mac # целевое поле в
нормализованном событии
```

- глобальный маппинг: false
- Отключить обязательную проверку TLS при соединении к БД: true
- Хранилище значений интервал обновлений: 60
- Использовать TLS шифрование: true

Результат обогащенного табличным списком события представлен на рисунке 159.

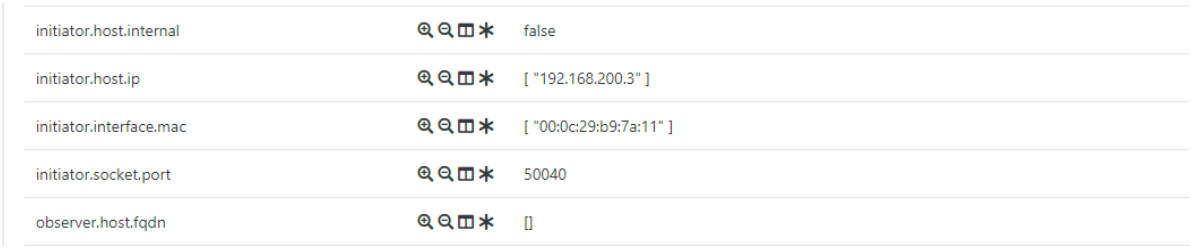

Рисунок 159 - Обогащенное RVS событие

10.5. Lookup обогащение

Lookup обогащение происходит на этапе нормализации событий.

Описание Lookup представлено в разделе *Специальные функции для работы с полями нормализации*.

11. Фильтрация событий

Платформа поддерживает фильтрацию входящих событий на двух этапах:

1. Фильтрация на этапе сбора лог-коллектором;
2. Фильтрация на этапе принятия событий модулем обработки событий.

11.1. Фильтрация на этапе сбора лог-коллектором

Фильтрацию на этапе сбора лог-коллектором можно разбить на два типа:

11.1.1. Фильтрация структурированных данных

Применима для таких типов источников, как:

- WMI
- Event log
- ODBC
- ETW

Фильтрация в компонентах сбора со структурированными данными работает как blacklist и настраивается при описании источника, в секции filters.

Пример фильтрации, исключающий сбор событий с уровнем Information:

```
filters:
  created: ''
  event_id: ''
```

```
  qualifiers: ''
  record_id: ''
  process_id: ''
  thread_id: ''
  version: ''
  computer_name: ''
  msg: ''
  level_text: 'Information'
  task_text: ''
  opcode_text: ''
  channel_text: ''
  provider_text: ''
```

11.1.2. Фильтрация неструктурированных данных

Фильтры можно указать для каждого коллектора с неструктурированными данными.

Фильтры содержат белый список (whitelist) и черный список (blacklist) с массивом регулярных выражений.

Все регулярные выражения проверяются перед запуском приложения. Сначала проверяется белый список, а затем черный.

Фильтрация по регулярным выражениям может быть включена в любом коллекторе с неструктурированными данными.

Включается путем добавления секции filters. В данной секции указываются два массива — whitelist, blacklist.

Все события сначала проходят фильтры указанные в whitelist, т.к. его приоритет выше. Затем события проверяются фильтрами, указанными в blacklist.

Whitelist — события, которые соответствуют регулярному выражению, попадают в очередь на отправку.

Blacklist — события, которые соответствуют регулярному выражению, блокируются и не попадают в очередь на отправку.

Пример фильтрации для неструктурированных данных:

```
filters:
    whitelist:
      - "^localhost.*$"
    blacklist:
      - "^.*[0-9]$"
```

11.2. Фильтрация на этапе принятия событий модулем обработки событий

Фильтрация на данном этапе осуществляется через файл `/etc/termite/processors.py`

Блок предварительной фильтрации - это простая функция Python, которая принимает сырое событие и определенное условие, в качестве входных данных, и возвращает логическое значение.

Значение False приводит к пропуску сырого события.

Пример использования, по которому, если в сыром событии присутствует слово "irrelevant" - событие пропускатеся:

```
@prefilter('drop-irrelevant')
def drop_irrelevant(raw):
    return 'irrelevant' not in raw
```

11.3. Настройка фильтрации поступающих событий

При получении сообщений от внешних источников можно настроить фильтрацию поступающих событий.

Для настройки фильтрации необходимо выполнить следующие действия:

1. Подключиться по SSH к узлу обработки событий (Worker) платформы.
2. Открыть на редактирование файл:

   `nano /opt/pangeoradar/termite2/venv/lib/python3.7/site-packages/termite_spu/prefilters/prefilters.py`
3. Добавить в конец файла следующую запись:

   `@prefilter('<имя>')`

   ```
   `def drop_demo_user(raw: str):`

   `return '"rrname":"<ключ>"' not in raw`
   ```

   где:

   - <*имя*> - уникальное имя фильтра.
   - <*ключ*> - строка из состава сырых данных события, по наличию которой будет происходить фильтрация поступающих событий.
4. Сохранить изменения в файле.
5. Открыть файл pipelines.yaml:

   `nano /opt/pangeoradar/configs/termite/pipelines.yaml`
6. Найти в файле раздел настроек источника, чьи поступающие сообщения необходимо фильтровать, и добавить в этот раздел следующую запись:

   `prefilters: ['<имя>']`,

   где: <*имя*> - имя фильтра, заданное на шаге 3.
7. Сохранить изменения в файле.

Для проверки работы фильтрации поступающих сообщений - перейти в веб-интерфейс платформы в раздел **"Просмотр событий"** и убедиться, что указанные для фильтрации события не поступают в платформу.

12. Агрегация событий

12.1. Настройка агрегации событий

В Платформе предусмотрена возможность агрегации событий. Настроить агрегацию можно либо при создании нового источника событий, либо при редактировании параметров ранее заведенного в Платформе источника.

Для настройки агрегации необходимо выполнить следующие действия (на примере редактирования):

1. В веб-интерфейсе Платформы зайти в подраздел "Источники" -> "Управление источниками" -> вкладка "Источники".
2. В текущем списке источников событий выбрать интересующий и нажать кнопку редактирования для данного источника  Кнопка для редактирования.
3. В форме редактирования источника в области параметров **Правила для termite** для параметра "**Агрегация**" выбрать в раскрывающемся списке одно или последовательно несколько полей, которые не должны меняться, и по которым будет происходить агрегация событий на данном источнике (см. рисунок 160). Расшифровка полей для агрегации дана в разделе «Описание полей нормализации».
4. Для сохранения введенных изменений нажать кнопку **Сохранить**.
5. Синхронизировать источники, нажав кнопку **Синхронизировать** на вкладке "Источники".

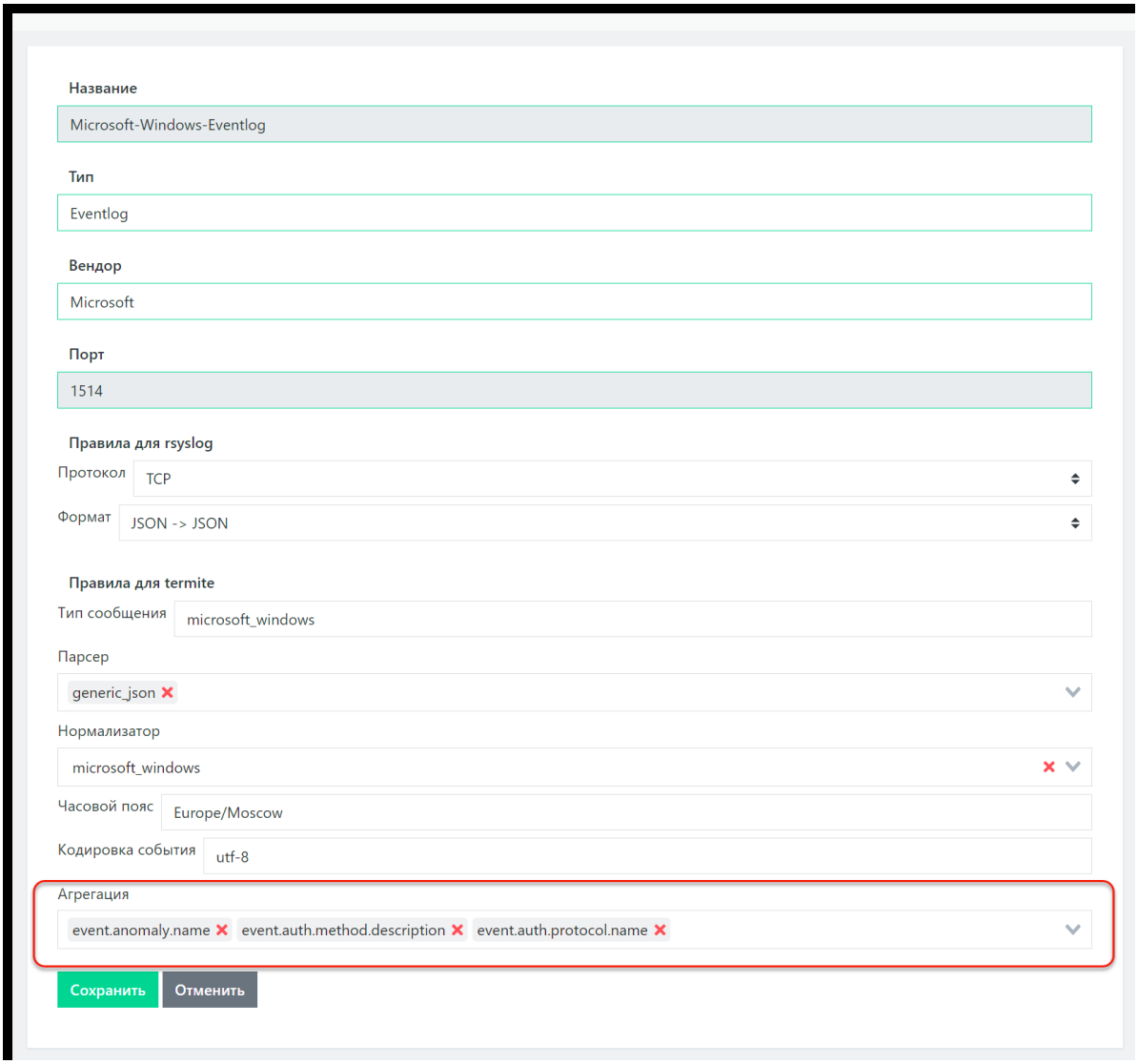

Рисунок 160 - Настройка агрегации событий

12.2. Просмотр результатов агрегации событий

Для просмотра результатов агрегации событий необходимо выполнить следующие действия:

1. В веб-интерфейсе Платформы зайти в раздел "Просмотр событий".
2. Задать временной интервал в поле **Время**.
3. Ввести или выбрать в раскрывающемся списке нужный индекс в поле **Индекс**.
4. Обновить данные на экране согласно заданным параметрам нажав Обновить данные.
5. В левой части экрана в области "**Доступные поля**" найти поле **grouped_occur_count** и нажать Добавление параметра.

Поле **grouped_occur_count** добавится в область "**Выбранные поля**". Колонка поля **grouped_occur_count** добавится в табличный список событий (см. рисунок 161). В данной колонке отображается количество агрегированных событий.

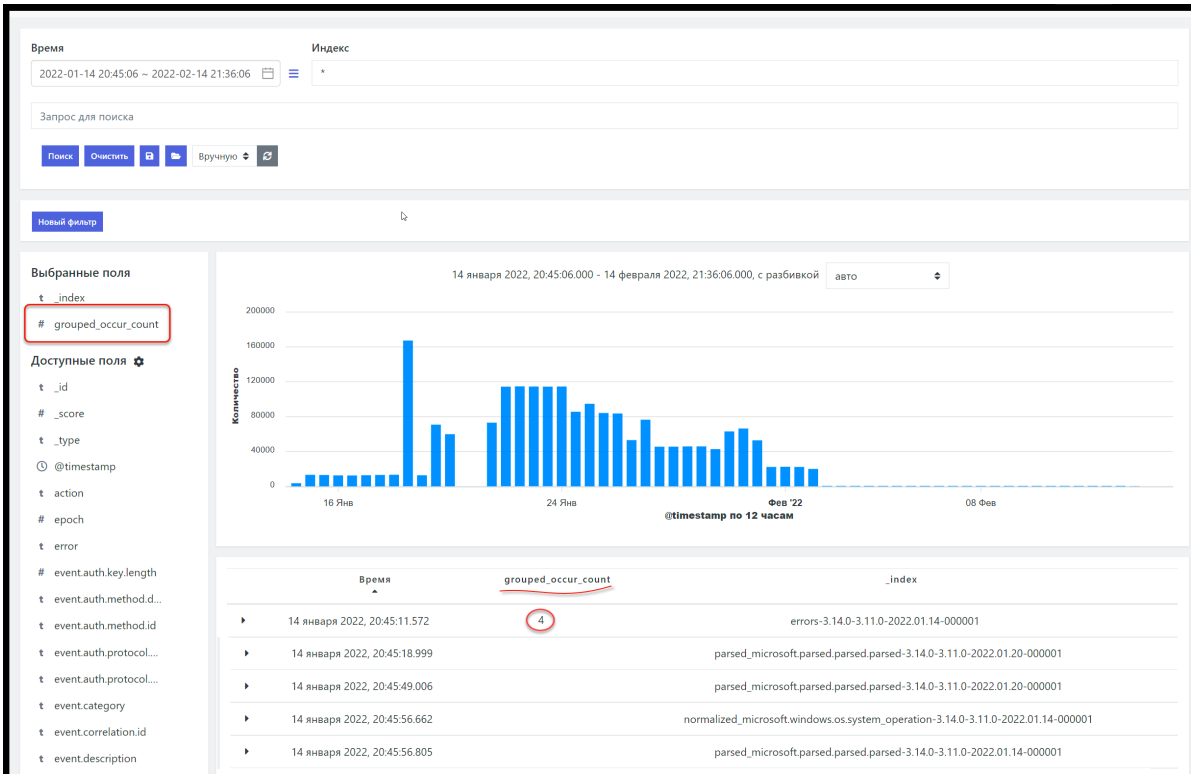

Рисунок 161 - Отображение агрегированных событий в табличном списке событий в разделе "Просмотр событий"

13. Руководство по настройке лог-коллектора. Активные источники событий

13.1. Радар лог-коллектор. Описание.

Радар лог-коллектор (RADAR LOG-COLLECTOR), далее лог-коллектор, предназначен для организации активного сбора событий от активов, не имеющих возможности отправки данных в сторонние системы. Лог-коллектор позволяет организовать различные схемы сбора событий от любых активов, участвующих в сетевом взаимодействии, создающих журналы событий.

Основные функции:

- сбор, локально и удалённо по различным протоколам;
- отправка событий в другие системы;
- обработка событий перед отправкой;

- пересылка событий в зашифрованном виде и со сжатием;
- отправка по расписанию;
- накопление событий при разрыве соединения и отправка после восстановления.

Поддерживаемые операционные системы следующих семейств:

- Windows Vista, Windows 7, Windows 8, Windows 10, Windows 11;
- Windows Server 2008, Windows Server 2011, Windows Server 2016, Windows Server 2019, Windows Server 2022;
- Linux Debian;
- Linux CentOS;
- Linux RedHat.

Поддерживаемые способы и протоколы отправки:

- TCP;
- SSL/TLS;
- UDP;
- Kafka;
- Запись в локальный файл.

Сбор событий

1. Локальный:
    - Event Tracing for Windows (ETW);
    - File Read;
    - Windows Event Log;
    - Результаты работы скрипта (Python/CMD/PowerShell/Bash/Perl).
2. Удалённый:
    - Windows Event Log via RPC;
    - WMI;
    - ODBC;
    - File via SMB;
    - File via SSH;
    - File via FTP;
    - File via SFTP;
    - File via HTTP(S);
    - Checkpoint OPSEC LEA.
3. Пассивный:
    - TCP;
    - UDP;
    - Netflow v5, v9;
    - Syslog;
    - SNMP Trap;
    - HTTP.

13.2. Основные характеристики

Программное обеспечение **лог-коллектор** обеспечивает решение перечисленных ниже основных задач:

- сбор/прием событий;
- обработка событий;

- отправка событий;
- временное хранение событий.

Сбор/прием *событий* может осуществляться в любом из трех режимов:

- **Локальный**. Лог-коллектор устанавливается в системе в виде агента, производит чтение файлов etw, eventlog, wmi, получение результатов выполнения скриптов (bash, perl, python, PowerShell) и их отправку в **Платформу Радар** (либо в промежуточный агент).
- **Пассивный**. Лог-коллектор осуществляет прием *событий* от систем, которые могут самостоятельно отправлять данные.
- **Удаленный/активный.** Лог-коллектор устанавливается на выделенный сервер и осуществляет удаленный сбор *событий* по различным протоколам. Также может быть установлен на конечном *источнике событий*, и осуществлять сбор не только с этого *источника событий*, но и с других систем. В этом случае необходимо предусмотреть дополнительные ресурсы для работы лог-коллектора.

Обработка событий позволяет обогатить события дополнительной технической информацией, изменить формат и кодировку перед их отправкой.

Отправка событий может осуществляться посредством их одновременной передачи в **Платформу Радар** и несколько внешних систем в зашифрованном виде со сжатием.
При отправке событий возможна маркировка событий коллектором. Данная маркировка может быть использована для определения, от какого коллектора (или территориально распределенной площадки) поступило событие.

Временное хранение событий предотвращает потерю *событий* при разрыве соединения, накапливая *события* для их последующей отправки после восстановления соединения.

С учётной записью/без учётной записи. Лог-коллектор имеет возможность сбора событий как с использованием указанной служебной учетной записи для доступа к событиям, так и без учётной записи (для транспортов, где это технически возможно).

13.3. Архитектура

Лог-коллектор имеет компонентную архитектуру и включает в себя следующие программные *компоненты*:

- **Контроллер** (controller)— осуществляет управление всеми компонентами лог-коллектора
- **Компонент сбора метрик** (metric_server)— осуществляет сбор статистики по работе лог-коллектора
- **Компонент API управления** (api_server)— предоставляет возможность удаленного управления лог-коллектором и мониторинга
- **Компонент журналирования** (journal)— осуществляет ведение журнала работы лог-коллектора
- **Компоненты сбора/приема событий** (inputs) — осуществляют сбор событий
- **Компоненты отправки событий** (outputs) — осуществляют оправку событий

Управление настройками лог-коллектора и его *компонентов* может осуществляться как через основной конфигурационный файл на сервере, где установлен экземпляр лог-коллектора, так и централизовано, из веб-интерфейса **Платформы Радар** Управление лог-коллектором.

13.4. Установка лог-коллектора

Архив файла установки лог-коллектора находится в личном кабинете заказчика на сайте portal.pangeoradar.ru

13.4.1. Требования к техническому и программному обеспечению

Минимальные требования к ресурсам:

- 4 Core
- 4 GB RAM
- 60 GB HDD

Минимальные требования к ресурсам при установке в системе с настроенным сервисом Windows Event Collector:

- 4 Core
- 8 GB RAM
- 500 GB HDD

> **Важно!** Требования к объему дискового пространства представлены без учета промежуточного хранения.

Для нормального функционирования лог-коллектора требуется установка на выделенные ресурсы одной из нижеперечисленных операционных систем, являющихся средой функционирования лог-коллектора:

- Windows Vista, Windows 10, Windows 11;
- Windows Server 2008, Windows Server 2011, Windows Server 2016, Windows Server 2019, Windows Server 2022;
- Linux Debian;
- Linux CentOS;
- Linux RedHat.

На приведенных выше минимальных требованиях к ресурсам лог-коллектор обеспечивает обработку потока 5000 событий в секунду.

13.4.2. Возможные схемы развертывания

- Установка на источнике для организации локального сбора *событий* с последующей передачей в **Платформу Радар** или в промежуточный лог-коллектор.
- Установка на выделенный сервер для организации удаленного сбора и пересылки *событий*.
- Установка цепочки лог-коллекторов для передачи *событий* в зашифрованном виде.

13.4.3. Установка лог-коллектора на различных ОС

> **Важно!** Установка лог-коллектора осуществляется под учетной записью с правами администратора.

Перед установкой необходимо скопировать на целевую систему архив с дистрибутивом, защищенный паролем. Пароль передается отдельно от архива.

13.4.3.1. Установка в ОС Windows

Разархивировать папку с дистрибутивом в корневой раздел диска C. При выполнении распаковки нужно будет ввести пароль от архива.

После распаковки в папке должны быть следующие файлы:

- log-collector.exe (дистрибутив лог-коллектора)

- config.yaml (файл конфигурации с минимально необходимыми для подключения настройками)
- example.yaml (пример общего файла конфигурации)

Необходимо запустить терминал от имени администратора и перейти в раздел, куда предварительно была распакована папка с дистрибутивом лог-коллектора.

