?????????? ??????? ??? ?????? Cisco ASA Firewall ?? ?????? ???????? ???????

> Информация, приведенная на данной странице, является разработкой команды pre-sales и/или community KUMA и **НЕ** является официальной рекомендацией вендора.

> Данный скрипт добавлен пользователем KUMA Community и предоставляется AS IS без каких-либо гарантий и ответственности.

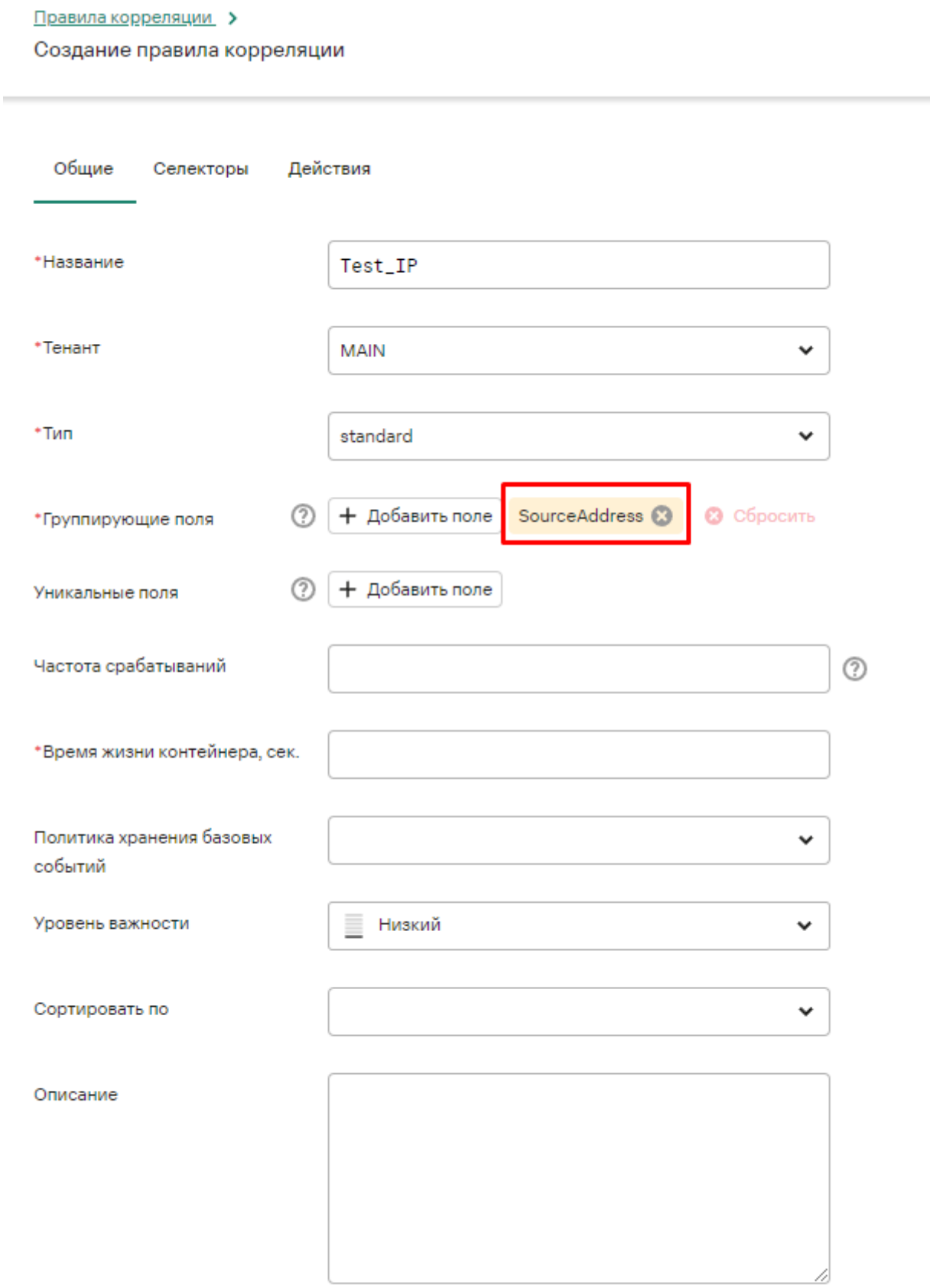
В нашем случае мы при помощи правила корреляции, на основе логов веб-сервера Apache, блокируем входящий трафик от внешних адресов.

