

# Интеграция с Kaspersky

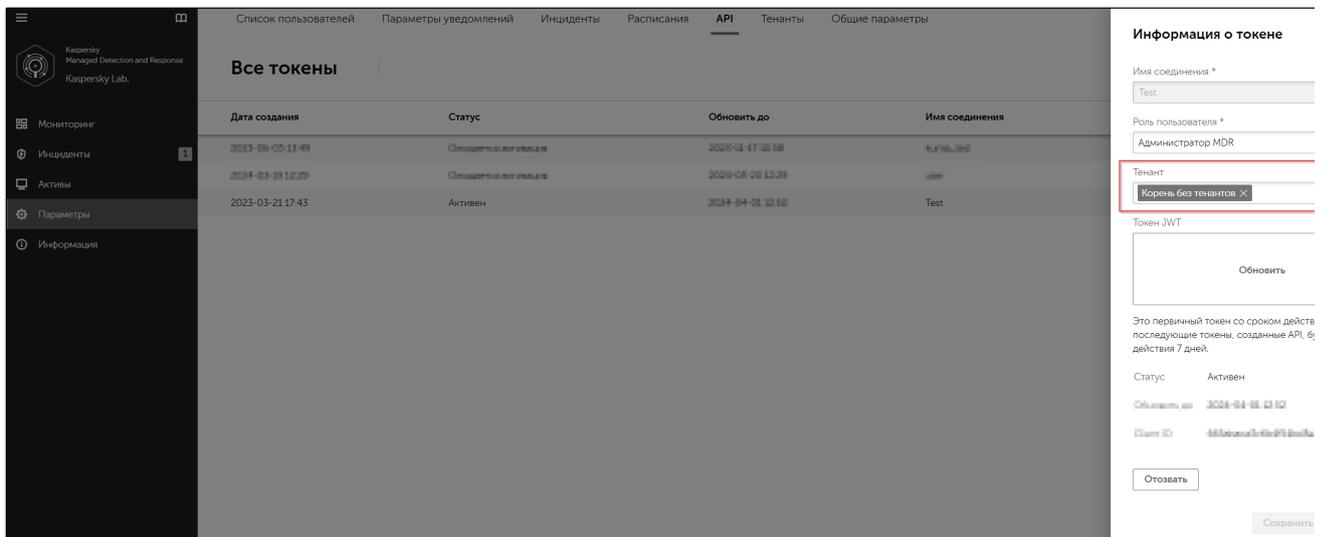
## MDR

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

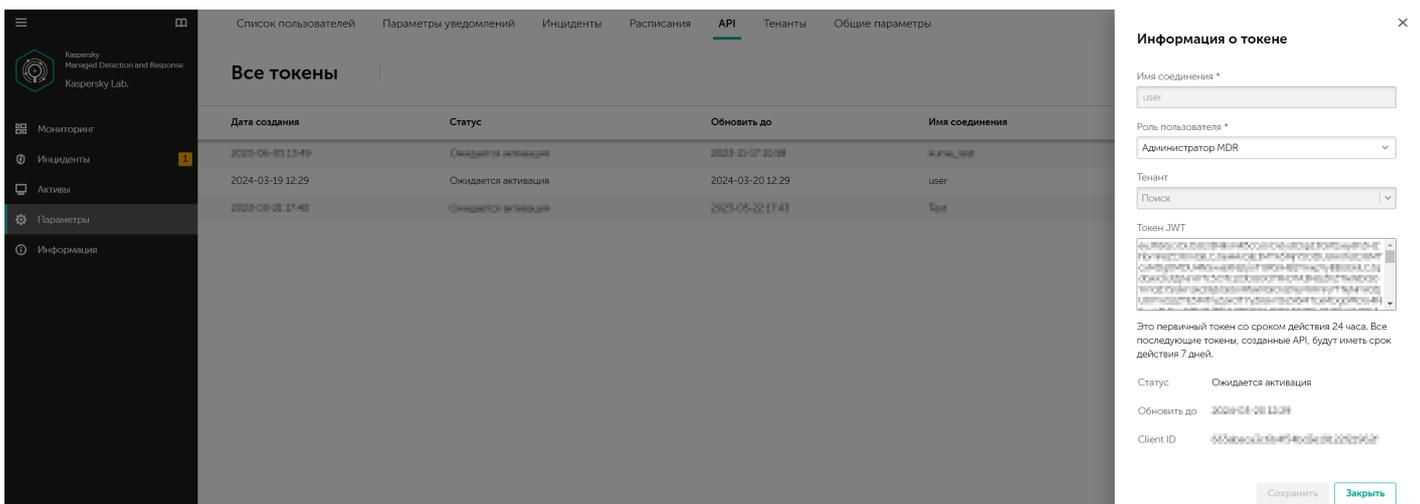
## Предварительные требования

- На пограничном МСЭ создайте правило доступа к **mdr.kaspersky.com** по порту **TCP/443** (<https://mdr.kaspersky.com>)
- В KUMA добавьте пользователя с ролью Администратор и доступом к API (права **POST /events**)
- В консоли MDR сгенерируйте токен для доступа к API:
  - Перейдите в **Settings -> API**
  - Нажмите **Add** и укажите **Connection Name** (данное имя будет использоваться, как имя пользователя при создании инцидентов/комментариев/вложений и т.д., так как токен доступа не привязан к конкретному пользователю)
  - Укажите **Role**, чтобы определить права доступа для токена
  - Укажите **Tenant** при необходимости