```
cd c:\log-collector
```

Выполнить установку с помощью команды:

```
log-collector-<версия релиза>-<тип архитектур>.exe winsvc
```

Данная команда установит лог-коллектор в качестве сервиса ОС.

Запуск сервиса

```
net start PangeoRadarLogCollector
```

Остановка сервиса

```
net stop PangeoRadarLogCollector
```

Также можно выполнить запуск/остановку/перезапуск сервиса из оснастки «Сервисы»

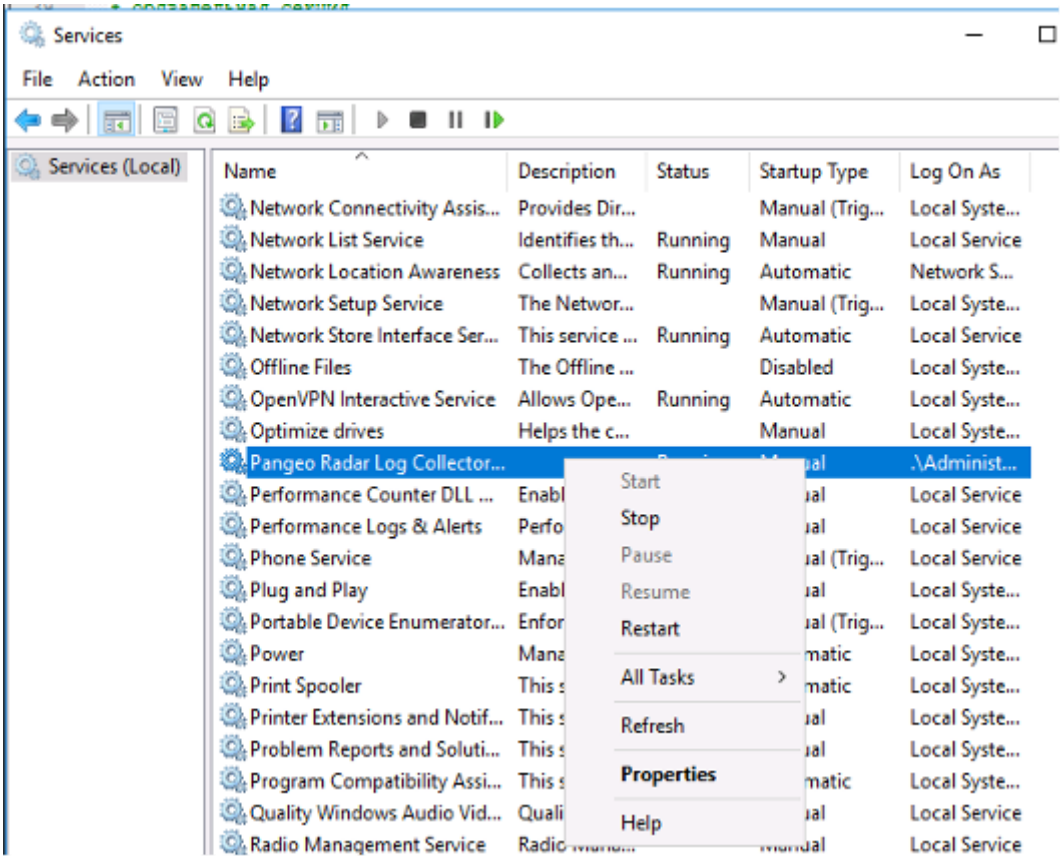

Рисунок 162 - Управление службой лог-коллектор в оснастке «Сервисы»

Если при запуске/перезапуске сервиса выводится ошибка, причину следует определять просмотром записей в журнале неудачных запусков лог-коллектора **logcollector.crash.log** в папке, из которой была выполнена установка сервиса.

Конфигурационный файл лог-коллектора **config.yaml** находится в папке, откуда была выполнена установка сервиса.

13.4.3.2. Установка в ОС Linux Debian

Перед установкой лог-коллектора необходимо установить утилиту `unixodbc` командой:

```
apt install unixodbc
```

При необходимости устранить проблемы установки командой:

```
apt --fix-broken install
```

Для выполнения установки необходимо распаковывать архив с дистрибутивами командой:

```
openssl enc -aes-256-cbc -d -in log-collector-<версия релиза>-<тип
архитектур>.tar.gz.enc | tar xz
```

Далее выполнить установку командой:

```
dpkg -iR ./
```

Проверка состояния сервиса

```
systemctl status pangeoradar-logcollector
```

Перезапуск сервиса

```
systemctl restart pangeoradar-logcollector
```

13.5. Основные настройки лог-коллектора

> **Важно!** Для успешного запуска лог-коллектора необходимо выполнить основные настройки в конфигурационном файле **config.yaml**

13.5.1. Настройка централизованного управления

Лог-коллектор может управляться через файл конфигурации непосредственно на сервере, где он развернут, или централизованно из веб-интерфейса **Платформы Радар**.

Для подключения экземпляра к **Платформе Радар** для управления из интерфейса, необходимо добавить в файл конфигурации следующие директивы:

```
# Централизованное управление
cluster:
  url: "http://<ip адрес Платформы Радар>:9000/cm/api/agent/"
  api_key: "<API ключ>"
```

> где:
> *<API ключ>* - ключ доступа к API, сгенерированный в веб-интерфейсе **Платформы Радар** ("Администрирование"->"Кластер"->"API ключи"->"global_api_key").

Более подробное описание по настройке централизованного управления приведено в Управление лог-коллектором.

13.5.2. Настройка контроллера

Для настройки контроллера необходимо добавить в конфигурационный файл следующие директивы:

```
controller:
  # Порт компонента, обязательный параметр
  port: 48000
```

13.5.3. Настройка компонента сбора метрик

Для настройки сбора метрик и статистики необходимо добавить в конфигурационный файл следующие директивы:

```
metric_server:
  # Порт компонента, обязательный параметр
  port: 48005
  log_level: "ERROR"
```

13.5.4. Настройка размещения защищенного хранилища

Защищенное хранилище используется для хранения чувствительных данных, которые не должны храниться в открытом виде.

Для настройки размещения необходимо добавить в конфигурационный файл следующие директивы:

```
# Путь к файлу с секретом
secret_file: "C:\\log-collector\\secret"
# Путь к хранилищу секретов
secret_storage: "C:\\log-collector\\secret.storage"
```

Для создания секрета используется команда:

```
# <путь>log-collector secrets set <ключ> <значение> --config <путь до
конфигурационного файла>
```

> где:
> <*путь*> - путь до каталога, в котором находится исполняемый файл лог-коллектора
> <*ключ*> - имя значения
> <*значение*> - данные, которые нужно скрыть в конфигурационном файле

После создания секрета его можно подставить в виде конструкции - `{{.ключ}}`, вместо любой строки в конфигурационном файле. Таким образом, все учетные данные могут быть скрыты.

Пример создания секретов и использования их в конфигурационном файле:

```
log-collector secrets set user User --config /etc/log-collector/config.yaml
```

```
log-collector secrets set user_password $ecure_P@sswOrd --config /etc/log-
collector/config.yaml
```

Пример использования в конфигурационном файле:

```
ssh_collector: &ssh_collector
  # Уникальный идентификатор компонента, отображается в журналах и метриках.
  # Обязательный параметр
  id: "ssh_collector"
  # Имя пользователя для удаленного подключения, обязательный параметр\
  user: "{{.user}}"
  # Список хостов для подключения, обязательный параметр
  remote_servers: ["<ip адрес удаленного узла>", "<имя удаленного узла>"]
  # Порт для подключения (default: 22)
  port: 22
  # Путь к файлу с ssh ключами, обязательный параметр
  rsa: "./ssh"
  # Пароль от файла с ключами
  password: "{{.user_password}}"
  # Команда для выполнения по ssh, обязательный параметр
  command: "tail -F -n +$$$line$$$ /var/log/sshfile.txt"
  # Если установлено - файл будет читаться с последней позиции в следующем тике
  # или после перезапуска
  read_from_last: true
  # Интервал между выполнением команд (в секундах)
  ticker: 30
  # Формат отправки сообщения - как есть(raw), с обогащением(json)
  format: "raw"
  # Уровень логирования, если не указан используется уровень компонента
журналирования
  log_level: "INFO"
```

Для просмотра созданных ключей используется команда:

```
# <путь>log-collector secrets list --config <путь до конфигурационногофайла>
```

> где:
> <*путь*> - путь до каталога, в котором находится исполняемый файл лог-коллектора

Для удаления ключей используется команда:

```
<путь>log-collector secrets remove <ключ> - -config <путь до конфигурационного
файла>
```

> где:
> <*путь*> - путь до каталога, в котором находится исполняемый файл лог-коллектора
> <*ключ*> - имя значения, которое нужно удалить.

> **Важно!** Прописывать ключи в файле конфигурации нужно в кавычках, как в примере выше **"{{.ключ}}"**, иначе при запуске приложения упадет с ошибкой из-за того что не сможет прочитать файл конфигурации. Соответствующее сообщение будет в журнале неудачных запусков Лог-коллектора **logcollector.crash.log**.

13.5.5. Настройка API

API необходимо для удаленного управления экземпляром лог-коллектора, получения журнала *событий* работы лог-коллектора и сбора статистики. Для настройки работы API необходимо добавить в конфигурационный файл следующие директивы:

```
api_server:
  # ip адрес, на котором будем слушать http сервер
  address: "<Внещний ip адрес сервера, на котором установлен лог-коллектор>"
  # порт, на котором будем слушать http сервер, обязательный параметр
  port: 8080
  # Таймаут чтения (получение запроса), обязательный параметр
  read_timeout: 60
  # Таймаут записи (отправка запроса), обязательный параметр
  write_timeout: 60
  # Время ожидания окончания обработки запроса при получении сигнала на остановку
приложения
  # Обязательный параметр
  wait: 5
  # Включение https (защищенное соединение)
  enable_tls: false
  # Путь для файла сертификатов, если enable_tls: false параметр не обязательный
  cert_file: "certs/server.crt"
  # Путь для файла ключей, если enable_tls: false параметр не обязательный
  key_file: "certs/server.key"
  # Пароль для расшифровки файла ключей, если не указан считаем, что файл не
зашифрован
  cert_key_pass: ""
  # Включение проверки клиентского сертификата, обязательный параметр
  require_client_cert: true
  # Путь до корневого сертификата, обязательный параметр
  ca_file: "certs/ca.crt"
  # Уровень логирования, если не указан используется указанный в компоненте
журналирования
  log_level: "INFO"
```

13.5.6. Настройка журналирования

В журнал лог-коллектора записываются все *события*, происходящие в *компонентах* лог-коллектора с уровнем логирования, указанным для каждого *компонента*. Если уровень логирования не указан для конкретного *компонента*, то логирование происходит с уровнем, выставленным в *компоненте* журналирования.

> **Важно**! Уровень логирования DEBUG используется только для отладки работы *компонентов*. В промышленной эксплуатации рекомендуется использовать уровень INFO.

```
journal:
  # Порт компонента, обязательный параметр
  port: 48004
  # Уровень логирования по умолчанию. Возможные значения - DEBUG, INFO, WARN,
ERROR.
  # Обязательный параметр
  log_level: "INFO"
  # Путь к файлу журнала, обязательный параметр
```

```
  log_path: "C:\\log-collector\\journal.log"
  # Порог ротации файла логов, указывается в мегабайтах, обязательный параметр
  rotation_size: 30
  # Порог количества файлов истории, если не указано файлы удаляться не будут
  max_backups: 7
  # Максимальное количество дней для хранения старых файлов журнала на основе
метки времени
  # Если не указано, файлы удаляться не будут (в днях).
  max_age: 7
```

13.6. Фильтрация событий

13.6.1. Структурированные данные

Фильтрация в *компонентах* сбора со структурированными данными работает как **blacklist** и применима к коллекторам wmi, eventlog, odbc, etw.

13.6.2. Неструктурированные данные

Фильтры можно указать для каждого коллектора с неструктурированными данными. Фильтры содержат белый список (**whitelist**) и черный список (**blacklist**) с массивом регулярных выражений. Все регулярные выражения проверяются перед запуском приложения. Сначала проверяется белый список, а затем черный.
Фильтрация по регулярным выражениям может быть включена в любом коллекторе с неструктурированными данными (кроме odbc, wmi, etw, eventlog). Включается путем добавления секции **filters**. В данной секции указываются два массива — **whitelist**, **blacklist**. Все события сначала проходят фильтры указанные в **whitelist**, т.к. его приоритет выше. Затем события проверяются фильтрами, указанными в **blacklist**.
Whitelist — события, которые соответствуют регулярному выражению,
попадают в очередь на отправку.
Blacklist — события, которые соответствуют регулярному выражению,
блокируются и не попадают в очередь на отправку.

```
ssh_collector: &ssh_collector
  # Уникальный идентификатор компонента, отображается в журналах и метриках.
  # Обязательный параметр
  id: "ssh_collector"
  # Имя пользователя для удаленного подключения, обязательный параметр
  user: "user"
  # Список хостов для подключения, обязательный параметр
  remote_servers: ["<ip адрес удаленного узла>", "<имя удаленного узла>"]
  # Порт для подключения (default: 22)
  port: 22
  # Путь к файлу с ssh ключами, обязательный параметр
  rsa: "./ssh"
  # Пароль от файла с ключами
  password: "password"
  # Команда для выполнения по ssh, обязательный параметр
  command: "tail -F -n +$$$line$$$ /var/log/sshfile.txt"
  # Если установлено - файл будет читаться с последней позиции в следующем тике
  # или после перезапуска
  read_from_last: true
  # Интервал между выполнением команд (в секундах)
```

```
  ticker: 30
  # Формат отправки сообщения - как есть(raw), с обогащением(json)
  format: "raw"
  # Уровень логирования, если не указан используется уровень компонента
журналирования
  log_level: "INFO"

  filters:
    whitelist:
      - "^localhost.*$"
    blacklist:
      - "^.*[0-9]$"
```

13.7. Настройка очереди отправки событий

Применимо к *компонентам* отправки событий TCP и KAFKA

```
# Максимальное количество сообщений в буфере
queue_length_limit: 1500
# максимальное время жизни событий в очереди (в секундах)
queue_time_limit: 300
```

13.8. Формат отправки данных

Применим ко всем неструктурированным *компонентам* сбора событий (кроме eventlog, wmi, odbc, etw)

```
# Формат отправки сообщения - как есть(raw), с обогащением(json)
format: "raw"
```

> **raw** - данные отправляются в том виде, в котором пришли
> **json** - пришедшие данные обогащаются дополнительной технической информацией и упаковываются в пакет json

13.9. Кодировка

Данная секция позволяет изменить кодировку входящих данных. Если не указывать original_encoding, лог-коллектор сам попытается определить кодировку.

```
# Опции смены кодировки
encoding:
  # Использовать кодировку в UTF-8
  change_to_utf8: false
  # Кодировка оригинала
  original_encoding: "cp1251"
```

13.10. Описание хранилища приложения

Хранилище ключей значений в файловой системе ОС (LevelDB). Используется для хранения ссылок и буферизации событий. Не предусмотрено стороннее редактирование.
Папка с файлами хранилища (.storage) располагается в рабочей директории лог-коллектора.

13.11. Компоненты лог-коллектора

Каждый компонент сбора и отправки может иметь один и более экземпляров в файле конфигурации.

> **Важно!** При настройке не использовать повторяющихся названий настроек компонентов, ссылок на них и уникальных идентификаторов.

Пример:

```
# Название конфигурации компонента и ссылка на него для запуска
<наименование настройки компонента>: &<уникальная ссылка на настройку>
# Уникальный идентификатор компонента, отображается в логах и метриках.
Обязательный параметр
  id: "<уникальный идентификатор>"
```

13.11.1. Компоненты сбора событий (inputs)

Для включения компонента сбора необходимо добавить в файл конфигурации его настройки.

13.11.1.1. Компонент Eventlog {#eventlog}

> **Важно!** Работает только на ОС Windows.

Позволяет выполнить локальный или удаленный сбор событий Windows.

Пример настроек по умолчанию:

```
# Название конфигурации компонента и ссылка на него для запуска
eventlog_collector: &eventlog_collector
  # Уникальный идентификатор компонента, отображается в журналах и метриках.
  # Обязательный параметр
  id: "eventlog_collector"
  # Имя канала (Security, Application, System, ForwardedEvents)
  # Используется если не указан путь к файлу
  channel: ['Security']
  # Запрос описывающий тип получаемого события. Может быть в формате
  # XPath 1.0 или структурированный XML запрос. Если XPath содержит более 20
параметров,
  # следует использовать структурированный XML запрос.
  # Чтобы получить все параметры укажите "*"
  query: "*"
  # Полный путь к файлу журналов событий
  # Поддерживаемые форматы: .evt, .evtx, .etl
  file: ""
  # Размер запроса
  batch_size: 50
  # Таймаут запроса в секундах
  timeout: 5
  # Интервал между запуском запроса в секундах
  poll_interval: 30
  # Чтение с последней сохраненной позиции
  read_from_last: true
  # Конвертировать SID в имя
  resolve_sid: false
```

```
  # формат отправки сообщения - как есть(raw), с обогащением(json)
  format: "raw"
  # Уровень логирования. Если не указан используется уровень компонента
журналирования
  log_level: "INFO"
  # количество параллельных воркеров (по умолчанию 1), 2 и более позволяет
распараллелить сбор событий
  worker_count: 1
  # Параметры удаленного подключения
  remote:
    # Включение удаленного соединения
    enabled: false
    # Имя пользователя, обязательно если enabled: true
    user: ""
    # Пароль пользователя, обязательно если enabled: true
    password: ""
    # Домен пользователя
    domain: "."
    # Адрес удаленного сервера
    remote_servers: ["<ip адрес удаленного узла>", "<имя удаленного узла>"]
    # Доступные методы авторизации: Negotiate, Kerberos, NTLM
    auth_method: "Negotiate"
    # Фильтрация по полям события, регулярные выражения
  filters:
    # Время
    # формат 2020-08-13 10:02:55.9689259 +0000 UTC
    created: ''
    # Числовые фильтры
    # Пример для числовых фильтров - ^([5-9]\d|\d{3,})$
    event_id: ''
    qualifiers: ''
    record_id: ''
    process_id: ''
    thread_id: ''
    version: ''
    # Строковые фильтры
    # пример: DESKTOP-IDCMV6G
    computer_name: ''
    msg: ''
    # Возможные значения: Information, Warning, Error
    level_text: ''
    # Пример: Service State Event
    task_text: ''
    # Пример: ServiceShutdown
    opcode_text: ''
    # Пример: System
    channel_text: ''
    # Пример: System
    provider_text: ''

    # Возможно применение опций смены кодировки
```

13.11.1.2. Компонент MS-EVEN6

```
mseven6: &mseven6
  # названия модуля, отображается в логах и метриках, уникальный обязательный
параметр
  id: "test_mseven6"
  # Список источников для сбора событий
  sources:
    -
      # Адрес удаленного сервера, с которого будут собираться события
      host: "192.168.56.6"
      # домен
      domain: ""
      #  Имя пользователя
      user: "user"
        # Пароль пользователя
      password: "0000"
      # Список каналов
      channel: ["Application"]
      # Запрос описывающий тип получаемого события. есть возможность указать
      # XPath 1.0 или структурированный XML запрос. Если XPath содержит более 20
параметров, следует
      # использовать структурированный XML запрос. Чтобы получить все параметры
укажите "*"
      query: "*"
  # Размер запроса
  batch_size: 20
  # интервал между запуском запроса в секундах (default: 1)
  poll_interval: 1
  # Чтение с последней сохраненной позиции, (default: false)
  read_from_last: true
  # уровень логирования, если не указан используется уровень модуля
журналирования
  log_level: "DEBUG"
  # путь для запуска python (лучше использовать venv, создается командой make
mseven6venv)
  python_path: "./bin/mseven6venv/bin/python"
  # порт для взаимодействия с python сервисом-прослойкой
  python_service_port: 9999
  # Фильтрация по полям события, регулярные выражения (блэклист)
  filters:
    # Время
    # формат 2020-08-13 10:02:55.9689259 +0000 UTC
    created: ''
    # Числовые фильтры
    # Пример для числовых фильтров - ^([5-9]\d|\d{3,})$
    event_id: ''
    qualifiers: ''
    record_id: ''
    process_id: ''
    thread_id: ''
    version: ''
    # Строковые фильтры
    # пример: DESKTOP-IDCMV6G
    computer_name: ''
```

```
    msg: ''
    # Возможные значения: Information, Warning, Error
    level_text: ''
    # Пример: Service State Event
    task_text: ''
    # Пример: ServiceShutdown
    opcode_text: ''
    # Пример: System
    channel_text: ''
    # Пример: System
    provider_text: 'System'
  # Опции смены кодировки
  encoding:
    # Использовать кодировку в UTF-8
    change_to_utf8: false
    # Кодировка оригинала
    original_encoding: "cp1251"
```

13.11.1.3. Компонент ODBC {#odbc}

Позволяет осуществлять сбор событий из баз данных.