Для настройки реагирования средствами Cisco ASA (блокировки IP адресов) (пример, при подключении по SSH) необходимо:

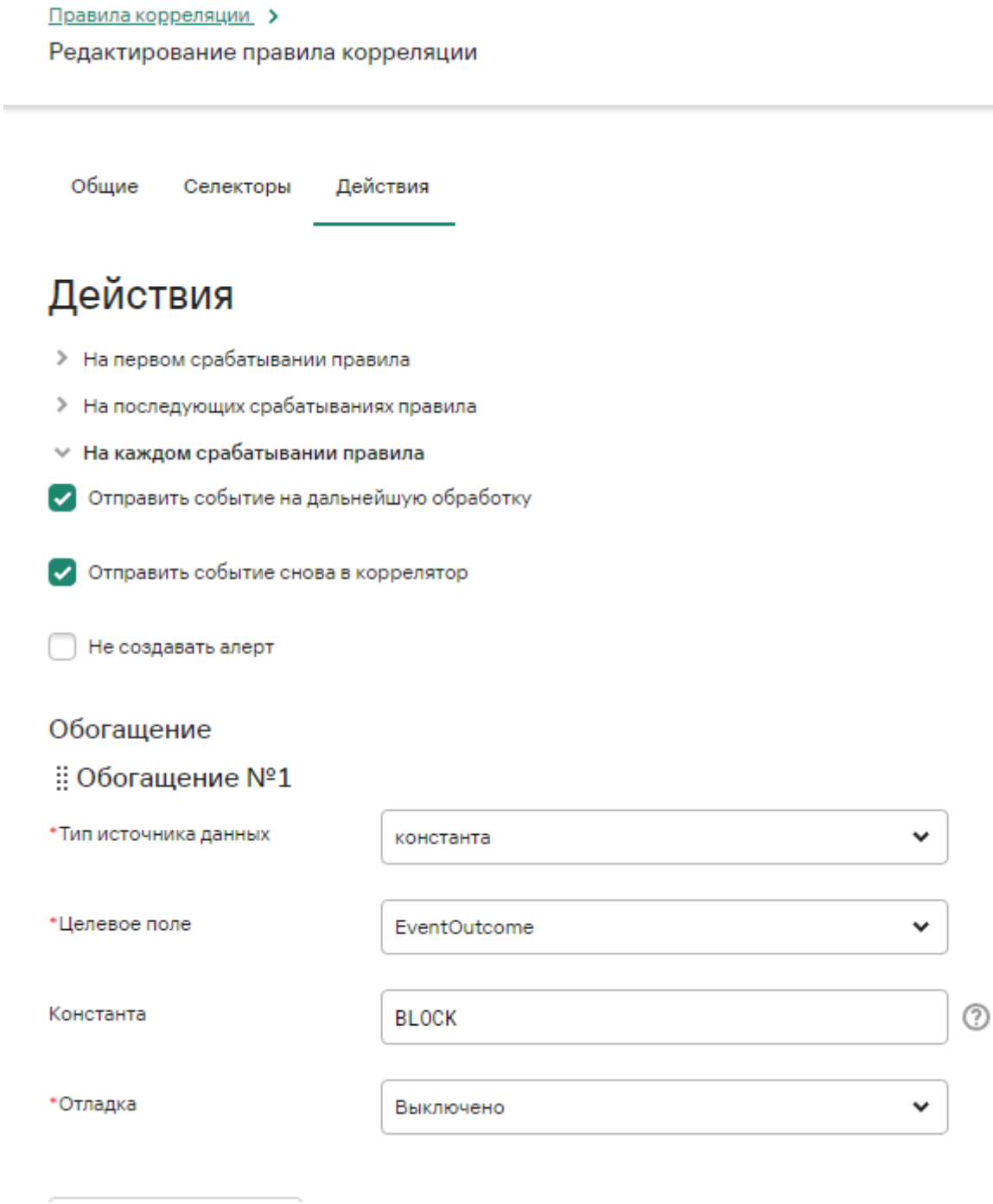
1. Авторизоваться на ASA с привилегиями, позволяющими переходить в режим конфигурирования, отправить команду `conf t`
2. Создать объект, в который наш скрипт будет добавлять адреса (черный список), назовём его BLACKLIST. Создаём командой `object network BLACKLIST`
3. На ASA создаём access-list, который будет блокировать входящие сетевые соединения из интернета от IP находящихся в нашем объекте. Пример: `access-list INTERNET extended deny ip object-group BLACKLIST any`

