

Сканеры уязвимостей

Добавление активов в KUMA по отчетам различных сканеров уязвимостей

- [Nessus и OWASP ZAP](#)
- [Импорт информации об активах RedCheck](#)
- [Импорт данных из отчетов MaxPatrol в KUMA 3.2](#)

Nessus и OWASP ZAP

В KUMA можно импортировать сведения об активах из отчетов о результатах сканирования устройств с помощью Nessus, OWASPZAP, системы контроля защищенности и соответствия стандартам. Импорт происходит через API с помощью утилиты `import_asset_Nessus_OWASP.py`

(скрипт находится в Пресейл-Пак в папке Assets [ссылка доступна из [Дисклеймера](#)]).

Импортированные активы отображаются в веб-интерфейсе KUMA в разделе Активы. При необходимости вы можете редактировать параметры активов.

Предварительные настройки в KUMA

Создайте пользователя с ролью **главный администратор** со следующими **правами доступа для API**:

- GET /tenants
- GET /users/whoami
- POST /assets/import

Пользователь

☐ Выключен

*Имя

asset-api

*Логин

asset-api

*Адрес электронной почты

asset-api@lo.cal

☐ Получать уведомления по п

☐ Скрывать ресурсы из общег

☒ Может взаимодействовать

☐ Получать уведомления о со

☒ Группа главных администра

☒ Доступ к объектам КИИ

Взаимодействие с KUMA чере

Сгенерировать токен

Пр

Изменить пароль

Права доступа через API

GET /activeLists

☐

GET /alerts

☐

GET /assets

☐

GET /dictionaries

☐

GET /events/clusters

☐

GET /resources

☐

GET /resources/download/id

☐

GET /services

☐

GET /settings/id/id

☐

GET /system/backup

☐

GET /tenants

☒

GET /users/whoami

☒

POST /activeLists/import

☐

POST /alerts/close

☐

POST /assets/delete

☐

POST /assets/import

☒

POST /dictionaries/update

☐

POST /events

☐

POST /resources/export

☐

POST /resources/import

☐

POST /resources/toc

☐

POST /resources/upload

☐

POST /system/restore

☐

Сохранить

Сохраните настройки, **сгенерируйте токен** и отдельно сохраните его, например, в каком-либо текстовом редакторе. Нажмите **Сохранить**.

Добавьте **дополнительно поле** с названием «Description» в разделе **Параметры - Активы - Пользовательские атрибуты**.

Активы

Подробнее о пользовательских полях активов см. [в онлайн-справке](#).

При удалении настраиваемых полей также удаляются содержащиеся в них данные.

Пользовательские атрибуты

Название

Маска (регулярное выражение,
RE2)

Значение по умолчанию



Description



Сохраните изменения.

Импорт отчета от Nessus

Пример отчета от Nessus в формате CSV:

```
Plugin ID,CVE,CVSS v2.0 Base Score,Risk,Host,Protocol,Port,Name,Synopsis,Description,Solution,See Also,Plugin
Output,STIG Severity,CVSS v3.0 Base Score,CVSS v2.0 Temporal Score,CVSS v3.0 Temporal Score,VPR
Score,Risk Factor,BID,XREF,MSKB,Plugin Publication Date,Plugin Modification Date,Metasploit,Core
Impact,CANVAS
"70658","CVE-2008-5161","2.6","Low","1.2.3.9","tcp","22","SSH Server CBC Mode Ciphers Enabled","The SSH
server is configured to use Cipher Block Chaining.","The SSH server is configured to support Cipher Block
Chaining (CBC)encryption. This may allow an attacker to recover the plaintext messagefrom the ciphertext.Note
that this plugin only checks for the options of the SSH server anddoes not check for vulnerable software
versions.","Contact the vendor or consult product documentation to disable CBC modecipher encryption, and
enable CTR or GCM cipher mode encryption.",,"The following client-to-server Cipher Block Chaining (CBC)
algorithmsare supported:3des-cbcaes128-cbcaes192-cbcaes256-cbcblowfish-cbcast128-cbcThe following
server-to-client Cipher Block Chaining (CBC) algorithmsare supported:3des-cbcaes128-cbcaes192-cbcaes256-
cbcbowfish-cbcast128-
cbc","","","1.9","","2.5","Low","32319","CERT:958563;CWE:200","","2013/10/28","2018/07/30","","",""
```

Далее необходимо хапустить скрипт `import_asset_Nessus_OWASP.py` по этому отчету указав необходимые параметры для его корректного запуска. Возможные опции скрипта:

```
# python import_asset.py --help
usage: import_asset.py [-h] --kuma KUMA --token TOKEN --tenant TENANT --vendor {Nessus,OWASPZAP} --
filepath FILEPATH
```

options:

-h, --help	show this help message and exit
--kuma KUMA	IP адрес сервера KUMA
--token TOKEN	Токен API
--tenant TENANT	Имя Тенанта
--vendor {Nessus,OWASPZAP}	Наименование вендора
--filepath FILEPATH	Путь до отчета

Пример запуска по отчету Nessus:

```
python3 import_asset_Nessus_OWASP.py --kuma 10.68.85.126 --token 98417b064c2a5cdfdf6bd011126c6453 --tenant Main --vendor Nessus --filepath C:\Users\ose\Downloads\nessus.csv
```

В KUMA актив будет выглядеть следующим образом:

Поиск...