```
odbc_collector: &odbc_collector
  # Уникальный идентификатор компонента, отображается в журналах иметриках.
  # Обязательный параметр
  id: "odbc_collector"
  # Интервал между запуском запроса в секундах
  poll_interval: 5
  # Чтение с последней сохраненной позиции
  read_from_last: true
  # Строка подключения, обязательный параметр
  connection_string: "server=<ip адрес удаленного узла> или <имя удаленного
узла>;port=<порт>;driver=<название драйвера>;database=<название базы данных>;Uid=
<пользователь>;Pwd=<пароль>"
  # SQL запрос, обязательный параметр
  sql: >
    SELECT id, name, dsc
    FROM test WHERE id > ?;
  # Поле, которое будет использоваться как закладка для сохранения позиции,
обязательный параметр
  # Поле должно быть целочисленным
  # Поле должно быть указано в операторе SELECT
  bookmark_field: "id"
```

Примеры строк подключения:

> **PostgreSQL**:
> Driver={PostgreSQL};Server=IP address;Port=5432;Database=myDataBase;Uid=myUsername;Pwd=myPassword;

> **MSSQL**:
> Driver={ODBC Driver 17 for SQL Server};Server=myServerAddress;Database=myDataBase;UID=myUsername;PWD=myPassword;Driver={ODBC Driver 17 for SQL Server};Server=myServerAddress;Database=myDataBase;Trusted_Connection=yes;

> **Oracle**:
> Driver={Oracle ODBC Driver};UID=Kotzwinkle;PWD=whatever;DBQ=instl_alias;DBA=W;

13.11.1.4. Компонент WMI

> **Важно!** Работает только на ОС Windows

```
wmi_collector: &wmi_collector
  # Уникальный идентификатор компонента, отображается в журналах и метриках.
  # Обязательный параметр
  id: "wmi_collector"
  # Интервал между запуском запроса в секундах
  poll_interval: 5
  # Список серверов к которым уйдет wmi запрос, обязательный параметр
  remote_servers:
    - "localhost"
    - "имя удаленного узла"
    - "ip адрес удаленного узла"
  # Имя пользователя, обязательно если это не локальный сбор
  # Для сбора в домене "domain\\user"
  user: "user"
  # Пароль пользователя, обязательно если это не локальный сбор
  password: ""
  # Чтение с последней сохраненной позиции
  read_from_last: true
  # Уровень логирования, если не указан используется уровень компонента
журналирования
  log_level: "INFO"
  # изменение кодировки входящего события, может быть прописана у любого
коллектора
  encoding:
    # включение изменения кодировки
    change_to_utf8: false
    # оригинальная кодировка события, если оставить пустым, произойдет попытка
определить кодировку
    # нет 100% гарантии определения
    original_encoding: "cp1251"
  # Собирать события начиная с заданного момента
  start_from_date: "2022-03-24T00:00:00+03:00"
  # Список журналов, из которых собираются события (Application, System и т.п.).
Если пустой или не указан, собираются все события
  logfiles: ["Application"]
  # Блэклист фильтры по полям события, используются регулярные выражения
  wmi_filters:
    # Числовые поля
    category: ''
    event_code: ''
    event_identifier: ''
    event_type: ''
    record_number: ''
    # Строковые поля
    computer_name: ''
    message: ''
    source_name: ''
    type: ''
```

```
    user: ''
    time_generated: ''
    time_written: ''

  # Возможно применение опций смены кодировки
```

13.11.1.5. Компонент ETW

> **Важно!** Работает только на ОС Windows

```
etw_collector: &etw_collector
  # Имя провайдера или GUID
  # Формат GUID должен быть "{9E814AAD-3204-11D2-9A82-006008A86939}"
  id: "etw_collector"
  provider: "{A68CA8B7-004F-D7B6-A698-07E2DE0F1F5D}"
  kernel_args: [ "ALPC", "CSWITCH", "DBGPRINT", "DISK_FILE_IO", "DISK_IO",
"DISK_IO_INIT", "DISPATCHER", "DPC", "DRIVER", "FILE_IO", "FILE_IO_INIT",
"IMAGE_LOAD", "INTERRUPT", "MEMORY_HARD_FAULTS", "MEMORY_PAGE_FAULTS",
"NETWORK_TCPIP", "NO_SYSCONFIG", "PROCESS", "PROCESS_COUNTERS", "PROFILE",
"REGISTRY", "SPLIT_IO", "SYSTEMCALL", "THREAD", "VAMAP", "VIRTUAL_ALLOC" ]
  provider_level: "Information"
  # Уровень логирования, если не указан используется уровень компонента
журналирования
  log_level: "INFO"

  # Возможно применение опций смены кодировки
```

13.11.1.6. Компонент OPSEC LEA

> **Важно!** Работает только на ОС Linux

```
opsec_lea_collector: &opsec_lea_collector
  # Уникальный идентификатор компонента, отображается в журналах и метриках.
  # Обязательный параметр
  id: "opsec_lea_collector"
  # Директория расположения утилиты lea_client
  exec_path: "opsec"
  # Периодичность проверки наличия новых записей в журналах
  poll_interval: 5
  # Сохранение позиции последнего чтения из журнала (сохранение на диск),
возобновление чтения с последней сохраненной позиции.
  read_from_last: false
  # Сервер для сбора событий.
  remote_server: "<ip адрес или имя удаленного узла>"
  # Порт для аутентификации.
  auth_port: 18184
  # Аутентификация для OPSEC
  auth_type: "sslca"
  # Параметры авторизации
  opsec_sic_name: "CN=lea_logger,O=vmfw..ktz7qd"
  opsec_sslca_file: "/home/lea/lea_client/opsec.p12"
  opsec_entity_sic_name: "cn=cp_mgmt,o=vmfw..ktz7qd"
  opsec_sic_policy_file: ""
  # Название собираемого журнала.
```

```
  log_filename: "fw.log"
  # Формат отправки сообщения - как есть(raw), с обогащением(json)
  format: "raw"
  # Уровень логирования, если не указан используется уровень компонента
журналирования
  log_level: "INFO"
```

13.11.1.7. Компонент SSH

```
ssh_collector: &ssh_collector
  # Уникальный идентификатор компонента, отображается в журналах и метриках.
  # Обязательный параметр
  id: "ssh_collector"
  # Имя пользователя для удаленного подключения, обязательный параметр
  user: "user"
  # Список хостов для подключения, обязательный параметр
  remote_servers: ["<ip адрес удаленного узла>", "<имя удаленного узла>"]
  # Порт для подключения (default: 22)
  port: 22
  # Путь к файлу с ssh ключами, обязательный параметр
  rsa: "./ssh"
  # Пароль от файла с ключами
  password: ""
  # Команда для выполнения по ssh, обязательный параметр
  command: "tail -F -n +$$$line$$$ /var/log/sshfile.txt"
  # Если установлено - файл будет читаться с последней позиции в следующем тике
или после перезапуска
  read_from_last: true
  # Интервал между выполнением команд (в секундах)
  ticker: 30
  # Формат отправки сообщения - как есть(raw), с обогащением(json)
  format: "raw"
  # Уровень логирования, если не указан используется уровень компонента
журналирования
  log_level: "INFO"

  # Возможно применение опций смены кодировки
```

13.11.1.8. Компонент SMB

```
smb_collector: &smb_collector
  # Уникальный идентификатор компонента, отображается в журналах и метриках.
  # Обязательный параметр
  id: "smb_collector"
  # Список хостов для подключения, обязательный параметр
  remote_servers: ["<ip адрес удаленного узла>", "<имя удаленного узла>"]
  # Порт подключения
  port: 445
  # SMB share. sharename должен соответсвовать формату `<share>` или `\\\\
<server>\\<share>`, обязательный параметр
  share: "<путь к общему ресурсу>"
  # Домен
  domain: "."
  # NTLMv2 пользователь, обязательный параметр
```

```
  user: "user"
  # NTLMv2 пароль(или hash), обязательный параметр
  password: "password"
  # настройки аутентификации kerberos
  kerberos:
    # включение авторизации kerberos
    enabled: false
    # имя целевого сервиса (service principal name)
    target_spn: "pdc"
    # kerberos realm
    realm: "TEST.TEST"
    # путь до конфигурации kerberos
    config_path: "assets/krb5/krb5.conf"
  # Интервал между запуском сканирования файлов в секундах
  poll_interval: 5
  # Список файлов для чтения, обязательный параметр
  files: [ "smbfile.txt" ]
  # Если установлено - использовать регулярное выражение для поиска файлов
  using_regexp: true
  # Начальный каталог для поиска файлов
  regexp_starting_dir: "."
  # Регулярное выражение для поиска файлов
  regexp_expression: ".(?:txt|log)$"
  # Интервал проверки файлов (в секундах) в дереве каталогов
  dir_check_interval: 5
  # Если установлено - файл будет читаться с последней позиции в следующем тике
  # или после перезапуска
  read_from_last: true
  # Формат отправки сообщения - как есть(raw), с обогащением(json)
  format: "raw"
  # Уровень логирования, если не указан используется уровень компонента
журналирования
  log_level: "INFO"

  # Возможно применение опций смены кодировки
```

13.11.1.9. Компонент FTP

```
ftp_collector: &ftp_collector
  # Уникальный идентификатор компонента, отображается в журналах и метриках.
  # Обязательный параметр
  id: "ftp_collector"
  # Список хостов для подключения, обязательный параметр
  remote_servers: ["<ip адрес удаленного узла>", "<имя удаленного узла>"]
  # Порт для ftp запросов, обязательный параметр
  port: 21
  # ftp пользователь, обязательный параметр
  user: ""
  # ftp пароль, обязательный параметр
  password: ""
  # Интервал между сканированием файла в секундах
  poll_interval: 5
  # Список файлов для чтения, обязательный параметр
  files: ["ftpfile.txt"]
  # Если установлено - использовать регулярное выражение для поиска файлов
```

```
  using_regexp: true
  # Начальный каталог для поиска файлов
  regexp_starting_dir: "."
  # Регулярное выражение для поиска файлов
  regexp_expression: ".(?:txt|log)$" #"^.*\_logs$"
  # Интервал проверки файлов (в секундах) в дереве каталогов
  dir_check_interval: 5
  # Если установлено - файл будет читаться с последней позиции в следующем тике
  # или после перезапуска
  read_from_last: true
  # Формат отправки сообщения - как есть(raw), с обогащением(json)
  format: "raw"
  # Уровень логирования, если не указан используется уровень компонента
журналирования
  log_level: "INFO"

  # Возможно применение опций смены кодировки
```

13.11.1.10. Компонент SFTP

```
sftp_collector: &sftp_collector
  # Уникальный идентификатор компонента, отображается в журналах и метриках.
  # Обязательный параметр
  id: "sftp_collector"
  # Список хостов для подключения, обязательный параметр
  remote_servers: ["<ip адрес удаленного узла>", "<имя удаленного узла>"]
  # Порт для sftp запросов, обязательный параметр
  port: 22
  # Пользователь ssh, обязательный параметр
  user: "user"
  # Пароль ssh, обязательный параметр
  password: "password"
  # Интервал между сканированием файла в секундах
  poll_interval: 5
  # Список файлов для чтения, обязательный параметр
  files: ["sftptest.txt"]
  # Если установлено - использовать регулярное выражение для поиска файлов
  using_regexp: false
  # Начальный каталог для поиска файлов
  regexp_starting_dir: "upload"
  # Регулярное выражение для поиска файлов
  regexp_expression: ".(?:txt|log)$" #"^.*_logs$"
  # Интервал проверки файлов (в секундах) в дереве каталогов
  dir_check_interval: 5
  # Если установлено - файл будет читаться с последней позиции при следующем тике
или после перезапуска
  read_from_last: true
  # Формат отправки сообщения - как есть(raw), с обогащением(json)
  format: "raw"
  # Уровень логирования, если не указан используется уровень компонента
журналирования
  log_level: "INFO"

  # Возможно применение опций смены кодировки
```

13.11.1.11. Компонент NetFlow {#netflow}

```
netflow_input: &netflow_input
  # Уникальный идентификатор компонента, отображается в журналах и метриках.
  # Обязательный параметр
  id: "netflow_input"
  # Хост на каком запустится сервер (default: localhost)
  host: "<ip адрес лог-коллектора>"
  # Порт на каком запустится сервер (обязательное)
  port: 2162
  # Размер буфера сообщений (если не задано то берется из SO_RCVBUF)
  sock_buf_size: 0
  # Формат отправки сообщения - как есть(raw), с обогащением(json)
  format: "raw"
  # Уровень сообщений в логах, могут быть значения - DEBUG, INFO, WARN, ERROR
  log_level: "INFO"

  # Возможно применение опций смены кодировки
```

13.11.1.12. Компонент TCP {#tcp}

```
tcp_input: &tcp_input\
  # Уникальный идентификатор компонента, отображается в журналах и метриках.
  # Обязательный параметр
  id: "tcp_input"
  # Хост на каком запустится сервер
  host: "<ip адрес лог-коллектора>"
  # Порт на каком запустится сервер
  port: <порт для приема соединений>
  # Включение TLS соединения на сервере
  enable_tls: false
  # Файл с приватным ключом (обязательное поле при включенном TLS)
  key_file: "certs/server.key"
  # Файл с сертификатом должно быть подписанным сертификатом CA (обязательное
поле при включенном TLS)
  cert_file: "certs/server.crt"
  # Файл с паролем если сертификат подписывался с паролем
  cert_key_pass: ""
  # Файл с сертификатом CA (обязательное поле при включенном TLS)
  ca_file: "certs/ca.crt"
  # Проверять ли сертификаты клиента
  require_client_cert: false
  # Нужна ли распаковка тела запроса, ожидается, что клиент упаковал тело запроса
в архив (default: false)
  compression_enabled: false
  # Количество соединений, которые может принять сервер
  connections_limit: 10
  # Формат отправки сообщения - как есть(raw), с обогащением(json)
  format: "raw"
  # Уровень логирования, если не указан используется уровень компонента
журналирования
  log_level: "INFO"

  # Возможно применение опций смены кодировки
```

13.11.1.13. Компонент UDP {#udp}

```
udp_input: &udp_input
  # Уникальный идентификатор компонента, отображается в журналах и метриках.
  # Обязательный параметр
  id: "udp_input"
  # Хост на каком запустится сервер
  host: "<ip адрес лог-коллектора>"
  # Порт на каком запустится сервер
  port: <порт для приема соединений>
  # Размер буфера сообщений (если незаданно то берется из SO_RCVBUF)
  sock_buf_size: 0
  # формат отправки сообщения - как есть(raw), с обогащением(json)
  format: "raw"
  # Уровень логирования, если не указан используется уровень компонента
журналирования
  log_level: "INFO"

  # Возможно применение опций смены кодировки
```

13.11.1.14. Компонент HTTP приемник

```
http_input: &http_input
  # Уникальный идентификатор компонента, отображается в журналах и метриках.
  # Обязательный параметр
  id: "http_input"
  # Хост на каком запустится сервер (default: localhost)
  host: "<ip адрес лог-коллектора>"
  # Порт на каком запустится сервер (обязательное)
  port: <порт для приема соединений>
  # Включение TLS соединения на сервере (default: false)
  enable_tls: false
  # Файл с приватным ключом (обязательное поле при включенном TLS)
  key_file: "certs/server.key"
  # Файл с сертификатом должно быть подписанным сертификатом CA (обязательное
поле при ключенном TLS)
  cert_file: "certs/server.crt"
  # Файл с паролем если сертификат подписывался с паролем
  cert_key_pass: ""
  # Файл с сертификатом CA (обязательное поле при включенном TLS)
  ca_file: "certs/ca.crt"
  # Проверять ли сертификаты клиента (default: false)
  require_client_cert: false
  # Количество соединений, которые может принять сервер
  connections_limit: 10
  # формат отправки сообщения - как есть(raw), с обогащением(json)
  format: "raw"
  # Уровень логирования, если не указан используется уровень компонента
журналирования
  log_level: "INFO"

  # Возможно применение опций смены кодировки
```

13.11.1.15. Компонент HTTP сборщик

```
http_collector: &http_collector\
  # Уникальный идентификатор компонента, отображается в журналах и метриках.
  # Обязательный параметр
  id: "http_collector"
  # Удаленный адрес для вызовов http (обязательное)
  remote_server: "<ip адрес или имя удаленного узла>"
  # Удаленный порт (default: 80)
  port: <порт на целевой системе>
  # Имя пользователя для базовой авторизации, если пустое, считаем, что
авторизация выключена
  basic_auth_user: ""
  # Пароль для базовой авторизации
  basic_auth_password: ""
  # Ограничение по времени для запросов, сделанных http-клиентом в секундах
(default: 10)\
  timeout: 10
  # Если установлено - будет использоваться tls клиент
  enable_tls: false
  # Путь к .key файлу, обязательно если enable_tls: true
  key_file: "certs/server.key"
  # Путь к .crt файлу, обязательно если enable_tls: true
  cert_file: "certs/server.crt"
  # Пароль к файлу сертификатов
  cert_key_pass: ""
  # Путь к файлу с набором корневых центров сертификации, обязательно
  если enable_tls: true
  ca_file: "certs/ca.crt"
  # Имя файла для получения по http
  file: "httptest.txt"
  # Если установлено - файл будет читаться с последней позиции в следующем тике
или после перезапуска (default: false)
  read_from_last: true
  # Интервал между http-вызовами в секундах
  poll_interval: 5
  # Формат отправки сообщения - как есть(raw), с обогащением(json)
  format: "raw"
  # Уровень логирования, если не указан используется уровень компонента
журналирования
  log_level: "INFO"