????????? ?? ??????? KUMA

1. В группирующем поле **правила корреляции** должны находиться целевые поля, которые используются в правилах реагирования, в нашем примере это **sourceAddress**



2. В "селекторы" и "действия" задаём необходимые в данном кейсе параметры. Обязательно добавить обогащение событие EventOutcome (как указано на скриншоте), это ключ (триггер) для следующего этапа по запуску правила реагирования.



3. Размещаем скрипт (находится в папке пресейл-пака) предварительно заменив ключевые данные в скрипте, а именно:ip asa, login, password с правами, необходимыми для добавления в объект BLACK (см. выше, этап настройки ASA)
4. Скрипт помещаем на сервере(-ах) по пути `/opt/kaspersky/kuma/correlator/<id>/scripts/` и предоставляем права пользователю kuma, чтобы служба имела достаточные права для запуска скрипта, командами:
`chown kuma:kuma /opt/kaspersky/kuma/correlator/<id>/scripts/asa.py`
`chmod +x /opt/kaspersky/kuma/correlator/<id>/scripts/asa.py`

5. **Данный этап индивидуален и зависит конкретно от Вашего экземпляра ОС, возможно будут дополнительные ошибки при запуске скрипта, для проверки скрипт можно запускать вручную с сервера и проверять**

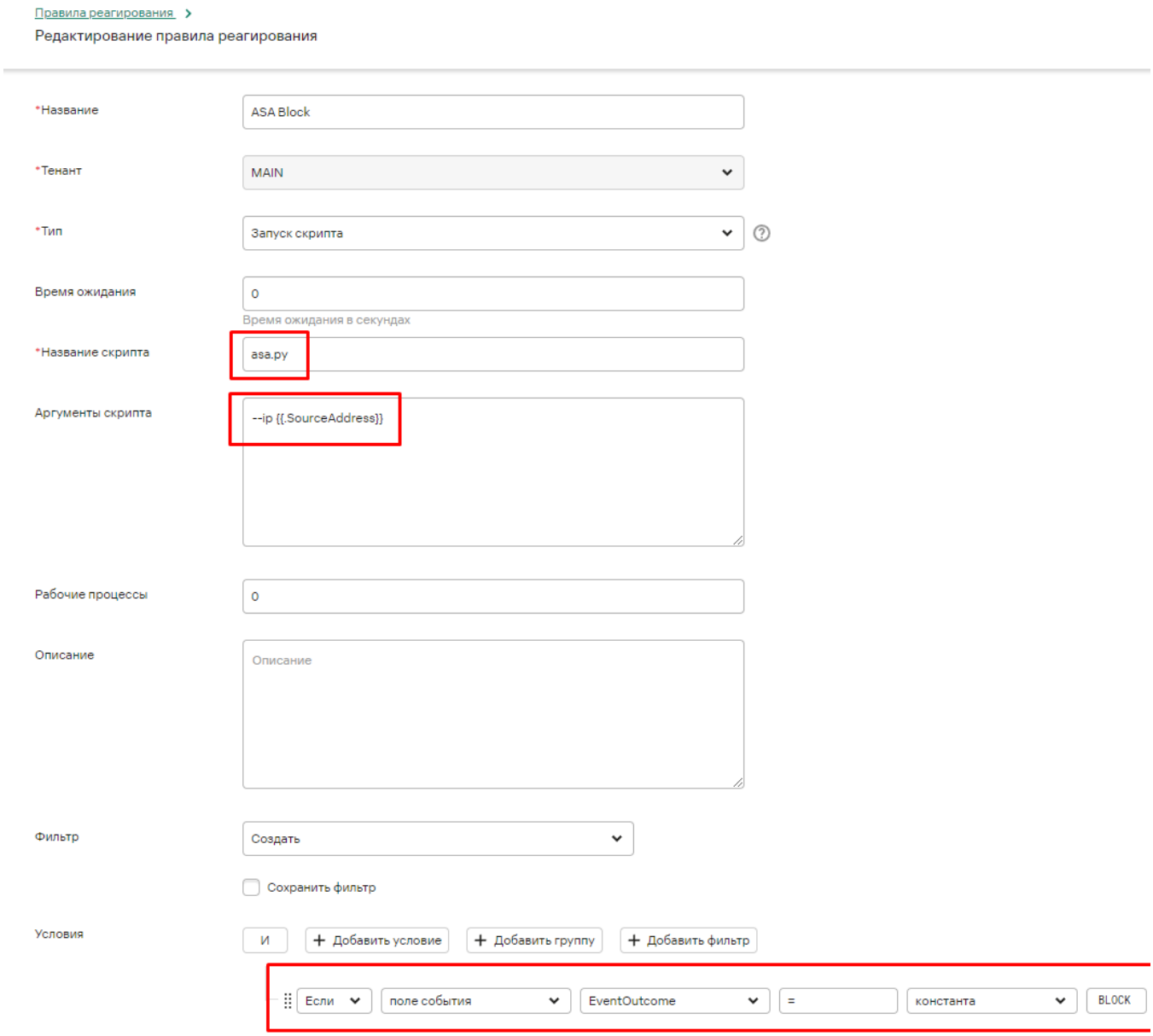
> **работоспособность**

Также на сервер коррелятора в pip3 необходимо до установить следующие библиотеки, для возможности запуска python3.*

threading
paramiko
sys
argparse
subprocess

команда:
`pip3 install threading paramiko sys argparse subprocess`

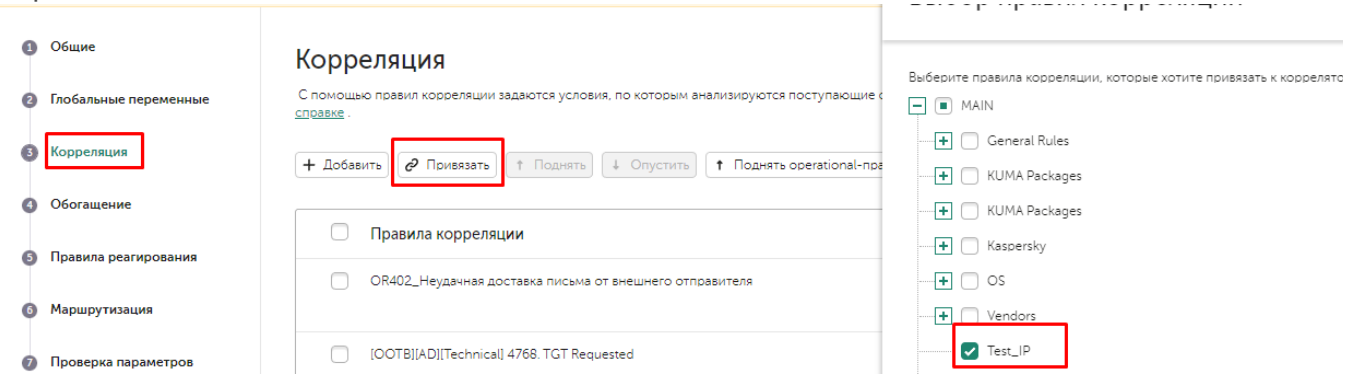
6. Создаём правило реагирования, которое будет непосредственно запускать скрипт на коллекторе (шаг 4)