Если для создаваемых в консоли MDR инцидентов в поле "Тенант" отсутствует значение, добавьте значение "**Корень без тенанта**", чтобы скрипт при подключении обнаружил данные инциденты



- Нажмите **Generate**
- После завершения процесса генерации будут получены:
  - **JWT Token** - он же **refresh\_token**, который требуется активировать, чтобы получить новую пару **refresh\_token** и **access\_token**
  - **ClientID** - ID-клиента для подключения к API (требуется указывать при каждом запросе к API MDR)



- Загрузите архив **kuma\_mdr\_integration.tar.gz** со скриптом отсюда - <https://box.kaspersky.com/f/11a58e42f63e4cef8741/>
- Загрузите актуальную цепочку сертификатов для консоли mdr.kaspersky.com в формате pem. После загрузки файлов выполните объединение сертификатов в один файл **mdr.pem** (см. скриншот ниже).

General **Details**

Certificate Hierarchy

- ▼ DigiCert Global Root G2 **3**
  - ▼ DigiCert Global G2 TLS RSA SHA256 2020 CA1 **2**
    - \*.mdr.kaspersky.com** **1**

Certificate Fields

- ▼ \*.mdr.kaspersky.com
  - ▼ Certificate
    - Version
    - Serial Number
    - Certificate Signature Algorithm
    - Issuer
  - ▼ Validity
    - Not Before

Field Value



Export...



# Описание интеграции

Описанная в данной статье интеграция с Kaspersky Managed Detection and Response позволяет автоматически импортировать инциденты из консоли MDR в KUMA.

## Настройка

- Скопируйте архив **kuma\_mdr\_integration.tar.gz** на сервер (в случае распределенной инсталляции на сервер Core) и распакуйте его в папку **/opt** с помощью следующей команды:

```
sudo tar -xf kuma_mdr_integration.tar.gz -C /opt
```

- Перейдите в папку **/opt/mdr/conf** и отредактируйте файл **config.yml**:
  - В секции **General settings** укажите **актуальный путь до папок (по умолчанию, указан /opt/mdr/\*)**, а также **client\_id**
  - В секции **Modules settings** -> **kuma** укажите:
    - **api\_url** (FQDN/IP:порт API-интерфейса KUMA)
    - **username** (ранее созданный пользователь с доступом к API)
    - **password** (пароль ранее созданного пользователя с доступом к API)
    - **tenantId** (тенант, в котором будут создаваться инциденты)

TenantID можно получить из события аудита KUMA

- В секции **Modules settings** -> **logging** также укажите **актуальный путь до папки со скриптом (по умолчанию, указан /opt/mdr/log)**
- Перейдите в папку **/opt/mdr/conf** и добавьте в файл **.refresh\_token** ранее сгенерированный токен для доступа к API MDR

После добавления токена в файл **.refresh\_token** проверьте, что в конце файла отсутствует символ новой строки `\n`, из-за которого попытка аутентификации будет неуспешной. См. команды ниже:

```
# проверяем наличие символа новой строки
wc -l /opt/mdr/conf/.refresh_token

# если вывод "1 .refresh_token", то удаляем символ
perl -p -i -e 'chomp if eof' /opt/mdr/conf/.refresh_token

# проверяем, что символ новой строки успешно удален
wc -l /opt/mdr/conf/.refresh_token
```

```
# должен быть вывод "0 .refresh_token"
```

- Перейдите в папку **/opt/mdr/conf** и в файле **.last\_check** укажите время, начиная с которого необходимо начать собирать инциденты. Для теста можно указать время до появления последнего инцидента. Формат в миллисекундах, то есть должно быть 13 цифр (пример, 1672520400000)
- Замените существующий файл **/opt/mdr/conf/mdr.pem** на актуальный (см. этап "Предварительные требования")
- Запустите скрипт **main.py** с помощью команды:

```
python3 ./main.py
```

Если при запуске скрипта появляются сообщения об отсутствии необходимых пакетов - выполните их установку

- Если после запуска скрипта в консоли отсутствуют ошибки (кроме предупреждений о невалидном сертификате), значит интеграция работает корректно.

Лог работы скрипта пишется в **/opt/mdr/log/app.log**

- Убедитесь, что выполнен импорт инцидентов, созданных в консоли MDR начиная с момента времени, указанного в файле **.last\_check**

## Incidents

<input type="checkbox"/>	Name	Created ↓	Severity	Tenant
<input type="checkbox"/>	W10-MDR-KES.evilcorp.local - проверка зашифрованного архива	2023-09-13 20:31:46	Low	Main

- Остановите выполнение скрипта **main.py**
- Запустите скрипт **main.py** в фоновом режиме с помощью команды:

```
nohup python3 /opt/mdr/main.py &
```

- Настройте автоматический запуск скрипта после перезагрузки сервера:

```
sudo crontab -e
@reboot sleep 300 && python3 /opt/mdr/main.py & # sleep в 5 минут добавлен, чтобы сервис kuma-core успел стартовать
```

Revision #20

Created 26 February 2024 14:04:55 by Dmitry Borisov

Updated 3 September 2024 08:48:10 by Koala