  # Возможно применение опций смены кодировки
```

13.11.1.16. Компонент File

```
file_input: &file_input
  # Уникальный идентификатор компонента, отображается в журналах и метриках.
  # Обязательный параметр
  id: "file_input"
  # Интервал между чтениями файла, в секундах)
  poll_interval: 5
  # Список файлов для чтения
  files: ["logfile.txt"]
```

```
  # Использовать regexp для поиска файлов
  using_regexp: false
  # Начальный каталог для поиска файлов
  regexp_starting_dir: "."
  # regexp для поиска файлов
  regexp_expression: "^.*_logs$"
  # Интервал поиска файлов в секундах, в дереве каталогов
  dir_check_interval: 5
  # Если установлено - файл будет читаться с последней позиции в следующем тике
или после перезапуска\
  read_from_last: true
  # Создает file watchers для всех файлов
  enable_watcher: true
  # Формат отправки сообщения - как есть(raw), с обогащением(json)
  format: "raw"
  # Уровень логирования, если не указан используется уровень компонента
журналирования
  log_level: "INFO"

  # Возможно применение опций смены кодировки
```

13.11.1.17. Компонент External Command

```
external_command_input: &external_command_input
  # Уникальный идентификатор компонента, отображается в журналах и метриках.
  # Обязательный параметр
  id: "external_command_input"
  # Интервал между выполнениями команд
  poll_interval: 5
  # Команда bash/cmd
  command: "<команда выполнения скрипта>"
  # Формат отправки сообщения - как есть(raw), с обогащением(json)
  format: "raw"
  # Уровень логирования, если не указан используется уровень компонента
журналирования
  log_level: "INFO"

  # Возможно применение опций смены кодировки
```

13.11.1.18. Компонент SNMP Trap

```
snmp_trap: &snmp_trap
  # Уникальный идентификатор компонента, отображается в журналах и метриках.
  # Обязательный параметр
  id: "snmp_trap"
  # Адресс snmp менеджера
  host: "<ip адрес лог-коллектора>"
  # Порт для запуска snmp менеджера
  port: 162
  # Принимать только аутентифицированные SNMP v3 Traps
  allow_authenticated_only: false
  # Список директорий с .mib файлами для конвертации oid
  # Если не указаны, oid будут передаваться в сыром виде
  mib_dirs:
```

```
    - dir1
    - dir2
    - dir3
  # Параметры безопасности
  # Методы аутентификации. Возможные значения:
  # - MD5
  # - SHA
  auth_proto: "SHA"
  # Методы шифрования. Поддерживается только DES.
  encrypt_proto: "DES"
  # Имя SNMP пользователя
  user_name: "user"
  # Пароль аутентификации. Используется с MD5 или SHA
  authentication_passphrase: "user_pass"
  # Пароль шифрования для DES
  privacy_passphrase: "priv_user_pass"
  # Используется в SNMPv3 для идентификации сущностей.
  authoritative_engine_id: "880000009fe71969bdd782bbc691c06b524d70324abe96c0755"
  # Формат отправки сообщения - как есть(raw), с обогащением(json)
  format: "raw"
  # Уровень логирования, если не указан используется уровень компонента
журналирования
  log_level: "INFO"

  # Возможно применение опций смены кодировки
```

13.11.2. Компоненты отправки событий (outputs)

Для включения компонента отправки событий необходимо добавить в файл конфигурации его настройки.

13.11.2.1. Компонент отправки событий по протоколу TCP {#tcp_send}

Позволяет отправлять данные по протоколу TCP. Также есть возможность отправки в зашифрованном виде, со сжатием, с настройкой буферизации.

```
tcp_output: &tcp_output
  # Уникальный идентификатор компонента, отображается в журналах и метриках.
  # Обязательный параметр
  id: "tcp_output"
  # Адрес куда отправлять события, обязательный параметр
  target_host: "<ip адрес или имя удаленного узла>"
  # Порт куда отправлять события, обязательный параметр
  port: <порт на целевой системе>
  # Включение batch режима
  batch_mode_enable: false
  # Период отправки пакета в секунундах при включенном batch режиме
  batch_flush_interval: 5
  # Количество сообщений, которые попадут в пакет при включенном batch режиме
  batch_flush_limit: 500
  # Включение сжатия, включение при выключенном batch режиме ощутимо замедляет
отправку
  ssl_compression: false
  # Включение проверки сертификата
  require_cert: false
```

```
  # Включение ssl\
  ssl_enable: false
  # Путь для файла сертификатов, если enable_tls: false параметр не обязательный
  cert_file: "client-cert.pem"
  # Путь для файла ключей, если enable_tls: false параметр не обязательный
  key_file: "client-key.pem"
  # Пароль для расшифровки файла ключей, если не указан считаем, что файл не
зашифрован
  cert_key_pass: ""
  # Путь до корневого сертификата, если enable_tls: false не обязательный
параметр
  ca_file: "ca.pem"
  # уровень логирования, если не указан используется уровень компонента
журналирования
  log_level: "INFO"
  # максимальное количество сообщений в буфере
  #queue_length_limit: 1500
  # максимальное время жизни событий в очереди. в секундах
  #queue_time_limit: 300
```

13.11.2.2. Компонент отправки событий по протоколу UDP

Позволяет отправлять данные по протоколу UDP.

```
udp_output: &udp_output
  # Уникальный идентификатор компонента, отображается в журналах и метриках.
  # Обязательный параметр
  id: "udp_output"
  # Адрес куда отправлять события
  target_host: "<ip адрес или имя удаленного узла>"
  # Порт, на который отправлять события. Обязательный параметр
  port: <порт на целевой системе>
  # Размер буфера для отправки, если не указан или равен нулю используется
системное значение
  sock_buf_size: 0
  # Уровень логирования, если не указан используется уровень компонента
журналирования
  log_level: "INFO"
```

13.11.2.3. Компонент отправки событий в KAFKA

Позволяет отправлять данные в KAFKA.

```
kafka_output: &kafka_output
  # Уникальный идентификатор компонента, отображается в журналах и метриках.
  # Обязательный параметр
  id: "kafka_output"
  # Включение проверки сертификата (default: false)
  require_cert: false
  # Включение ssl (default: false)
  ssl_enable: false
  # Путь для файла сертификатов, если ssl_enable: false параметр не обязательный
  cert_file: "certs/server.crt"
  # Путь для файла ключей, если ssl_enable: false параметр не обязательный
```

```
  key_file: "certs/server.key"
  # Пароль для расшифровки файла ключей, если не указан считаем, что файл не
зашифрован
  cert_key_pass: ""
  # Путь до корневого сертификата, если ssl_enable: false параметр не
обязательный
  ca_file: "certs/ca.crt"
  # Таймауту отправки события в секундах, обязательный параметр
  timeout: 10
  # Топик в который попадет событие, обязательный параметр
  topic: "<наименование топика>"
  # Уровень логирования, если не указан используется уровень компонента
журналирования
  log_level: "INFO"
  # Kafka брокеры, обязательный параметр
  brokers:
    - "<ip адрес или имя удаленного узла>:9092"
  # Максимальное количество сообщений в буфере
  #queue_length_limit: 1500
  # Максимальное время жизни событий в очереди (в секундах)
  #queue_time_limit: 300
```

13.11.2.4. Компонент записи событий в локальный файл

Позволяет записать входящие события в локальный файл.

```
out_file: &out_file
  # Уникальный идентификатор компонента, отображается в журналах и метриках.
  # Обязательный параметр
  id: "file_output"
  # Путь до файла куда будут записываться события, обязательный параметр
  file: "ouput_file.txt"
  # Порог ротации в мегабайтах, если указан ноль ил не указан совсем ротация не
происходит
  rotation_size: 10
  # Уровень логирования, если не указан используется уровень компонента
журналирования
  log_level: "INFO"
```

13.12. Включение компонентов {#on_components}

Для того, чтобы включить *компоненты* сбора или отправки, необходимо добавить их в разделах настроек collectors для *компонентов* сбора или senders для *компонентов* отправки в конфигурационном файле Лог-коллектора.

13.12.1. Включение компонентов сбора (collectors)

В разделе collectors необходимо прописать следующие настройки (пример для *компонента* сбора eventlog):

```
collectors:
  # Уровень логирования, если не указан используется уровень логирования
компонента журналирования\
  log_level: "INFO"
  # eventlog коллектор, работает только на windows vista и старше
  event_log:
  - <<: *eventlog_collector
```

13.12.2. Включение компонентов отправки (senders)

В разделе senders необходимо прописать следующие настройки (пример для *компонента* отправки по TCP):

```
senders:
  # Порт компонента, обязательный параметр
  port: 48002
  # Уровень логирования, если не указан используется уровень логирования
компонента журналирования
  log_level: "INFO"
  # Отправка по протоколу tcp
  tcp:
    - <<: *tcp_output
```

13.13. Маршрутизация событий {#event_route}

Для организации маршрутизации необходимо выполнить следующие действия для связывания *компонентов* сбора и *компонентов* отправки:

1. Настроить маршруты взаимодействия между *компонентами* сбора событий и *компонентами* отправки событий.

   Пример настройки маршрута:

   ```
   route_1: &route_1
   collector_id:
       - "eventlog_collector"
       - "tcp_input"
   sender_id:
       - "tcp_output"
   ```

2. Включить маршрут в разделе конфигурационного файла routers.

   Пример включения маршрута:

   ```
   routers:
   - <<: *route_1
   ```

> **Важно!** *Компоненты*, используемые в маршрутах, обязательно должны быть включены в разделах **collectors** и **senders** и настроены.

13.14. Инструкция по настройке AppLocker {#applocker}

1. Проверить статусы и при необходимости запустить службы Application Management и Application Identity.
2. Зайти в secpol.msc -> Application Control Policies -> AppLocker -> Configure rule enforcement.

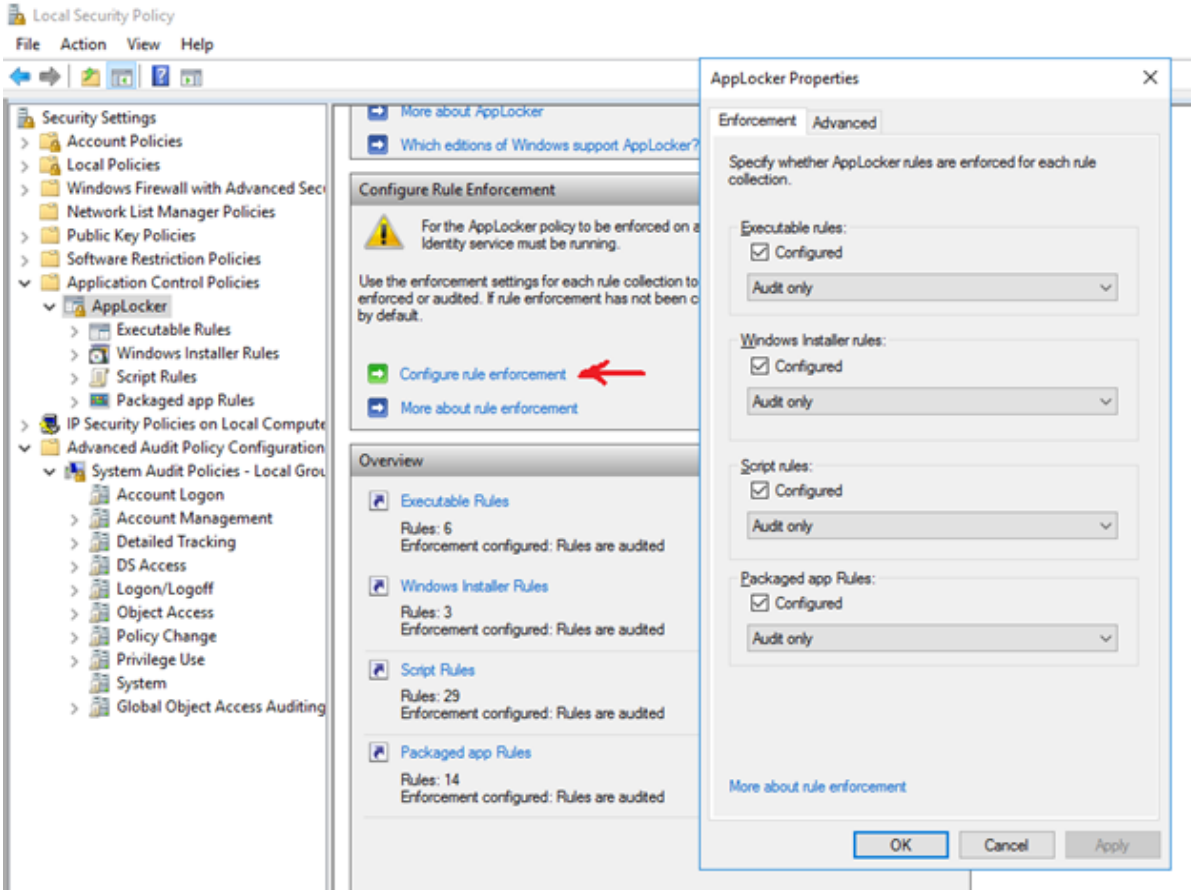

Рисунок 163

3. Включить все наборы правил и выставить тип работы «Audit only», кроме правил типа «Executable rules» (enforce - для получения события 8004).

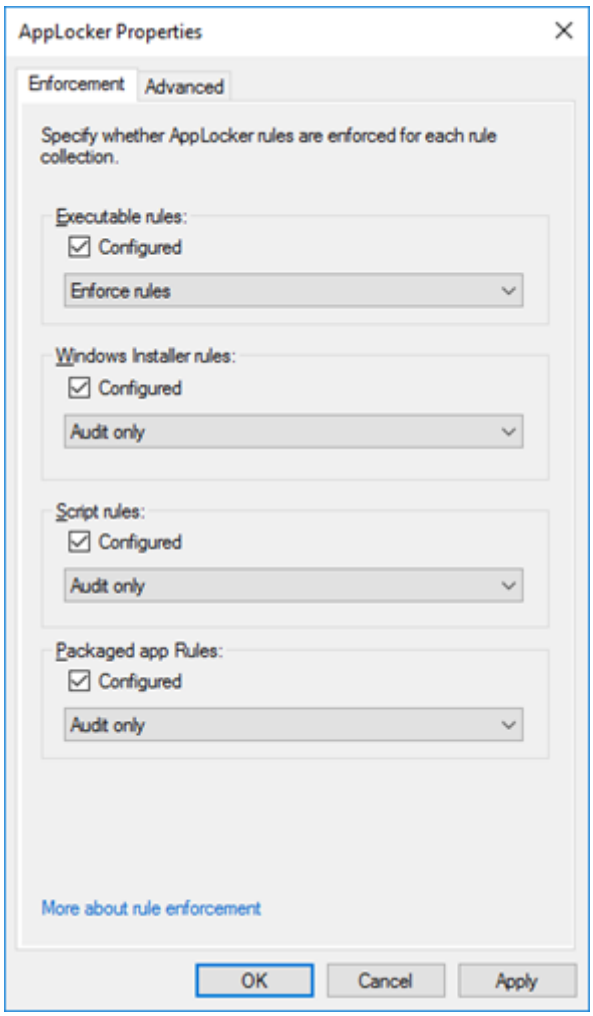

Рисунок 164

4. Наполнить наборы правилами. «Automatically Generate Rules» - автоматически, «Create New Rule» - вручную.

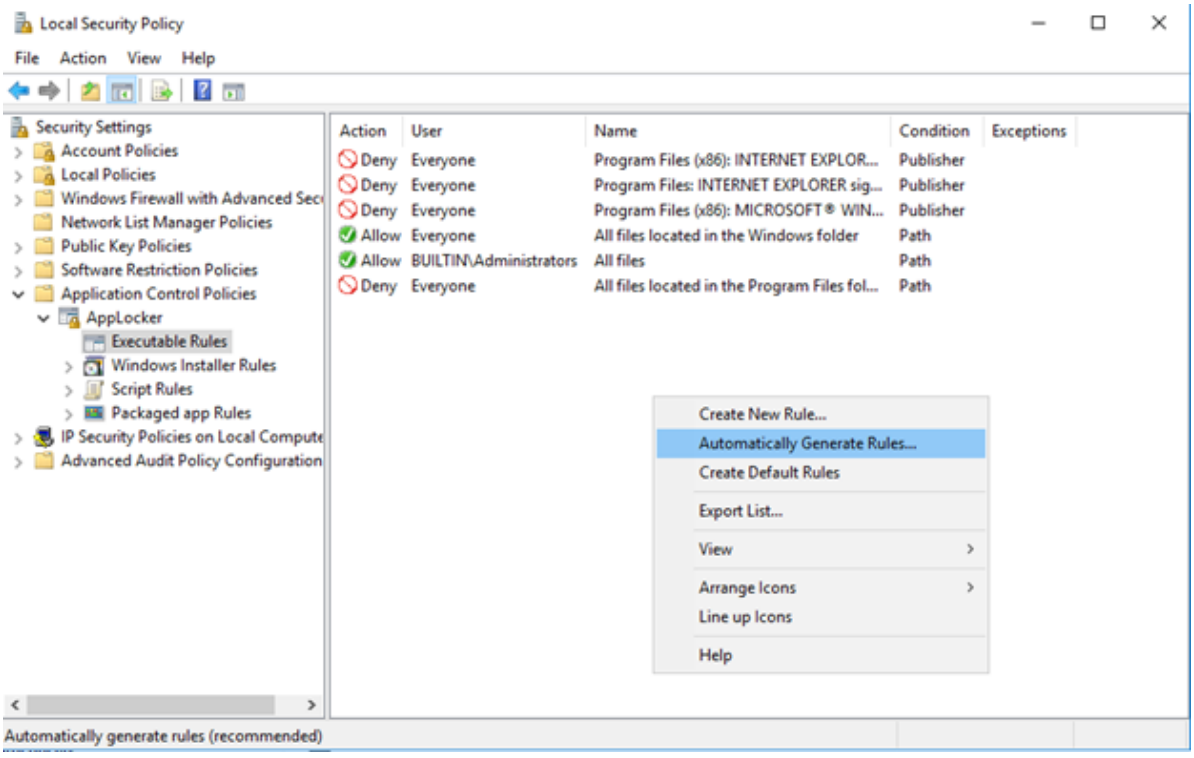

Рисунок 165

- Автоматически.

Указать пользователя или группу пользователей, на кого будет применяться правило, а также указать путь, который включает файлы для анализа.

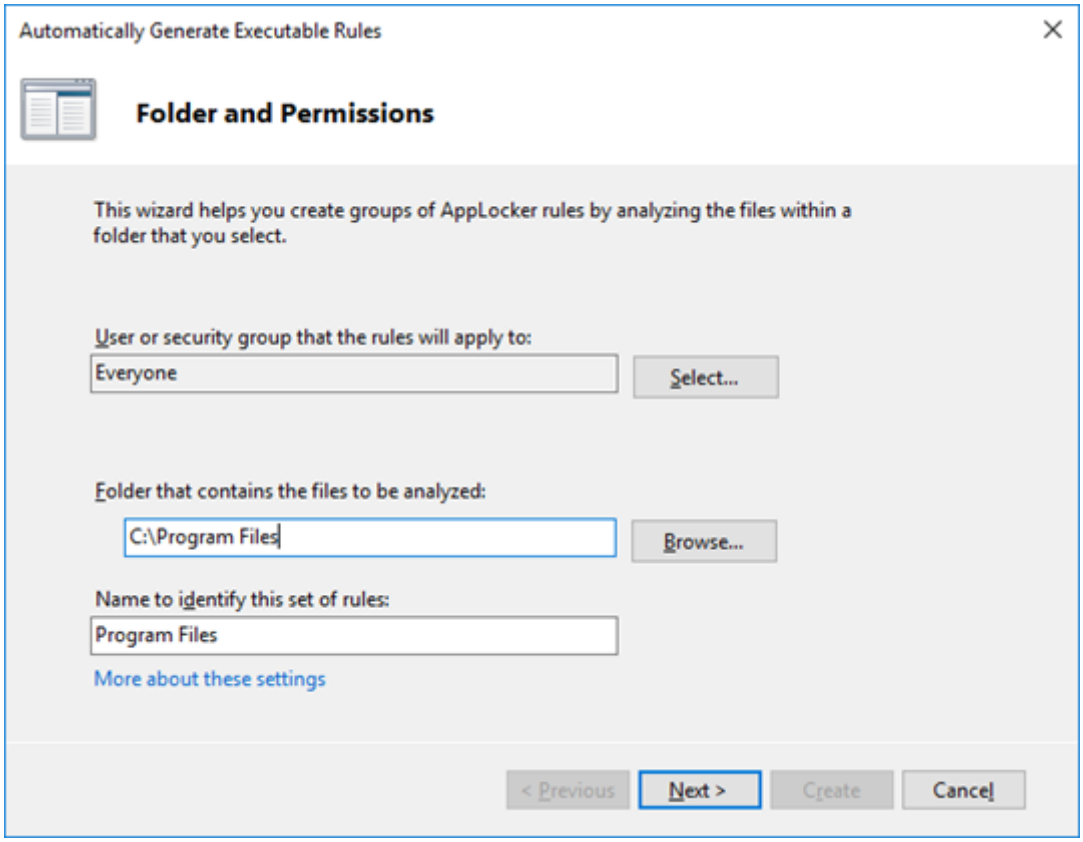

Рисунок 166

Указать, как будут анализироваться файлы: по сертификату, по хэшу или по пути.

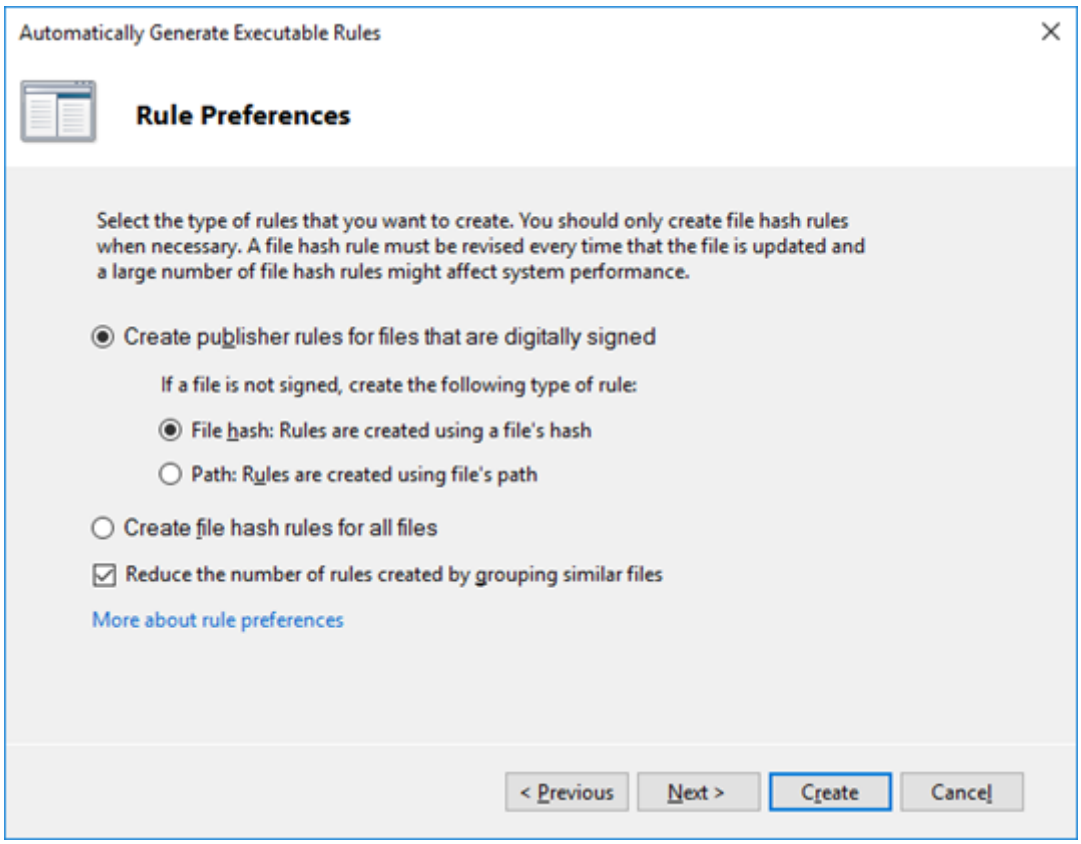

Рисунок 167

