

HPE Configuration Collector (HCC)

НСС позволяет собирать информацию о конфигурации оборудования напрямую из iLO, используя при этом базовые запросы, аналогичные запросам пользователя к менеджменту в браузере.

Принцип работы

НСС взаимодействует с iLO напрямую, что позволяет собирать всю информацию, которую пользователь может там увидеть.

Первым делом скрипт отправляет POST-запрос на логин и получает session-token, который используется в дальнейшем взаимодействии.

Далее скрипт GET-запросами обходит каждый endpoint (по одному на позицию собираемой информации) и получает от сервера json-объекты, содержащие необходимую информацию о конфигурации. Json-объекты собираются в общий Json-файл, который сохраняется в папку /output с серийным номером сервера в качестве имени. После этого скрипт закрывает соединение и переходит к следующему серверу в списке.

При написании скрипта запросы собирались в браузере и обрабатывались через Postman.

Коллекция Postman доступна для ознакомления в папке /Examples.

Собираемая информация:

- System Summary
- CPU
- RAM
- Storage
- NIC
- PCI
- PSU

Поддерживаемое оборудование

- Серверы HPE под управлением iLO4
- Серверы HPE под управлением iLO5
- Блейд-корзины c7000 (в разработке)

Установка

Для выполнения скрипта требуется Python3 и ряд дополнительных библиотек:

- progress==1.6
- Requests==2.31.0
- urllib3==2.0.2

Установить библиотеки по отдельности можно командой:

```
pip install ИмяБиблиотеки
```

Либо все библиотеки разом, используя requirements.txt (находится в корне репозитория):

```
pip install -r /path/to/requirements.txt
```

Input-файл

На вход скрипт может принимать .csv файл следующего формата:

```
ip;login;password
172.24.1.1;testlogin;testpass
172.24.1.2;logintest;passtest
.....
```

Пример файла также можно посмотреть в папке /Examples.

Запуск

Скрипт принимает на вход следующие аргументы:

Параметр	Описание
-f	Путь к файлу input.csv
-d	Кастомный разделитель для input.csv . По-умолчанию - ;
-r	Диапазон ip-адресов для сбора со стандартными логинам-паролями без input.csv
-l	Стандартный логин
-p	Стандартный пароль
-dm	Debug mode

Параметры **-f** и **-d** не могут использоваться совместно с параметрами **-r**, **-l**, **-p**.

Примеры

Запуск с файлом, содержащим ip-адреса iLO, логины и пароли:

```
python start.py -f C:\Users\micke\Desktop\UltimateHPEParser\input.csv
```

Запуск с файлом, содержащим ip-адреса iLO, логины и пароли + кастомный разделитель:

```
python start.py -f C:\Users\micke\Desktop\UltimateHPEParser\input.csv -d .
```

Запуск с ip-диапазоном на последнем октете и одинаковыми логинами-паролями:

```
python start.py -r 172.24.0.1-50 -l testlogin -p testpass
```

Запуск с ip-диапазоном в нотации CIDR и одинаковыми логинами-паролями:

```
python start.py -r 172.24.0.0/28 -l testlogin -p testpass
```

Output-файлы

После работы скрипта сгенерируется папка `/output` , содержащая информацию о конфигурации оборудования в формате `.json` .

Также появится файл `logs.log` . Необходимо скопировать этот файл в папку `/output` и заархивировать любым доступным архиватором.

Примеры этих файлов можно найти в папке `/Examples`.