☐

Название ↑

☐

12.3.11

☐

12.3.12

☐

12.3.4

☐

12.3.9

☐

10.68.85.1

☐

10.68.85.11

☐

10.68.85.13

☐

10.68.85.145

☐

10.68.85.2

☐

195.98.36.92

☐

Asset 1 Name

☐

Asset 2

☐

Asset 2 Name

☐

Asset 3 Name

☐

KSCNEW

☐

assets.name

☐

demo.lab

Информация об активе

Удалить

Изменить

Регистрирование KEDR

Название

12.3.9

Тенант

Main

Источник актива

Создан вручную

Идентификатор

53dc3630-a855-4308-a5e8-92eed8faf0dc

Создано

12.12.2023 18:04:45

Последнее обновление

12.12.2023 18:18:09

IP-адрес

12.3.9

Владелец

asset-api

Категория КИИ

Информационный ресурс не является объектом КИИ

Настраиваемые поля

Description

Создано на основе отчета системы Nessus.

Категории

Другие уязвимости

SSH Server CBC Mode Ciphers Enabled

CVE 1 : CVE-2008-5161

Импорт отчета от OWASP ZAP

Пример отчета от Nessus в формате CSV:

```
{
  "@programName": "OWASP ZAP",
  "@version": "2.13.0",
  "@generated": "Mon, 25 Sep 2023 11:43:20",
  "site": [
    {
      "@name": "https://demo.lab",
      "@host": "demo.lab",
      "@port": "443",
      "@ssl": "true",
      "alerts": [
        {
          "pluginid": "10035",
          "alertRef": "10035",
          "alert": "Strict-Transport-Security Header Not Set",
          "name": "Strict-Transport-Security Header Not Set",
          "riskcode": "1",
          "confidence": "3",
          "riskdesc": "Low (High)",
          "reference": "https://ya.ru",
          "desc": "<p>HTTP Strict Transport Security (HSTS)</p>"
        }
      ]
    }
  ]
}
```

Далее необходимо хапустить скрипт `import_asset_Nessus_OWASP.py` по этому отчету указав необходимые параметры для его корректного запуска. Возможные опции скрипта:

```
# python import_asset.py --help
```

```
usage: import_asset.py [-h] --kuma KUMA --token TOKEN --tenant TENANT --vendor {Nessus,OWASPZAP} --
filepath FILEPATH
```

options:

-h, --help	show this help message and exit
--kuma KUMA	IP адрес сервера KUMA
--token TOKEN	Токен API
--tenant TENANT	Имя Тенанта
--vendor {Nessus,OWASPZAP}	Наименование вендора

--filepath FILEPATH Путь до отчета

Пример запуска по отчету OWASP ZAP:

```
python3 import_asset_Nessus_OWASP.py --kuma 10.68.85.126 --token 98417b064c2a5cdfdf6bd011126c6453 --tenant Main --vendor OWASPZAP --filepath C:\Users\ose\Downloads\owasp.json
```

В KUMA актив будет выглядеть следующим образом:

Поиск...

Название ↑

10.68.85.11

10.68.85.13

10.68.85.145

10.68.85.2

195.98.36.92

Asset 1 Name

Asset 2

Asset 2 Name

Asset 3 Name

KSCNEW

assets.name

demo.lab

lamoda.ru

localhost

test asset for comm

test_vuln

test_vuln

Информация об активе

Удалить

Изменить

Реагирование KEDR

Название

demo.lab

Тенант

Main

Источник актива

Создан вручную

Идентификатор

0f285395-8d26-4809-b33d-7857435d9dde

Создано

12.12.2023 18:10:59

Последнее обновление

12.12.2023 18:18:09

Владелец

asset-api

Полное доменное имя

demo.lab

Категория КИИ

Информационный ресурс не является объектом КИИ

Настраиваемые поля

Description

Создано на основе отчета системы OWASP ZAP.

Категории

Другие уязвимости

10035 в Strict-Transport-Security Header Not Set

Импорт информации об активах RedCheck

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

Импорт информации об активах RedCheck

В KUMA можно импортировать сведения об активах из отчетов о результатах сканирования сетевых устройств с помощью RedCheck, системы контроля защищенности и соответствия стандартам. Импорт происходит через API с помощью утилиты `redcheck-tool.py`.