```
Проверить правило и нажать «Create».
```

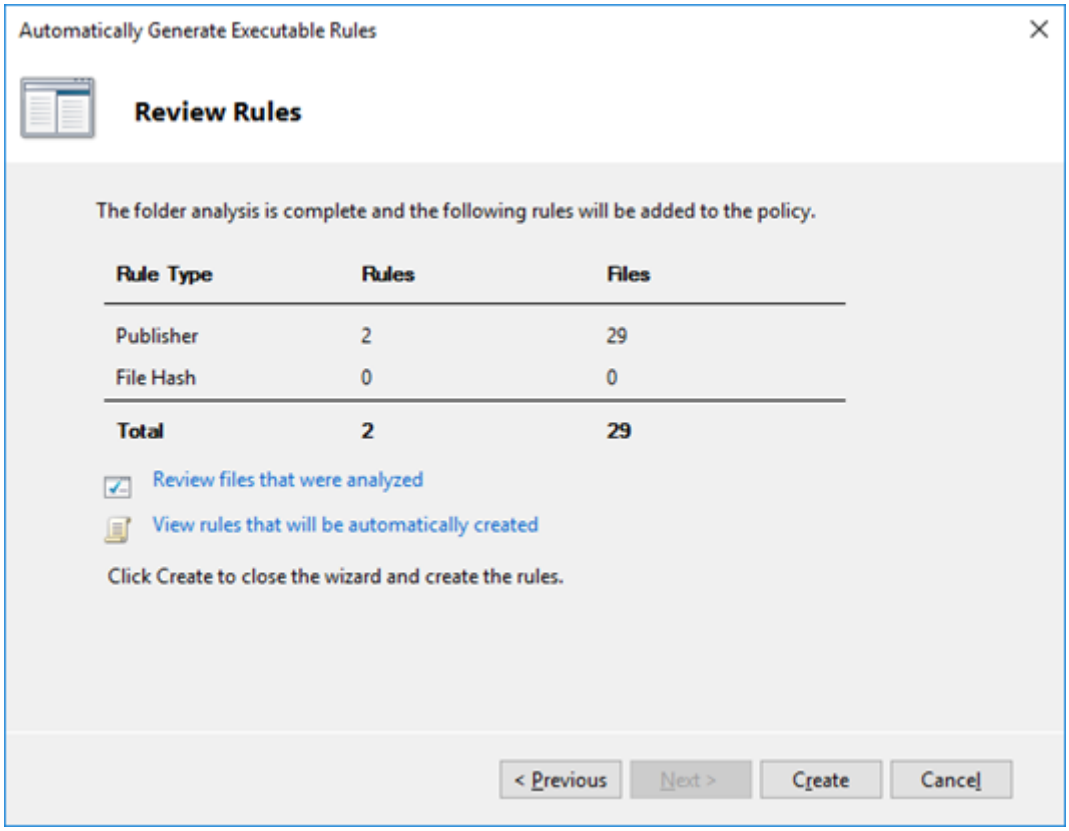

Рисунок 168

```
- Вручную.
```

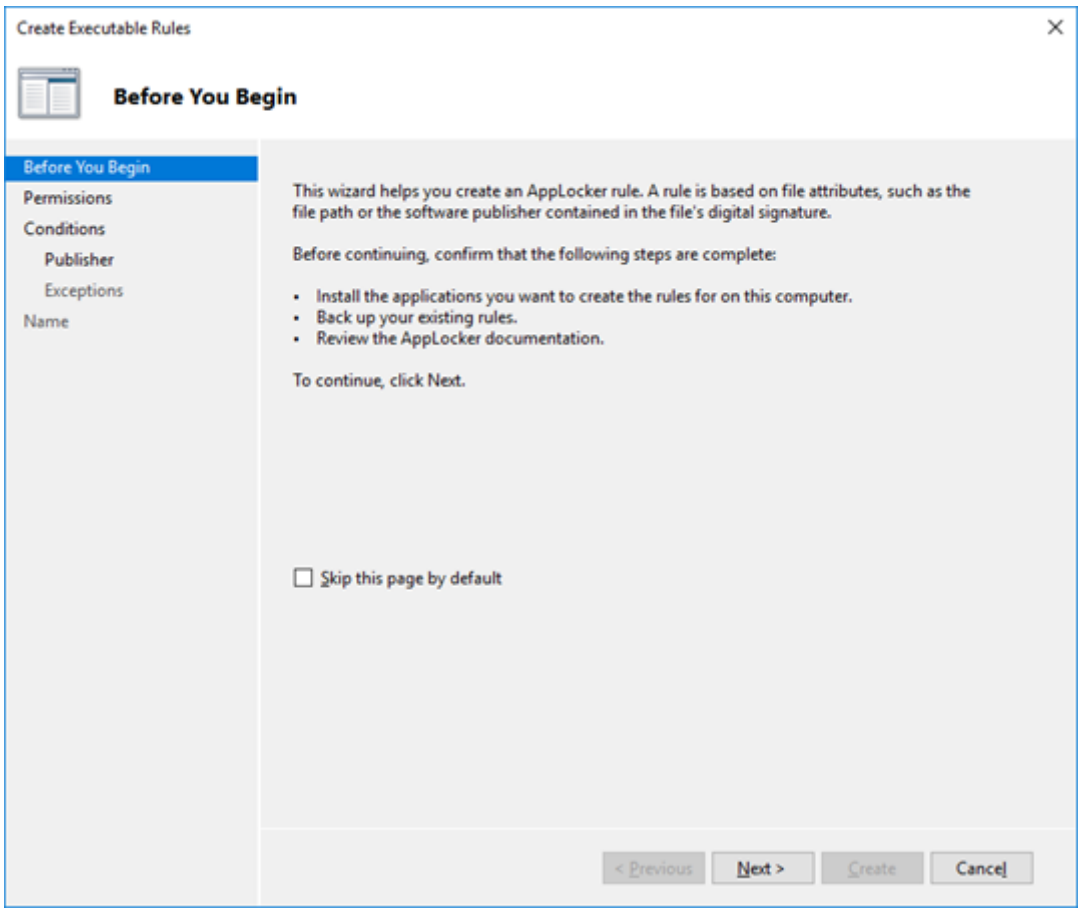

Рисунок 169

Указать действие (разрешить или запретить запуск) и пользователя (группу пользователей), на кого применится правило.

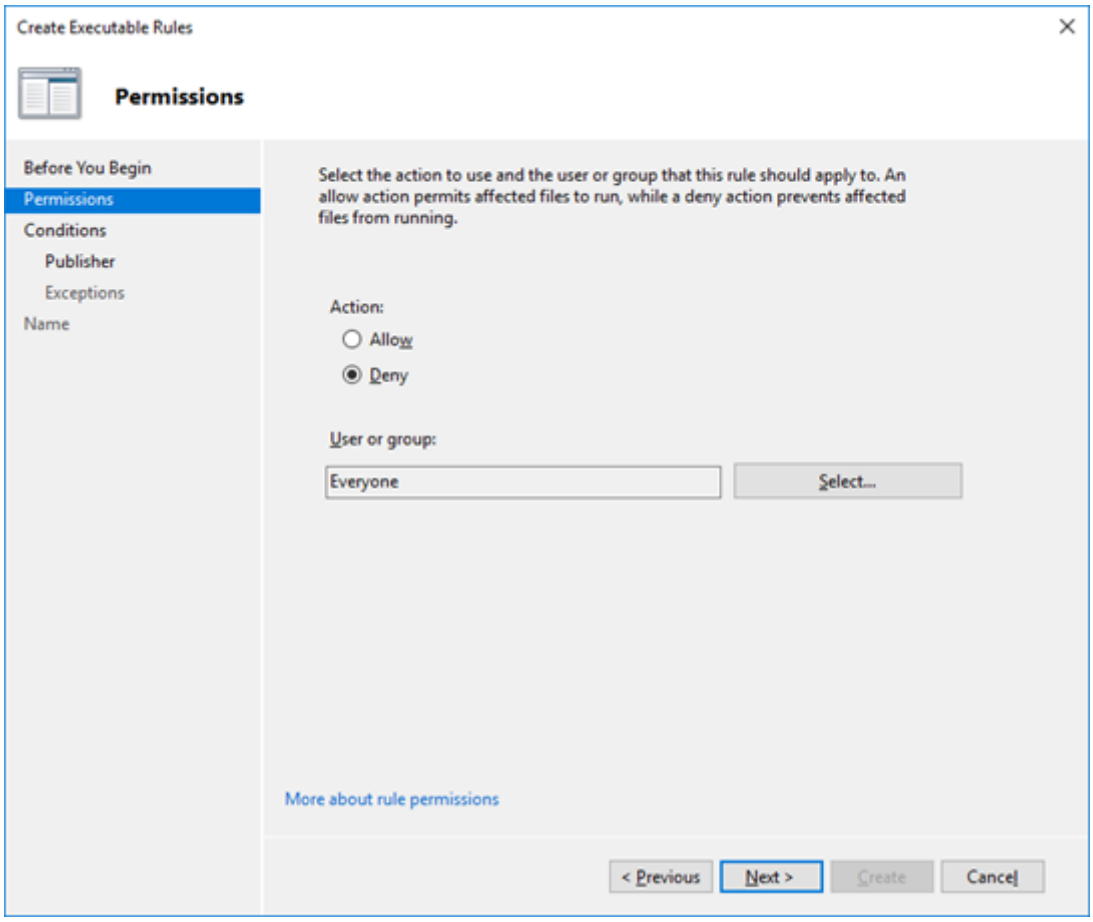

Рисунок 170

Выбрать тип проверки файла: по сертификату, либо по пути, либо по хэшу.

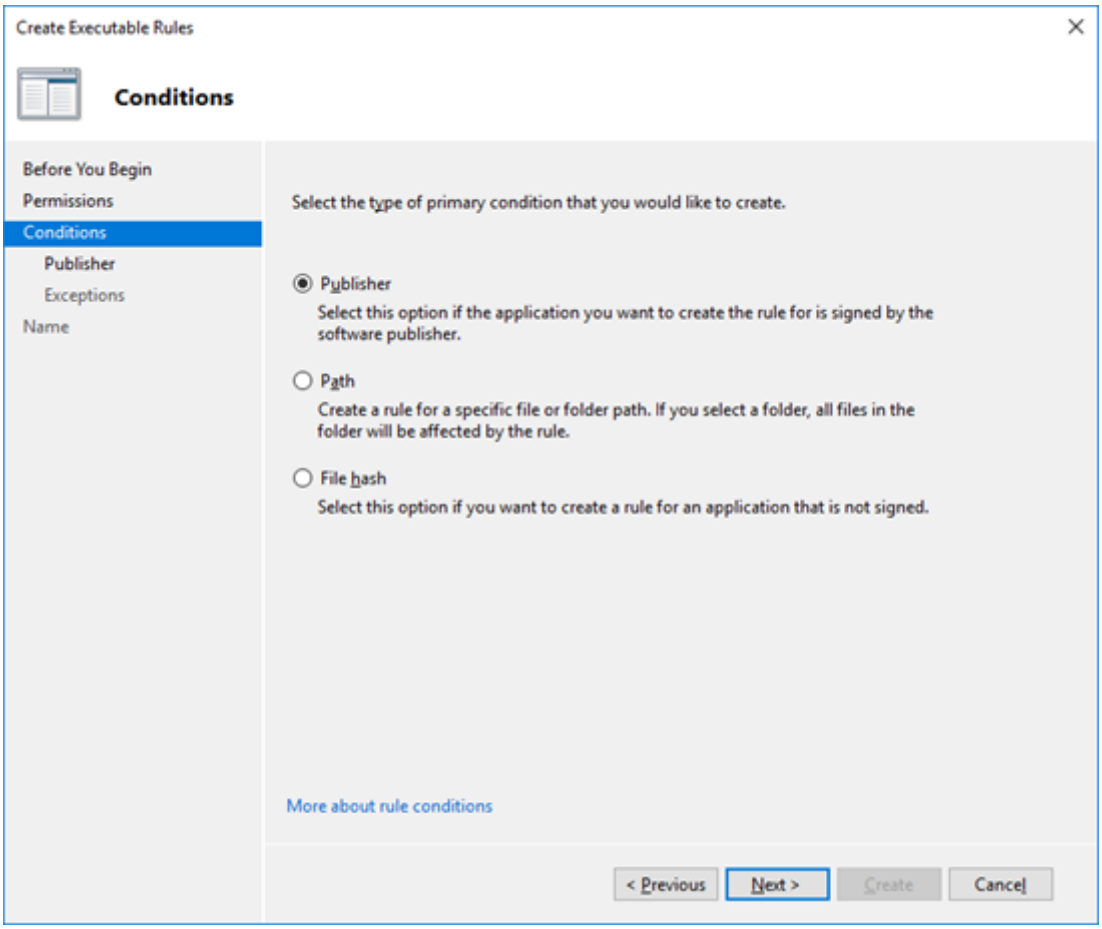

Рисунок 171

В зависимости от типа проверки файлов добавить условие (путь, либо хэш, либо сертификат).

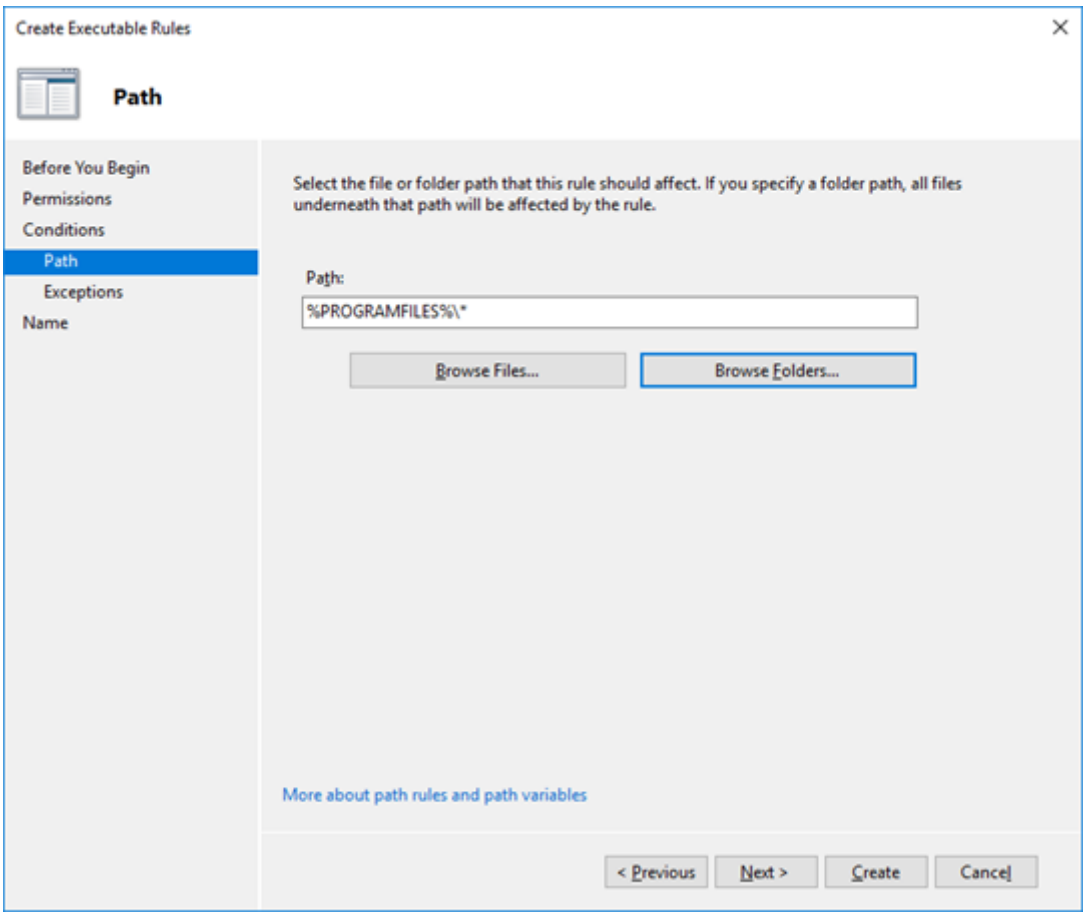

Рисунок 172

При необходимости добавить исключение из правила.

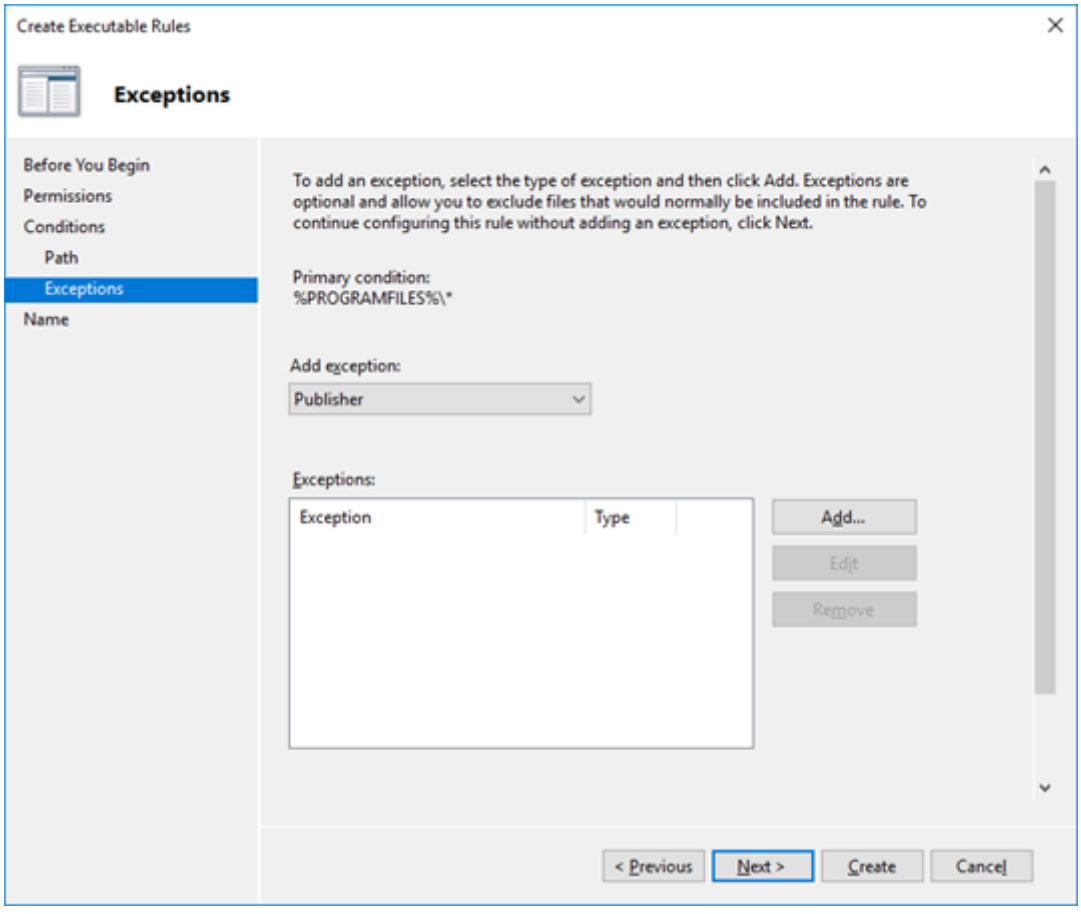

Рисунок 173