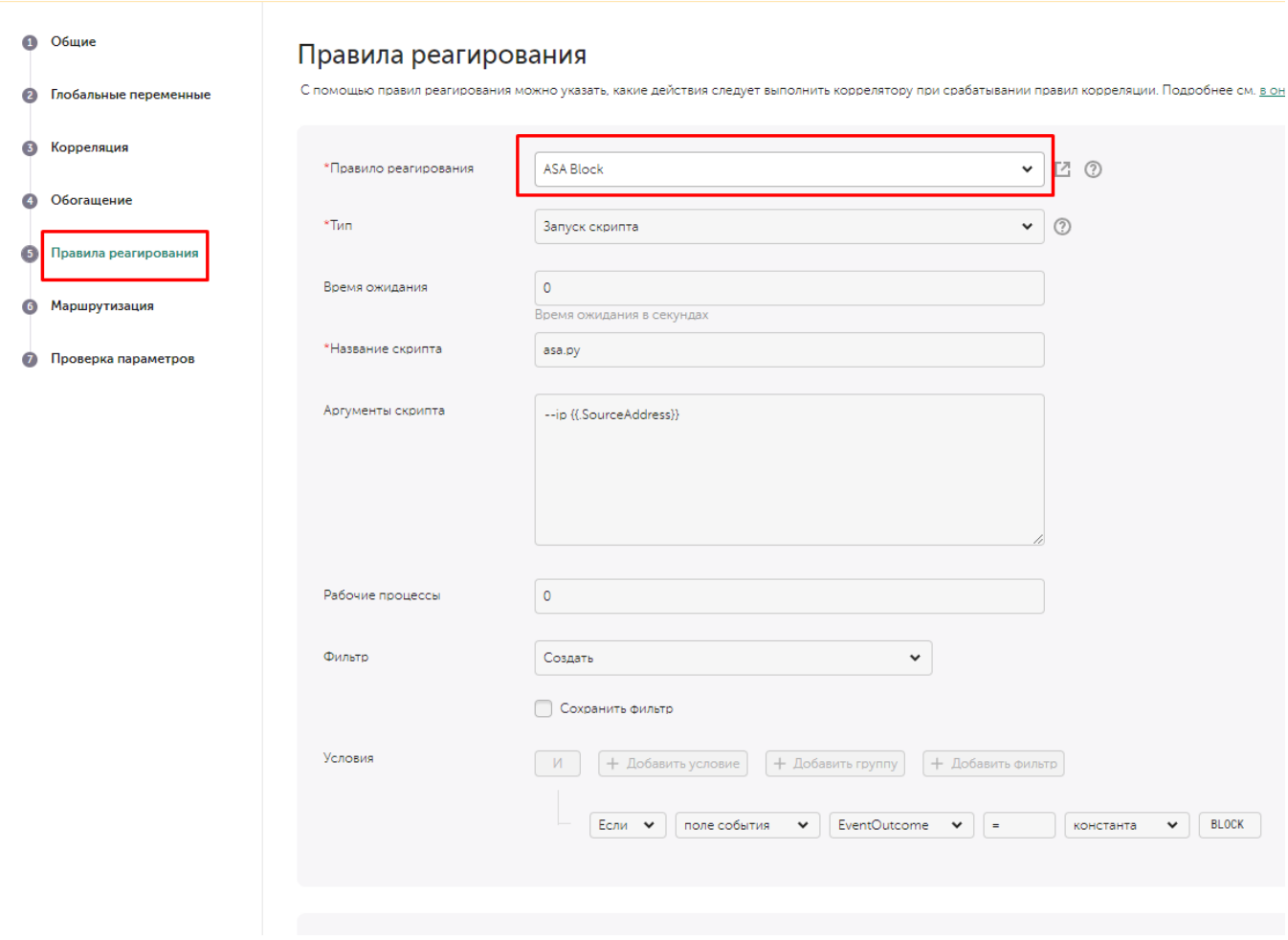
ключевое, задаём Название скрипта, который мы перенесли на сервер коррелятора (шаг 4), также аргументы скрипта, как на скриншоте и ключевое поле EventOutcome = BLOCK (добавляется при срабатывании алерта, при помощи обогащения) (указано

как пример, можно задать списком и другими полями).

7. Осталось привязать новое правило корреляции. Привязываем к коррелятору. Ресурсы -> Корреляторы -> Выбираем наш -> Корреляция, привязать (на скриншоте)



8. Также необходимо здесь же добавить правило реагирования



9. Готово, сохраняем и рестартуем коррелятор (ы)

Revision #9
Created 2023-12-08 17:34:20 UTC
Updated 2024-11-11 10:32:38 UTC by Koala