Импортированные активы отображаются в веб-интерфейсе KUMA в разделе Активы. При необходимости вы можете редактировать параметры активов.

Импорт поддерживается из RedCheck 2.6.8 и выше

Поддерживается импорт информации о хостах только под управлением ОС Windows

Для работы утилиты требуется python версии 3.6 или выше, а также библиотеки: csv, re, json, requests, argparse, sys

Чтобы импортировать данные об активах из отчета RedCheck:

1. Сформируйте в RedCheck отчет сканирования сетевых активов в формате CSV и скопируйте файл отчета на сервер со скриптом. Подробнее о задачах на сканирование и форматах выходных файлов см. в документации RedCheck.

Импорт доступен из "Простых" отчетов **"Уязвимости"** и **"Инвентаризация"** сгруппированных по хостам в формате CSV. Подробнее на сайте RedCheck:
<https://docs.redcheck.ru/articles/#!redcheck-user-269/reports>

2. Создайте токен для доступа к KUMA REST API.

Требования к учетным записям, для которых генерируется API-токен:

- Роль Администратора или Аналитика.
- Доступ к тенанту, в который будут импортированы активы.
- Настроены права на использование API-запросов GET /assets, GET /tenants, POST /assets/import

3. Скопируйте утилиту redcheck-tool.py на сервер ядра KUMA и сделайте файл утилиты исполняемым с помощью команды:

```
chmod +x <путь до файла redcheck-tool.py>
```

4. Запустите утилиту redcheck-tool.py:

```
python3 redcheck-tool.py --kuma-rest <адрес и порт сервера KUMA REST API> --token <API-токен> --tenant <название тенанта, куда будут помещены активы> --vuln-report <Полный путь к файлу отчета "Уязвимости"> --inventory-report <Полный путь к файлу с отчета "Инвентаризация">
```

Пример:

```
python3 --kuma-rest example.kuma.com:7223 --token 949fc03d97bad5d04b6e231c68be54fb --tenant Main --vuln-report /home/user/vuln.csv --inventory-report /home/user/inventory.csv
```

Вы можете использовать дополнительные флаги и команды для импорта. Например, команду для отображения расширенного отчета о полученных активах `-v`. Подробное описание доступных флагов и команд приведено в таблице Флаги и команды утилиты redcheck-tool.py. Также для просмотра информации о доступных флагах и командах вы можете использовать команду `--help`.

Информация об активах будет импортирована из отчета RedCheck в KUMA. В консоли отображаются сведения о количестве новых и обновленных активов.

Пример:

```
inventory has been imported for 2 host(s)
software has been imported for 5 host(s)
vulnerabilities has been imported for 4 host(s)
```

Пример расширенного вывода информации об импорте:

```
[inventory import]      Host: localhost    Code: 200  Response: {'insertedIDs': {'0': '52ca11c6-a0e6-4dfd-8ef9-bf58189340f8'}, 'updatedCount': 0, 'errors': []}
```

```
[inventory import]      Host: 10.0.0.2      Code: 200 Response: {'insertedIDs': {'0': '1583e552-5137-4164-92e0-01e60fb6edb0'}, 'updatedCount': 0, 'errors': []}
[software import][error] Host: localhost      Skipped asset with FQDN localhost or IP 127.0.0.1
[software import]      Host: 10.0.0.2      Code: 200 Response: {'insertedIDs': {}, 'updatedCount': 1, 'errors': []}
[vulnerabilities import] Host: 10.0.0.2      Code: 200 Response: {'insertedIDs': {}, 'updatedCount': 1, 'errors': []}
[vulnerabilities import] Host: 10.0.0.1      Code: 200 Response: {'insertedIDs': {'0': '0628f683-c20c-4107-abf3-d837b3dbbf01'}, 'updatedCount': 0, 'errors': []}
[vulnerabilities import] Host: localhost      Code: 200 Response: {'insertedIDs': {}, 'updatedCount': 1, 'errors': []}
[vulnerabilities import] Host: 10.0.0.3      Code: 200 Response: {'insertedIDs': {'0': 'ed01e0a8-dcb0-4609-ab2b-91e50092555d'}, 'updatedCount': 0, 'errors': []}
inventory has been imported for 2 host(s)
software has been imported for 1 host(s)
vulnerabilities has been imported for 4 host(s)
```

Поведение утилиты при импорте активов:

- Данные импортированных в KUMA через API активов перезаписываются, а сведения об их устаревших уязвимостях удаляются.
- Активы с недействительными данными пропускаются.