```
Добавить имя правила и нажать «Create».
```

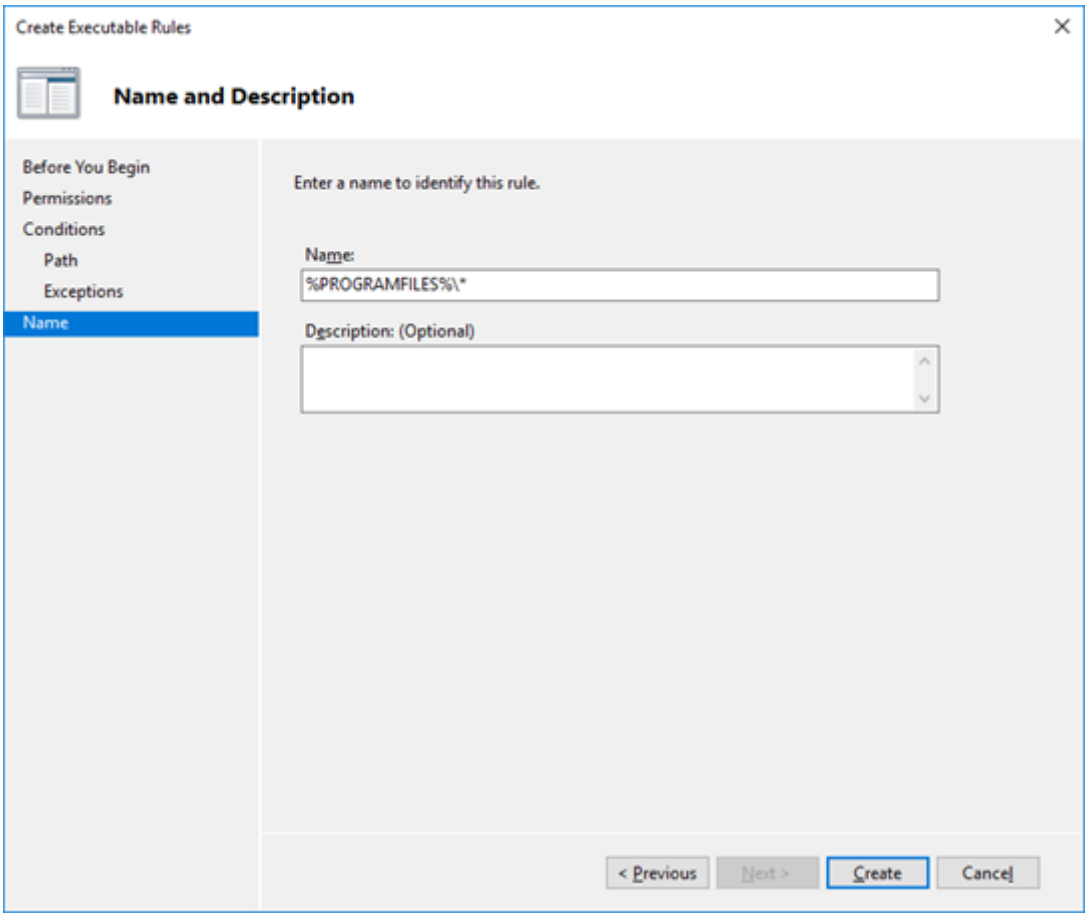

Рисунок 174

5. Применить политику, запустив в командной консоли:

   `gpupdate /force`

6. Проверить наличие событий AppLocker: EventViewer.msc -> Application and Service Log -> Microsoft -> Windows -> AppLocker

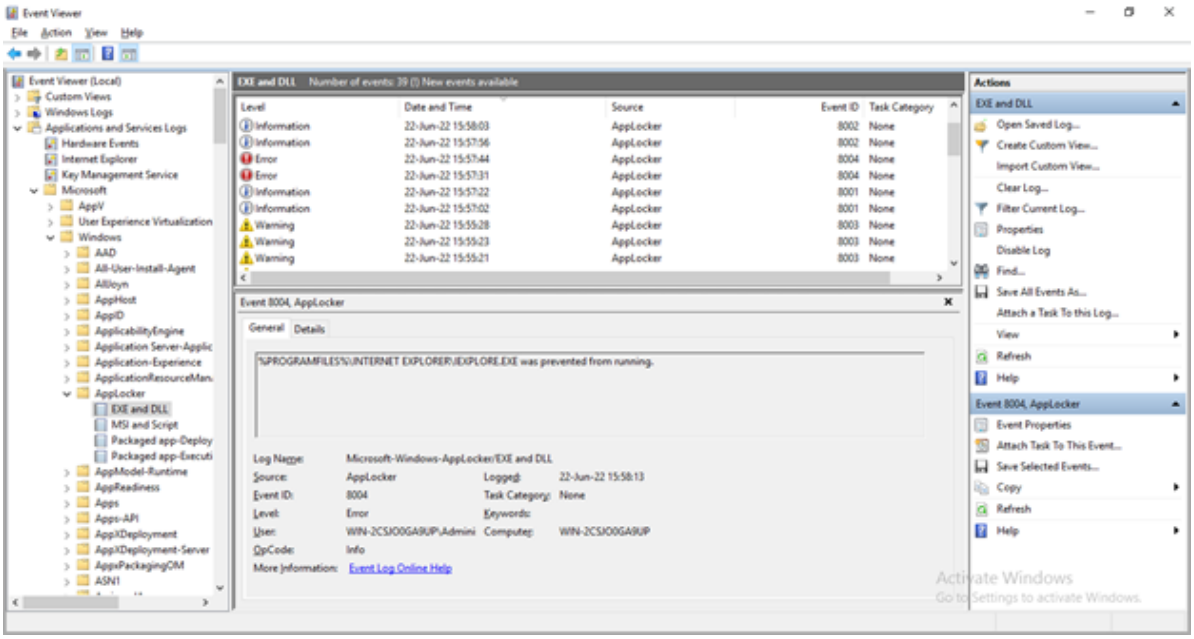

Рисунок 175

7. Добавить в конфигурационный файл лог-коллектора канал сбора событий AppLocker и перезапустить лог-коллектор (на примере Linux).

```
$ vi /opt/pangeoradar/configs/logcollector/config.yaml
.....
channel: ["Security", "System", "Windows PowerShell", "Microsoft-Windows-
AppLocker/EXE and DLL", "Microsoft-Windows-AppLocker/MSI and Script"]
.....
$ systemctl restart pangeoradar-logcollector-agent.service
```

14. Управление лог-коллектором из веб-интерфейса Платформы

Управление экземплярами Лог-коллектора, которые были настроены для централизованного управления с помощью директивы **cluster** в конфигурационном файле, осуществляется в интерфейсе Платформы в разделе «**Администрирование**» — «**Кластер**» на вкладке «**Узлы системы**» пункт «**Проверка**» (см. рисунок 176).

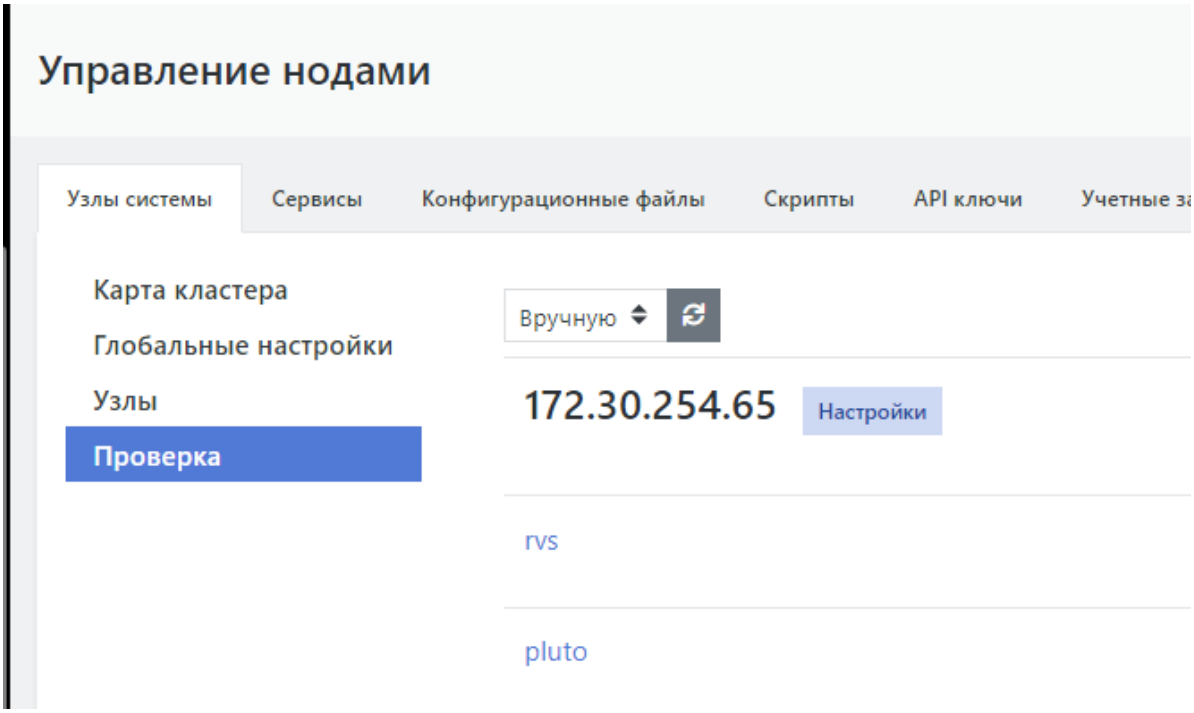

Рисунок 176 - Управление узлами

Необходимо выбрать нужный узел и нажать кнопку «**Настройка**» (см. рисунок 177).

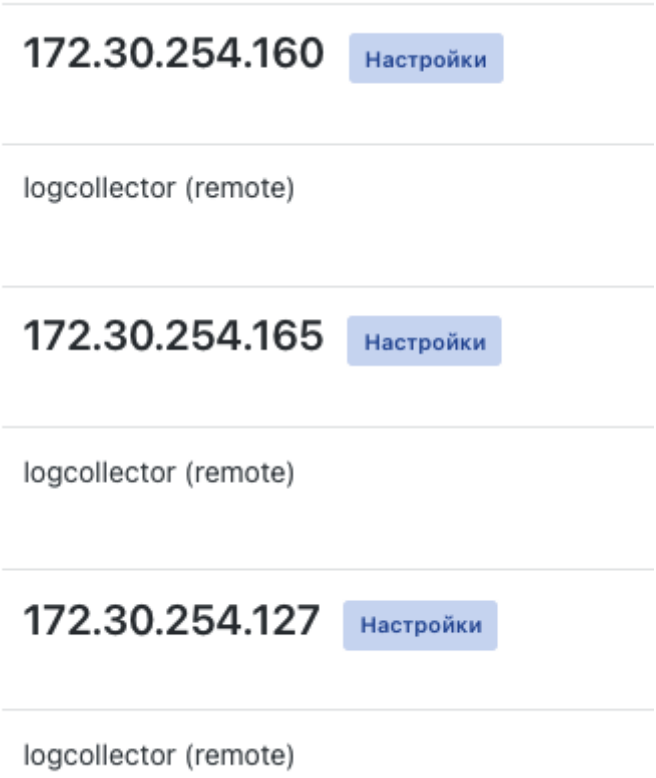

Рисунок 177 - Настройка узла

Откроется страница управления узлом (см. рисунок 178)

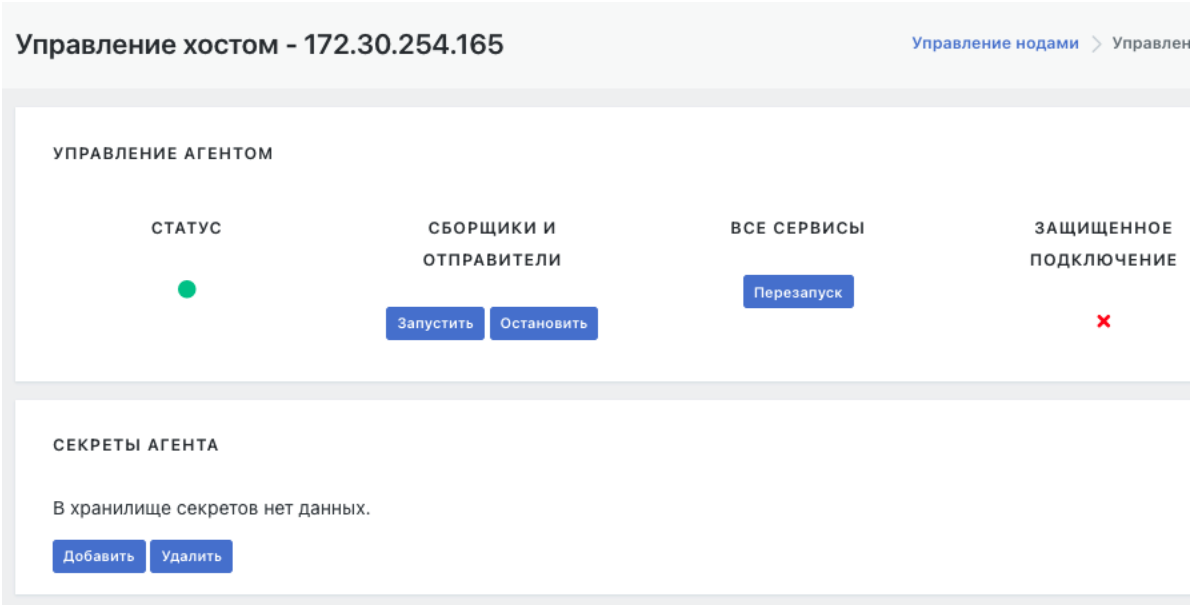

Рисунок 178 - Управление лог-коллектором

На странице Управление лог-коллектором доступны следующие действия:

- Остановка и запуск компонентов сбора и отправки
- Перезапуск всех сервисов
- Загрузка и редактирование конфигурационного файл удаленного лог-коллектора
- Сохранение конфигурационного файл на удаленный лог-коллектор

Для загрузки конфигурационного файла необходимо нажать кнопку «**Загрузить**» (см. рисунок 179)

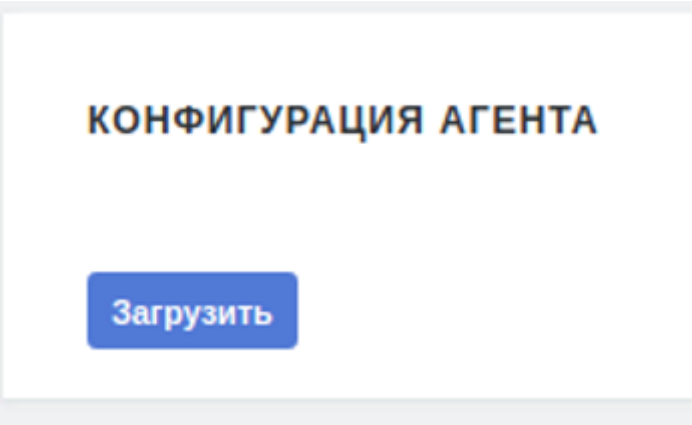

Рисунок 179 - Загрузка конфигурационного файла удаленного лог-коллектора

После нажатия кнопки Загрузить появится возможность редактирования конфигурационного файла на удаленном лог-коллекторе (см. рисунок 180).

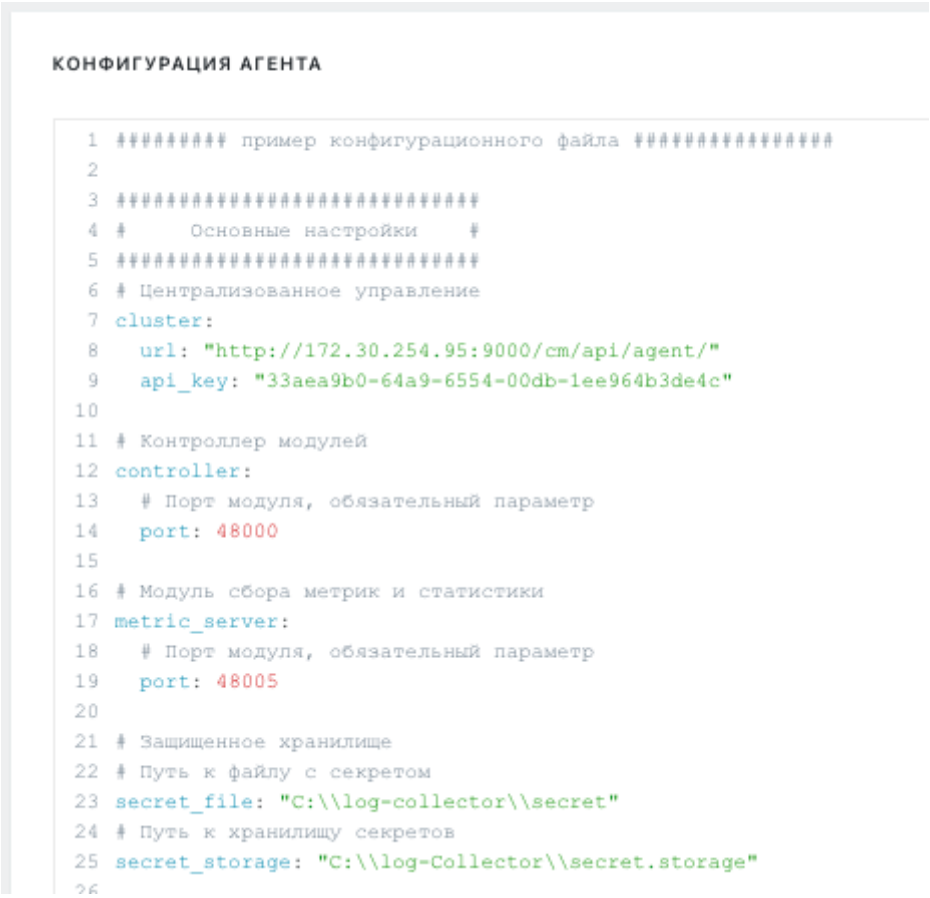

Рисунок 180 - Загруженный конфигурационный файл удаленного лог-коллектора

После внесения изменений в конфигурационный файл его необходимо загрузить на лог-коллектор с помощью нажатия кнопки «**Сохранить**» (см. рисунок 181).

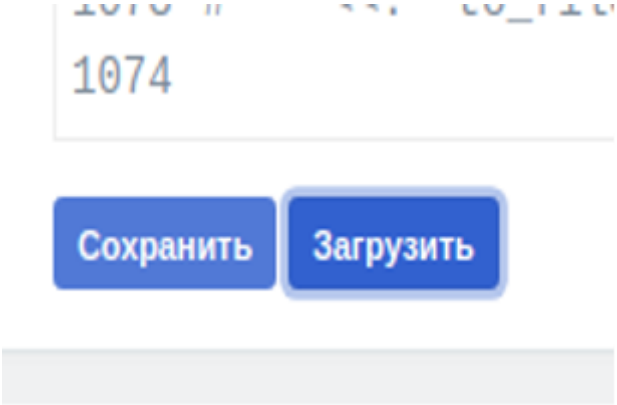

Рисунок 181 - Загруженный конфигурационный файл удаленного лог-коллектора

После чего необходимо перезапустить все сервисы удаленного лог-коллектора, нажав на кнопку «**Перезапуск**» (см. рисунок 178).

15. Пример конфигурационного файла лог-коллектора

```
############################
#     Основные настройки    #
############################
# Централизованное управление
cluster:
  url: "http://<ip адрес Платформы  Радар>:9000/cm/api/agent/"
  api_key: "<ключ API>"

# Путь до файла лицензии
license_path: "./pgr-agent.lic"

# обязательная секция
controller:
  # порт модуля, обязательный параметр
  port: 48000

# обязательная секция
metric_server:
  # порт модуля, обязательный параметр
  port: 48005
  log_level: "ERROR"

# путь к файлу с секретом
secret_file: "secret"
# путь к хранилищу секретов
secret_storage: "secret.storage"

# обязательная секция
api_server:
  # хост на котором будем слушать http сервер
  address: ""
  # порт на котором будем слушать http сервер, обязательный параметр
  port: 8080
  # таймаут чтения(получение запроса), обязательный параметр
```

```
  read_timeout: 60
  # таймаут записи(отправка запроса), обязательный параметр
  write_timeout: 60
  # время ожидания окончания обработки запроса при получении сигнала на остановку
приложения, обязательный параметр
  wait: 5
  # включение https
  enable_tls: false
  # путь для файла сертификатов, если enable_tls: false параметр не обязательный
  cert_file: "certs/server.crt"
  # путь для файла ключей, если enable_tls: false параметр не обязательный
  key_file: "certs/server.key"
  # пароль для расшифровки файла ключей, если не указан считаем что файл не
зашифрован
  cert_key_pass: ""
  # включение проверки клиентского сертификата, обязательный параметр
  require_client_cert: true
  # путь до корневого сертификата, обязательный параметр
  ca_file: "certs/ca.crt"
  # уровень логирования, если не указан используется указанный в модуле
журналирования
  log_level: "ERROR"

# обязательная секция
journal:
  # порт модуля, обязательный параметр
  port: 48003
  # Дефолтный уровень логирования. Возможные значения - DEBUG, INFO, WARN, ERROR.
Обязательный параметр
  log_level: "WARN"
  # путь к файлу логов, обязательный параметр
  log_path: "journal.log"
  # порог ротации файла логов, указывается в мегабайтах, обязательны параметр
  rotation_size: 10
  # порог количества файлов истории, если не указано файлы удаляться не будут
  max_backups: 10
  # максимальное количество дней для хранения старых файлов журнала на основе
метки времени, если не указано файлы удаляться не будут
  max_age: 28 #in days

# outputs kafka
kafka: &kafka
  # названия модуля, отображается в логах и метриках, уникальный обязательный
параметр
  id: "kafka1"
  # включение проверки сертификата (default: false)
  require_cert: false
  # включение ssl (default: false)
  ssl_enable: false
  # путь для файла сертификатов, если ssl_enable: false параметр не обязательный
  cert_file: "certs/server.crt"
  # путь для файла ключей, если ssl_enable: false параметр не обязательный
  key_file: "certs/server.key"
  # пароль для расшифровки файла ключей, если не указан считаем что файл не
зашифрован
```

```
  cert_key_pass: ""
  # путь до корневого сертификата, если ssl_enable: false параметр не
обязательный
  ca_file: "certs/ca.crt"
  # таймауту отправки события в секундах, обязательный параметр
  timeout: 10
  # топик в который попадет событие, обязательный параметр
  topic: "foo"
  # уровень логирования, если не указан используется уровень модуля
журналирования
  log_level: "DEBUG"
  # kafka брокеры, обязательный параметр
  brokers:
    - "localhost:19092"
    - "localhost:29092"
    - "localhost:39092"
  # максимальное количество сообщений в буфере
  #queue_length_limit: 3
  # максимальное время жизни событий в очереди. в секундах
  #queue_time_limit: 3

# вывод в файл
out_file: &out_file
  # названия модуля, отображается в логах и метриках, уникальный обязательный
параметр
  id: "file_outputs"
  # путь до файла куда будут записываться события, обязательный параметр
  file: "test_file.txt"
  # порог ротации в мегабайтах, если указан ноль ил не указан совсем ротация не
происходит (default: 0)
  rotation_size: 0
  # уровень логирования, если не указан используется уровень модуля
журналирования
  log_level: "INFO"

# отправка по протоколу udp
udp_sender: &udp_sender
  # названия модуля, отображается в логах и метриках, уникальный обязательный
параметр
  id: "udp_sender"
  # адрес куда отправлять события (default: "0.0.0.0")
  target_host: "0.0.0.0"
  # порт, на который отправлять события куда отправлять события, обязательный
параметр
  port: 15483
  # размер буфера для отправки, если не указан или равен нулю используется
системное значение (default: 0)
  sock_buf_size: 0
  # уровень логирования, если не указан используется уровень модуля
журналирования
  log_level: "INFO"

# outputs tcp
tcp_sender: &tcp_sender
```

```
  # названия модуля, отображается в логах и метриках, уникальный обязательный
параметр
  id: "tcp_sender"
  # адрес куда отправлять события, обязательный параметр (default: "0.0.0.0")
  target_host: "0.0.0.0"
  #  порт куда отправлять события, обязательный параметр
  port: 15481
  # Включение batch режима (default: false)
  batch_mode_enable: false
  # период отправки пакета в секунундах при включенном batch режиме (default: 5)
  batch_flush_interval: 5
  # количество сообщений которые попадут в пакет при включенном batch режиме
(default: 500)
  batch_flush_limit: 500
  # включение сжатия, включение при выключенном batch режиме ощутимо замедляет
отправку (default: false)
  ssl_compression: false
  # включение проверки сертификата (default: false)
  require_cert: false
  # включение ssl (default: false)
  ssl_enable: false
  # путь для файла сертификатов, если enable_tls: false параметр не обязательный
  cert_file: "certs/server.crt"
  # путь для файла ключей, если enable_tls: false параметр не обязательный
  key_file: "certs/server.key"
  # пароль для расшифровки файла ключей, если не указан считаем что файл не
зашифрован
  cert_key_pass: ""
  # путь до корневого сертификата, если enable_tls: false не обязательный
параметр
  ca_file: "certs/ca.crt"
  # уровень логирования, если не указан используется уровень модуля
журналирования
  log_level: "DEBUG"
  # максимальное количество сообщений в буфере
  #queue_length_limit: 3
  # максимальное время жизни событий в очереди. в секундах
  #queue_time_limit: 3

event_log_settings: &event_log
  # названия модуля, отображается в логах и метриках, уникальный обязательный
параметр
  id: "test_event_log"
  # Имя канала, используется если не указан путь к файлу
  channel: ['Application']
  # Запрос описывающий тип получаемого события. есть возможность указать
  # XPath 1.0 или структурированный XML запрос. Если XPath содержит более 20
параметров, следует
  # использовать структурированный XML запрос. Чтобы получить все параметры
укажите "*"
  query: "*"
  # Полный путь к лог файлу
  # Поддерживаемые форматы: .evt, .evtx, .etl
  file:
  # Размер запроса
```

```
  batch_size: 31
  #  Таймаут запроса в секундах
  timeout: 3
  # интервал между запуском запроса в секундах (default: 1)
  poll_interval: 1
  # Чтение с последней сохраненной позиции, (default: false)
  read_from_last: false
  # Конвертировать SID в имя.
  resolve_sid: true
  # уровень логирования, если не указан используется уровень модуля
журналирования
  log_level: "INFO"
  # количество параллельных воркеров (по умолчанию 1)
  worker_count: 1
  # Параметры удаленного подключения
  remote:
    # Включение удаленного соединения
    enabled: false
    # Имя пользователя, обязательно если enabled: true
    user: ""
    # Пароль пользователя, обязательно если enabled: true
    password: ""
    # Домен пользователя
    domain: ""
    # Адрес удаленного сервера
    remote_servers: ["localhost"]
    # Доступные методы авторизации: Negotiate, Kerberos, NTLM
    auth_method: "Negotiate"
  # фильтрация по полям события, регулярные выражения
  filters:
    # Время
    # формат 2020-08-13 10:02:55.9689259 +0000 UTC
    created: ''
    # Числовые фильтры
    # Пример для числовых фильтров - ^([5-9]\d|\d{3,})$
    event_id: ''
    qualifiers: ''
    record_id: ''
    process_id: ''
    thread_id: ''
    version: ''
    # Строковые фильтры
    # пример: DESKTOP-IDCMV6G
    computer_name: ''
    msg: ''
    # Возможные значения: Information, Warning, Error
    level_text: ''
    # Пример: Service State Event
    task_text: ''
    # Пример: ServiceShutdown
    opcode_text: ''
    # Пример: System
    channel_text: ''
    # Пример: System
    provider_text: 'System'
```

```
  # Опции смены кодировки
  encoding:
    # Использовать кодировку в UTF-8
    change_to_utf8: false
    # Кодировка оригинала
    original_encoding: "cp1251"

# postgres sql connection string example
# Driver={PostgreSQL};Server=IP
address;Port=5432;Database=myDataBase;Uid=myUsername;Pwd=myPassword;
# mssql connection string example
# Driver={ODBC Driver 17 for SQL
Server};Server=myServerAddress;Database=myDataBase;UID=myUsername;PWD=myPassword
;
# Driver={ODBC Driver 17 for SQL
Server};Server=myServerAddress;Database=myDataBase;Trusted_Connection=yes;
# Oracle
# DRIVER={Oracle ODBC Driver};UID=Kotzwinkle;PWD=whatever;DBQ=instl_alias;DBA=W;
odbctest: &odbctest
  # названия модуля, отображается в логах и метриках, уникальный обязательный
параметр
  id: "odbc_test"
  # Категория к которой относится данный input
  categories: "file"
  # интервал между запуском запроса в секундах (default: 1)
  poll_interval: 1
  # Чтение с последней сохраненной позиции (default: false)
  read_from_last: false
  # Строка подключения, обязательный параметр
  connection_string: "server=localhost;port=3306;driver=MySQL ODBC 8.0
Driver;database=pangeo;user=root;password=example;"
  # SQL запрос, обязательный параметр
  sql: >
    SELECT id, message
    FROM pangeo.logs_table WHERE id > ?;
  # Поле, которое будет использоваться как закладка для сохранения позиции,
обязательный параметр
  # Поле должно быть целочисленным
  # Поле должно быть указано в операторе SELECT
  bookmark_field: "id"
  # Опции смены кодировки
  encoding:
    # Использовать кодировку в UTF-8
    change_to_utf8: false
    # Кодировка оригинала
    original_encoding: "cp1251"

wmi_settings: &wmi_settings
  # названия модуля, отображается в логах и метриках, уникальный обязательный
параметр
  id: "test_wmi"
  # интервал между запуском запроса в секундах (default: 1)
  poll_interval: 1
  # Список серверов к которым уйдет wmi запрос, обязательный параметр
  remote_servers:
```

```
    - "localhost"
  #  Имя пользователя, обязательно если это не локальный сбор
  user:
  #  Пароль пользователя, обязательно если это не локальный сбор
  password:
  # Чтение с последней сохраненной позиции, (default: false)
  read_from_last: false
  # уровень логирования, если не указан используется уровень модуля
журналирования
  log_level: "DEBUG"
  # изменение кодировки входящего события, может быть прописана у любого
коллектора
  encoding:
    # включение изменения кодировки
    change_to_utf8: false
    # оригинальная кодировка события, если оставить пустым, произойдет попытка
определить кодировку
    # нет 100% гарантии определения
    original_encoding: "cp1251"
  # Собирать события начиная с заданного момента
  start_from_date: "2022-03-24T00:00:00+03:00"
  # Список журналов, из которых собираются события (Application, System и т.п.).
Если пустой или не указан, собираются все события
  logfiles: ["Application"]
  # блэклист фильтры по полям события, используются регулярные выражения -
https://wiki.andersenlab.com/pages/viewpage.action?pageId=153062062
  wmi_filters:
    # числовые поля
    category: '0+'
    event_code: ''
    event_identifier: ''
    event_type: ''
    record_number: ''
    # строковые поля
    computer_name: ''
    message: ''
    source_name: ''
    type: ''
    user: ''
    time_generated: ''
    time_written: ''

etw_settings: &etw
  # Provider name or guid.
  # GUID should be in format "{9E814AAD-3204-11D2-9A82-006008A86939}".
  id: "etw1"
  provider: "Windows Kernel Trace"
  kernel_args: [ "ALPC", "CSWITCH", "DBGPRINT", "DISK_FILE_IO", "DISK_IO",
"DISK_IO_INIT", "DISPATCHER",
                 "DPC", "DRIVER", "FILE_IO", "FILE_IO_INIT", "IMAGE_LOAD",
"INTERRUPT", "MEMORY_HARD_FAULTS",
                 "MEMORY_PAGE_FAULTS", "NETWORK_TCPIP", "NO_SYSCONFIG",
"PROCESS", "PROCESS_COUNTERS",
                 "PROFILE", "REGISTRY", "SPLIT_IO", "SYSTEMCALL", "THREAD",
"VAMAP", "VIRTUAL_ALLOC" ]
```

```
  provider_level: "Critical"
  log_level: "DEBUG"
  # Опции смены кодировки
  encoding:
    # Использовать кодировку в UTF-8
    change_to_utf8: false
    # Кодировка оригинала
    original_encoding: "cp1251"

opsec_lea: &opsec_lea
  id: "opsec_lea"
  # Директория расположения утилиты lea_client.
  exec_path: "/pangeo_radar/opsec"
  agent_addr: "127.0.0.1"
  agent_port: 48181
  # периодичность проверки наличия новых записей в журналах.
  poll_interval: 1
  #  Сохранение позиции последнего чтения из журнала (сохранение на диск),
возобновление чтения с последней
  #  сохраненной позиции.
  read_from_last: false
  # Сервер для сбора событий.
  remote_server: "192.168.1.254"
  # Порт для аутентификации.
  auth_port: 18184
  # Аутентификация для OPSEC.
  auth_type: "sslca"
  # Параметры авторизации.
  opsec_sic_name: "CN=SyslogClient,O=sms.local.gc95e2"
  opsec_sslca_file: "/home/user/opsec.p12"
  opsec_entity_sic_name: "CN=cp_mgmt,O=sms.local.gc95e2"
  opsec_sic_policy_file: ""
  #  Название собираемого журнала.
  log_filename: "fw.log"
  log_level: "DEBUG"
   # формат отправки сообщения - как есть(raw), с обогащением(json)
  format: "json"

sshtest: &sshtest
  # названия модуля, отображается в логах и метриках, уникальный обязательный
параметр
  id: "sshtest"
  # имя пользователя для удаленного подключения, обязательный параметр
  user: "anduser"
  # список хостов для подключения, обязательный параметр
  remote_servers: ["127.0.0.1"]
  # порт для подключения (default: 22)
  port: 22
  # путь к файлу с ssh ключами, обязательный параметр
  rsa: "~/.ssh/id_rsa"
  # пароль от файла с ключами
  password: ""
  # команда для выполнения по ssh, обязательный параметр
  command: "tail -F -n +$$$line$$$ /opt/pangeo/test.log"
```