Флаги и команды утилиты redcheck-tool.py

Флаги и команды	Обязательный	Описание
--kuma-rest <адрес и порт сервера ядра KUMA>	Да	По умолчанию для обращения по API используется порт 7223. При необходимости его можно изменить.
--token <токен>	Да	Значение в параметре должно содержать только токен. Учетной записи, для которой генерируется API-токен, должна быть присвоена роль Администратора или Аналитика.
--tenant <название тенанта>	Да	Название тенанта KUMA, в который будут импортированы активы из отчета RedCheck
--vuln-report <полный путь к файлу отчета "Уязвимости">	Да	Файл должен содержать отчет "Уязвимости" в формате CSV
--inventory-report <полный путь к файлу отчета "Инвентаризация">	Нет	Файл должен содержать отчет "Инвентаризация" в формате CSV

-v	Нет	Выведение расширенной информации об импорте активов
----	-----	---

Возможные ошибки

Сообщение об ошибке	Описание
Tenant %w not found	Имя тенанта не найдено
Tenant search error: Unexpected status Code: %d	При поиске тенанта был получен неожиданный код ответа HTTP
Asset search error: Unexpected status Code: %d	При поиске актива был получен неожиданный код ответа HTTP
[%w import][error] Host: %w Skipped asset with FQDN localhost or IP 127.0.0.1	При импорте информации инвентаризации/уязвимостей был пропущен хост с fqdn=localhost или ip=127.0.0.1

Импорт данных из отчетов MaxPatrol в KUMA 3.2

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

Ссылка на документацию <https://support.kaspersky.com/help/KUMA/3.2/ru-RU/265426.htm>

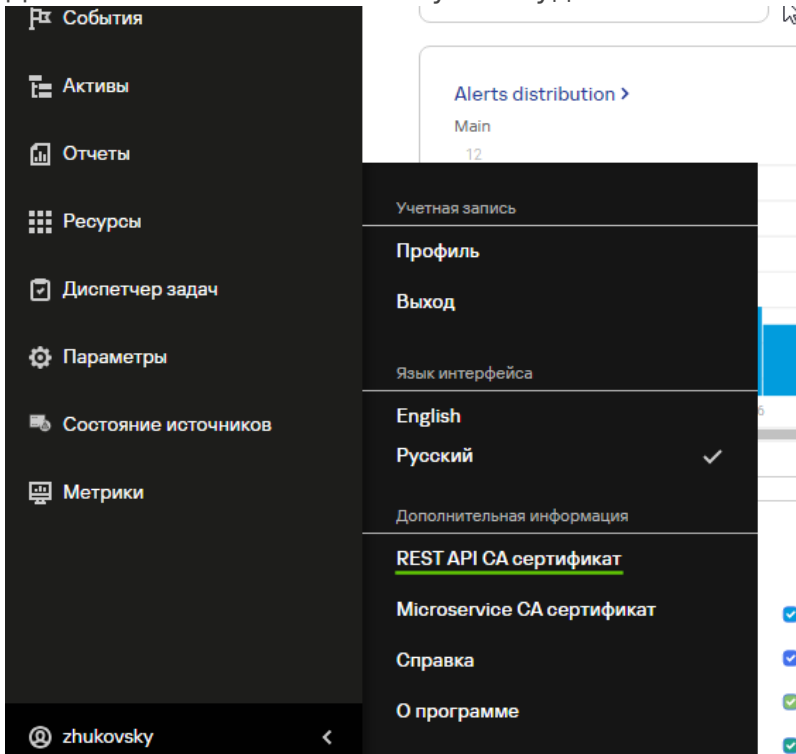
Импорт

Утилита для импорта находится в следующей директории **kuma-ansible-installer/roles/kuma/files/maxpatrol-tool**

Для импорта необходимо выполнить следующие шаги:

1. Сформировать отчет **сканирования сетевых активов в формате XML file**
2. Поместить отчет и скрипт в одной директории (можно выполнить на ядре KUMA)
3. Подготовить API права для утилиты:
 1. В KUMA создать пользователя kuma-mp
 2. Дать права на **GET /users/whoami** и **POST /assets/import**
 3. Сгенерировать токен (его сохранить в файл на сервере, где будет выполняться команда

4. Далее в веб-консоли KUMA нужно будет скачать REST API CA



5. Этот сертификат нужно перенести на сервер, где будет выполняться скрипт

6. Выполнить команду:

```
./maxpatrol-tool --kuma-rest <адрес и порт сервера KUMA REST API> --token <путь и имя файла с API-токеном> --tenant <название тенанта, куда будут помещены активы> <путь и имя файла с отчетом MaxPatrol> --cert <путь к файлу сертификата Ядра KUMA>
```

Пример:

```
./maxpatrol-tool --kuma-rest example.kuma.com:7223 --token token.txt --tenant Main mp.xml --cert core-external-ca.cert
```

7. Для автоматизации добавления активов из папки по отчетам - можно воспользоваться [скриптом](#).