```
  # если установлено - файл будет читаться с последней позиции в следующем тике
или после перезапуска (default: false)
  read_from_last: false
  # интервал между выполнением команд(в секундах)
  ticker: 30
  # уровень логирования, если не указан используется уровень модуля
журналирования
  log_level: "DEBUG"
  # формат отправки сообщения - как есть(raw), с обогащением(json)
  format: "json"
  # Опции смены кодировки
  encoding:
    # Использовать кодировку в UTF-8
    change_to_utf8: false
    # Кодировка оригинала
    original_encoding: "cp1251"

smb: &smb
  # названия модуля, отображается в логах и метриках, уникальный обязательный
параметр
  id: "smb1"
  # аддреса подключения, обязательный параметр
  remote_servers: ["pdc.pangeo.test"]
  # порт подключения (default: 445)
  port: 445
  # SMB share. sharename должен соответсвовать формату `<share>` или `\\<server>\
<share>`, обязательный параметр
  share: "\\\\pdc.pangeo.test\\access"
  # домен
  domain: "pangeo.test"
  # имя пользователя, обязательный параметр
  user: "test1"
  # пароль, обязательный параметр
  password: "1qaz2WSX"

  # настройки аутентификации kerberos
  kerberos:
    # включение авторизации kerberos
    enabled: false
    # имя целевого сервиса (service principal name)
    target_spn: "pdc"
    # kerberos realm
    realm: "PANGEO.TEST"
    # путь до конфигурации kerberos
    config_path: "assets/krb5/krb5.conf"

  # интервал между запуском сканирования файлов в секундах (default: 1)
  poll_interval: 1
  # список файлов для чтения, обязательный параметр
  files: [ "hello.txt" ]

  # если установлено - использовать регулярное выражение для поиска файлов
  using_regexp: false
  # начальный каталог для поиска файлов
  regexp_starting_dir: "."
```

```
  # регулярное выражение для поиска файлов
  regexp_expression: "^hello1.txt$"
  # интервал проверки файлов (в секундах) в дереве каталогов (default: 2)
  dir_check_interval: 2

  # если установлено - файл будет читаться с последней позиции в следующем тике
или после перезапуска
  read_from_last: true
  # уровень логирования, если не указан используется уровень модуля
журналирования
  log_level: "DEBUG"
  # формат отправки сообщения - как есть(raw), с обогащением(json)
  format: "json"
  # Опции смены кодировки
  encoding:
    # Использовать кодировку в UTF-8
    change_to_utf8: false
    # Кодировка оригинала
    original_encoding: "cp1251"

ftptest: &ftptest
  # названия модуля, отображается в логах и метриках, уникальный обязательный
параметр
  id: "ftptest"
  # аддреса для ftp запросов, обязательный параметр
  remote_servers: ["localhost"]
  # порт для ftp запросов, обязательный параметр
  port: 21
  # ftp пользователь, обязательный параметр
  user: "testuser"
  # ftp пароль, обязательный параметр
  password: "testpass"

  # интервал между сканированием файла в секундах (default: 1)
  poll_interval: 1
  # список файлов для чтения, обязательный параметр
  files: ["apache_logs"]

  # если установлено - использовать регулярное выражение для поиска файлов
  using_regexp: true
  # начальный каталог для поиска файлов
  regexp_starting_dir: "."
  # регулярное выражение для поиска файлов
  regexp_expression: "^.*_logs$"
  # интервал проверки файлов (в секундах) в дереве каталогов (default: 2)
  dir_check_interval: 2

  # если установлено - файл будет читаться с последней позиции в следующем тике
или после перезапуска
  read_from_last: true

  # уровень логирования, если не указан используется уровень модуля
журналирования
  log_level: "DEBUG"
  # формат отправки сообщения - как есть(raw), с обогащением(json)
```

```
  format: "json"
  # Опции смены кодировки
  encoding:
    # Использовать кодировку в UTF-8
    change_to_utf8: false
    # Кодировка оригинала
    original_encoding: "cp1251"

sftptest: &sftptest
  # названия модуля, отображается в логах и метриках, уникальный обязательный
параметр
  id: "sftptest"
  # адреса для sftp запросов, обязательный параметр
  remote_servers: ["localhost"]
  # порт для sftp запросов, обязательный параметр
  port: 22
  # пользователь ssh, обязательный параметр
  user: "foo"
  # пароль ssh, обязательный параметр
  password: "pass"

  # интервал между сканированием файла в секундах (default: 1)
  poll_interval: 1
  # список файлов для чтения, обязательный параметр
  files: ["/upload/apache_logs"]

  # если установлено - использовать регулярное выражение для поиска файлов
  using_regexp: true
  # начальный каталог для поиска файлов
  regexp_starting_dir: "upload"
  # регулярное выражение для поиска файлов
  regexp_expression: "^.*_logs$"
  # интервал проверки файлов (в секундах) в дереве каталогов (default: 2)
  dir_check_interval: 2

  # если установлено - файл будет читаться с последней позиции в следующем тике
или после перезапуска
  read_from_last: true
  # уровень логирования, если не указан используется уровень модуля
журналирования
  log_level: "DEBUG"
  # формат отправки сообщения - как есть(raw), с обогащением(json)
  format: "json"
  # Опции смены кодировки
  encoding:
    # Использовать кодировку в UTF-8
    change_to_utf8: false
    # Кодировка оригинала
    original_encoding: "cp1251"

nf_receiver: &nftest
  # названия модуля, отображается в логах и метриках, уникальный обязательный
параметр
  id: "netflow"
  # Хост на каком запустится сервер (default: localhost)
```

```
  host: "localhost"
  # Порт на каком запустится сервер (обязательное)
  port: 15487
  # Уровень сообщений в логах, могут быть значения - DEBUG, INFO, WARN, ERROR
  log_level: "INFO"
  # Размер буфера сообщений (если не задано то берется из SO_RCVBUF)
  sock_buf_size: 0

tcp_receiver: &tcptest
  # названия модуля, отображается в логах и метриках, уникальный обязательный
параметр
  id: "tcp"
  # Хост на каком запустится сервер (default: localhost)
  host: "localhost"
  # Порт на каком запустится сервер (обязательное)
  port: 15486
  # Уровень сообщений в логах, могут быть значения - DEBUG, INFO, WARN, ERROR
  log_level: "INFO"
  # Включение TLS соединения на сервере (default: false)
  enable_tls: false
  # Файл с приватным ключом (обязательное поле при включенном TLS)
  key_file: "certs/server.key"
  # Файл с сертификатом должно быть подписанным сертификатом CA (обязательное
поле при включенном TLS)
  cert_file: "certs/server.crt"
  # Файл с паролем если сертификат подписывался с паролем
  cert_key_pass: ""
  # Файл с сертификатом CA (обязательное поле при включенном TLS)
  ca_file: "certs/ca.crt"
  # Проверять ли сертификаты клиента (default: false)
  require_client_cert: false
  #Нужна ли распаковка тела запроса, ожидается, что клиент упаковал тело запроса
в архив (default: false)
  compression_enabled: false
  # Количество соединений которые может принять сервер
  connections_limit: 10
  # формат отправки сообщения - как есть(raw), с обогащением(json)
  format: "json"
  # Опции смены кодировки
  encoding:
    # Использовать кодировку в UTF-8
    change_to_utf8: false
    # Кодировка оригинала
    original_encoding: "cp1251"

http_receiver: &httptest
  # названия модуля, отображается в логах и метриках, уникальный обязательный
параметр
  id: "http"
  # Хост на каком запустится сервер (default: localhost)
  host: "localhost"
  # Порт на каком запустится сервер (обязательное)
  port: 15484
  # Уровень сообщений в логах, могут быть значения - DEBUG, INFO, WARN, ERROR
  log_level: "INFO"
```

```
  # Включение TLS соединения на сервере (default: false)
  enable_tls: false
  # Файл с приватным ключом (обязательное поле при включенном TLS)
  key_file: "certs/server.key"
  # Файл с сертификатом должно быть подписанным сертификатом CA (обязательное
поле при включенном TLS)
  cert_file: "certs/server.crt"
  # Файл с паролем если сертификат подписывался с паролем
  cert_key_pass: ""
  # Файл с сертификатом CA (обязательное поле при включенном TLS)
  ca_file: "certs/ca.crt"
  # Проверять ли сертификаты клиента (default: false)
  require_client_cert: false
  # Количество соединений которые может принять сервер
  connections_limit: 10
  # формат отправки сообщения - как есть(raw), с обогащением(json)
  format: "json"
  # Опции смены кодировки
  encoding:
    # Использовать кодировку в UTF-8
    change_to_utf8: false
    # Кодировка оригинала
    original_encoding: "cp1251"

udp_receiver: &udptest
  # названия модуля, отображается в логах и метриках, уникальный обязательный
параметр
  id: "udp"
  # Хост на каком запустится сервер (default: localhost)
  host: "localhost"
  # Порт на каком запустится сервер (обязательное)
  port: 15485
  # Уровень сообщений в логах, могут быть значения - DEBUG, INFO, WARN, ERROR
  log_level: "INFO"
  # Размер буфера сообщений (если незаданные то берется из SO_RCVBUF)
  sock_buf_size: 0
  # формат отправки сообщения - как есть(raw), с обогащением(json)
  format: "json"
  # Опции смены кодировки
  encoding:
    # Использовать кодировку в UTF-8
    change_to_utf8: false
    # Кодировка оригинала
    original_encoding: "cp1251"

http_collector: &httpcollector
  # названия модуля, отображается в логах и метриках, уникальный обязательный
параметр
  id: "http"
  # удаленный адрес для вызовов http (обязательное)
  remote_server: "localhost"
  # удаленный порт (default: 80)
  port: 80
  # имя пользователя для базовой авторизации, если пустое, считаем что
авторизация выключена
```

```
  basic_auth_user: ""
  # пароль для базовой авторизации
  basic_auth_password: ""
  # ограничение по времени для запросов, сделанных http-клиентом в секундах
(default: 10)
  timeout: 10
  # уровень логирования, если не указан используется уровень модуля
журналирования
  log_level: "INFO"
  # если установлено - будет использоваться tls клиент
  enable_tls: false
  # путь к .key файлу, обязательно если enable_tls: true
  key_file: "certs/server.key"
  # путь к .crt файлу, обязательно если enable_tls: true
  cert_file: "certs/server.crt"
  # пароль к файлу сертификатов
  cert_key_pass: ""
  # путь к файлу с набором корневых центров сертификации, обязательно если
enable_tls: true
  ca_file: "certs/ca.crt"
  # имя файла для получения по http
  file: ""
  # если установлено - файл будет читаться с последней позиции в следующем тике
или после перезапуска (default: false)
  read_from_last: false
  # интервал между http-вызовами в секундах (default: 3)
  poll_interval: 3
  # формат отправки сообщения - как есть(raw), с обогащением(json)
  format: "json"
  # Опции смены кодировки
  encoding:
    # Использовать кодировку в UTF-8
    change_to_utf8: false
    # Кодировка оригинала
    original_encoding: "cp1251"

file_settings: &filetest
  # названия модуля, отображается в логах и метриках, уникальный обязательный
параметр
  id: "filetest"
  # интервал между чтениями файла, в секундах (default: 1)
  poll_interval: 1
  # список файлов для чтения
  files: ["./assets/apache_logs"]

  # использовать regexp для поиска файлов
  using_regexp: false
  # начальный каталог для поиска файлов
  regexp_starting_dir: "."
  # regexp для поиска файлов
  regexp_expression: "^.*_logs$"
  # интервал поиска файлов в секундах, в дереве каталогов (default: 2)
  dir_check_interval: 2
  # если установлено - файл будет читаться с последней позиции в следующем тике
или после перезапуска
```

```
  read_from_last: true

  # создает file watchers для всех файлов
  enable_watcher: true
  # уровень логирования, если не указан используется уровень модуля
журналирования
  log_level: "DEBUG"
  # формат отправки сообщения - как есть(raw), с обогащением(json)
  format: "json"
  # Опции смены кодировки
  encoding:
    # Использовать кодировку в UTF-8
    change_to_utf8: false
    # Кодировка оригинала
    original_encoding: "cp1251"

external_command: &ectest
  # названия модуля, отображается в логах и метриках, уникальный обязательный
параметр
  id: "ectest"
  # интервал между выполнениями команд (default: 1)
  poll_interval: 1
  # команда bash/cmd
  command: "bash ./assets/bash/shortloop.sh"
  # уровень логирования, если не указан используется уровень модуля
журналирования
  log_level: "DEBUG"
  # Опции смены кодировки
  encoding:
    # Использовать кодировку в UTF-8
    change_to_utf8: false
    # Кодировка оригинала
    original_encoding: "cp1251"
  # формат отправки сообщения - как есть(raw), с обогащением(json)
  format: "json"

snmptraptest: &snmptraptest
  # названия модуля, отображается в логах и метриках, уникальный обязательный
параметр
  id: "snmptraptest"
  # уровень логирования, если не указан используется уровень модуля
журналирования
  log_level: "INFO"
  # адресс snmp менеджера
  host: "localhost"
  # порт для запуска snmp менеджера
  port: 22
  # Принимать только аутентифицированные SNMP v3 Traps
  allow_authenticated_only: false
  # список директорий с .mib файлами для конвертации oid
  # если не указаны, oid будут передаваться в сыром виде
  mib_dirs:
    - dir1
    - dir2
    - dir3
```

```
  # Параметры безопасности
  # методы аутентификации. Возможные значения:
  # - MD5
  # - SHA
  auth_proto: ""
  # методы шифрования. Поддерживается только DES.
  encrypt_proto: ""
  # Имя SNMP пользователя
  user_name: ""
  # Пароль аутентификации. Используется с MD5 или SHA
  authentication_passphrase: ""
  # Пароль шифрования для DES
  privacy_passphrase: ""
  # Используется в SNMPv3 для идентификации сущностей.
  authoritative_engine_id: ""
  # Опции смены кодировки
  encoding:
    # Использовать кодировку в UTF-8
    change_to_utf8: false
    # Кодировка оригинала
    original_encoding: "cp1251"
  # формат отправки сообщения - как есть(raw), с обогащением(json)
  format: "json"

# коллектор для сбора событий по протоколу MS-EVEN6 с Windows Vista и выше
mseven6: &mseven6
  # названия модуля, отображается в логах и метриках, уникальный обязательный
параметр
  id: "test_mseven6"
  # Список источников для сбора событий
  sources:
    -
      # Адрес удаленного сервера, с которого будут собираться события
      host: "192.168.56.6"
      # домен
      domain: ""
      #  Имя пользователя
      user: "user"
        # Пароль пользователя
      password: "0000"
      # Список каналов
      channel: ["Application"]
      # Запрос описывающий тип получаемого события. есть возможность указать
      # XPath 1.0 или структурированный XML запрос. Если XPath содержит более 20
параметров, следует
      # использовать структурированный XML запрос. Чтобы получить все параметры
укажите "*"
      query: "*"
  # Размер запроса
  batch_size: 20
  # интервал между запуском запроса в секундах (default: 1)
  poll_interval: 1
  # Чтение с последней сохраненной позиции, (default: false)
  read_from_last: true
```

```
  # уровень логирования, если не указан используется уровень модуля
журналирования
  log_level: "DEBUG"
  # путь для запуска python (лучше использовать venv, создается командой make
mseven6venv)
  python_path: "./bin/mseven6venv/bin/python"
  # порт для взаимодействия с python сервисом-прослойкой
  python_service_port: 9999
  # Фильтрация по полям события, регулярные выражения (блэклист)
  filters:
    # Время
    # формат 2020-08-13 10:02:55.9689259 +0000 UTC
    created: ''
    # Числовые фильтры
    # Пример для числовых фильтров - ^([5-9]\d|\d{3,})$
    event_id: ''
    qualifiers: ''
    record_id: ''
    process_id: ''
    thread_id: ''
    version: ''
    # Строковые фильтры
    # пример: DESKTOP-IDCMV6G
    computer_name: ''
    msg: ''
    # Возможные значения: Information, Warning, Error
    level_text: ''
    # Пример: Service State Event
    task_text: ''
    # Пример: ServiceShutdown
    opcode_text: ''
    # Пример: System
    channel_text: ''
    # Пример: System
    provider_text: 'System'
  # Опции смены кодировки
  encoding:
    # Использовать кодировку в UTF-8
    change_to_utf8: false
    # Кодировка оригинала
    original_encoding: "cp1251"

# обязательная секция
# Список всех запущеных инпутов
collectors:
  log_level: "DEBUG"
  # opsec_lea коллектор
  opsec_lea:
   - <<: *opsec_lea
  # etw коллектор, работает только на windows
#  etw:
#    - <<: *etw
  # чтение из локального файла
#  files:
#    - <<: *filetest
```

```
  # коллектор выполняющий сторонней командой
#  external_command:
#    - <<: *ectest
  # wmi коллектор, работает только на windows
#  wmi:
#    - <<: *wmi_settings
  # event_log коллектор, работает только на windows vista и старше
#  event_log:
#    - <<: *event_log
  # odbc коллектор
#  odbc:
#    - <<: *odbctest
  # ssh коллектор
#  ssh:
#    - <<: *sshtest
  # smb коллектор
#  smb:
#    - <<: *smb
  # ftp коллектор
#  ftp:
#    - <<: *ftptest
  # sftp коллектор
#  sftp:
#    - <<: *sftptest
  # tcp коллектор, пассивный прием
#  tcp_receiver:
#    - <<: *tcptest
  # udp коллектор, пассивный прием
#  udp_receiver:
#    - <<: *udptest
  # netflow коллектор, пассивный прием
#  nf_receiver:
#    - <<: *nftest
  # http коллектор, пассивный прием
#  http_receiver:
#    - <<: *httptest
  # http коллектор, удаленный сбор событий
#  http_collector:
#    - <<: *httpcollector
  # snmp trap коллектор, пассивный прием
#  snmp_trap:
#    - <<: *snmptraptest
  # коллектор для сбора событий по протоколу MS-EVEN6 с Windows Vista и выше
  # mseven6:
  #   - <<: *mseven6
# тестовый коллектор, каждый период времени ticker генерирует сообщение
# test_loop:
#     # названия модуля, отображается в логах и метриках, уникальный обязательный
параметр
#   - id: "loop_input_1"
#     # период генерации сообщений в секундах
#     ticker: 1
#     log_level: "DEBUG"
  categories_rules:
    # Список категорий допустимых в input
```

```
    - web
    - database
    - log
    - file

# обязательная секция
senders:
  # порт модуля, обязательный параметр
  port: 48001
  # уровень логирования, если не указан используется уровень логирования модуля
журналирования
  log_level: "INFO"
  # отправка в журнал
  # stdout:
  #   # названия модуля, отображается в логах и метриках, уникальный обязательный
параметр
  #   - id: "stdout"
  #     log_level: "DEBUG"
  # запись в файл
  # out_file:
  #   - <<: *out_file
  # отправка в kafka
  # kafka:
  #   - <<: *kafka
  # отправка по протоколу tcp
  tcp:
    - <<: *tcp_sender
  # отправка по протоколу udp
  # udp:
  #   - <<: *udp_sender

##################
#     routes     #
##################
# Настройка маршрутов
# Обязательная секция
route_1: &route_1
  collector_id:
    - "opsec_lea"
  sender_id:
    - "tcp_sender"

# Включение маршрутов
# Обязательная секция
routers:
  - <<: *route_1
```