

# Использование OpenAPI Kaspersky Security Center

В этой статье описаны возможности применения методов OpenAPI Kaspersky Security Center для автоматизации процедур развертывания и использования программы Kaspersky Security для виртуальных сред 5.2 Легкий агент в режиме multitenancy.

Подробное описание функций OpenAPI Kaspersky Security Center находится в файле **ksopenapi.chm**, который расположен в папке установки Kaspersky Security Center, папка установки по умолчанию C:\Program Files (x86)\Kaspersky Lab\Kaspersky Security Center.

<b>Сценарии использования OpenAPI Kaspersky Security Center .....</b>	<b>2</b>
<i>Перемещение виртуальной машины на виртуальный Сервер администрирования Kaspersky Security Center.....</i>	<i>2</i>
<i>Распространение инсталляционного пакета на виртуальный Сервер администрирования.....</i>	<i>2</i>
<i>Установка программы на виртуальную машину.....</i>	<i>3</i>
<i>Удаление программы с виртуальной машины .....</i>	<i>4</i>
<i>Удаление виртуальной машины из защищаемой инфраструктуры .....</i>	<i>4</i>
<b>Используемые методы OpenAPI Kaspersky Security Center .....</b>	<b>5</b>
<i>Авторизация на главном Сервере администрирования Kaspersky Security Center.....</i>	<i>5</i>
<i>Авторизация на виртуальном Сервере администрирования Kaspersky Security Center .....</i>	<i>5</i>
<i>Работа с виртуальными машинами .....</i>	<i>6</i>
Поиск виртуальных машин на Сервере администрирования Kaspersky Security Center .....	6
Перемещение виртуальной машины в группу администрирования .....	7
Удаление виртуальной машины .....	7
<i>Работа с перечислителем .....</i>	<i>8</i>
Получение количества элементов перечислителя.....	8
Получение элементов перечислителя .....	8
Освобождение перечислителя.....	9
<i>Работа с группами администрирования .....</i>	<i>10</i>
Поиск групп администрирования .....	10
Получение идентификатора группы Управляемые устройства .....	10
Получение идентификатора группы Нераспределенные устройства .....	11
<i>Получение списка виртуальных Серверов администрирования .....</i>	<i>11</i>
<i>Работа с инсталляционными пакетами.....</i>	<i>12</i>
Получение списка инсталляционных пакетов.....	12
Инсталляционные пакеты Агента администрирования Kaspersky Security Center и программы Kaspersky Security .....	13
Распространение инсталляционного пакета на виртуальный Сервер администрирования .....	13
Проверка состояния асинхронной операции.....	14
<i>Работа с задачами .....</i>	<i>15</i>
Создание задачи Смена Сервера администрирования .....	15
Создание задачи удаленной установки программы.....	16
Создание задачи удаленной деинсталляции программы .....	18
Запуск задачи.....	19

## Сценарии использования OpenAPI Kaspersky Security Center

### Перемещение виртуальной машины на виртуальный Сервер администрирования Kaspersky Security Center

Процедура позволяет перемещать на виртуальный Сервер администрирования виртуальные машины, для которых выполняются следующие условия:

- На виртуальной машине установлен Агент администрирования Kaspersky Security Center, в параметрах которого задан адрес главного Сервера администрирования.
- Виртуальной машине присвоен уникальный идентификатор.
- Виртуальная машина находится на главном Сервере администрирования Kaspersky Security Center (например, в группе Нераспределенные устройства).

Чтобы переместить виртуальную машину на виртуальный Сервер администрирования Kaspersky Security Center, нужно выполнить следующие действия:

1. Авторизоваться на главном Сервере администрирования Kaspersky Security Center, используя метод [Login](#).
2. Получить список виртуальных машин вызовом метода [HostGroup.FindHosts](#) с полями:
  - **KLHST\_WKS\_DN** – отображаемое имя виртуальной машины.
  - **KLHST\_WKS\_HOSTNAME** – уникальный идентификатор виртуальной машины.
  - **KLHST\_WKS\_IP\_LONG** – IP-адрес виртуальной машины.
3. В полученном списке найти нужную виртуальную машину по IP-адресу в поле **KLHST\_WKS\_IP\_LONG** и запомнить значение полей **KLHST\_WKS\_DN** и **KLHST\_WKS\_HOSTNAME**.
4. Авторизоваться на виртуальном Сервере администрирования, используя метод [Login-VirtualServer](#).
5. Создать на виртуальном Сервере администрирования Kaspersky Security Center задачу [Смены Сервера администрирования](#) и запомнить ее идентификатор **taskId**:
  - a. В поле **NewKscServerAddress** указать строку вида **MainKscServerAddress/VirtualServerName**.
  - b. Присвоить значения, полученные на шаге 3: **HostDispName** = **KLHST\_WKS\_DN** и **HostName** = **KLHST\_WKS\_HOSTNAME**.
  - c. Указать отображаемое название задачи в поле **DisplayName**.
6. [Запустить задачу](#), указав ее идентификатор **taskId**, полученный на предыдущем шаге.
7. Дождаться завершения задачи.

### Распространение инсталляционного пакета на виртуальный Сервер администрирования

Чтобы распространить инсталляционный пакет на виртуальный Сервер администрирования, нужно выполнить следующие действия:

1. Используя метод [PackagesApi.GetPackages](#), найти идентификатор пакета **KLPKG\_NPI\_PKGID** для программы **KLPKG\_NPI\_PRODUCT\_NAME** и версии **KLPKG\_NPI\_PRODUCT\_VERSION**.
2. Используя метод [VServers.GetVServers](#), найти идентификатор виртуального Сервера администрирования **KLVSrv\_ID**, соответствующего имени **KLVSrv\_DN**.

3. Запустить распространение пакета на виртуальный Сервер администрирования, используя метод [PackagesApi.RetranslateToVServerAsync](#). В результате возвращается идентификатор асинхронной операции.
4. Проверить состояние асинхронной операции по ее идентификатору, полученному на предыдущем шаге, вызовом [AsyncActionStateChecker.CheckActionState](#).

## Установка программы на виртуальную машину

Процедура позволяет удаленно устанавливать программу на виртуальную машину, если выполняются следующие условия:

- На виртуальной машине установлен Агент администрирования Kaspersky Security Center.
- Виртуальная машина видна на виртуальном Сервере администрирования (например, в группе Нераспределенные устройства).
- На виртуальном Сервере администрирования имеются инсталляционные пакеты Агента администрирования и Kaspersky Security для виртуальных сред 5.2 Легкий агент. Если пакетов нет, вам нужно предварительно их создать или [передать их на нужный виртуальный Сервер](#).

Чтобы установить программу Kaspersky Security для виртуальных сред 5.2 Легкий агент, нужно выполнить следующие действия:

1. Авторизоваться на Сервере администрирования, на котором будет запускаться задача удаленной установки программы. Для авторизации на главном Сервере администрирования используется метод [Login](#), для авторизации на виртуальном Сервере администрирования используется метод [Login-VirtualServer](#).
2. Получить список виртуальных машин вызовом метода [HostGroup.FindHosts](#) с полями:
  - **KLHST\_WKS\_DN** – отображаемое имя виртуальной машины.
  - **KLHST\_WKS\_HOSTNAME** – уникальный идентификатор виртуальной машины.
  - **KLHST\_WKS\_IP\_LONG** – IP адрес виртуальной машины.
3. В полученном списке найти нужную виртуальную машину по IP-адресу в поле **KLHST\_WKS\_IP\_LONG** и запомнить значение полей **KLHST\_WKS\_DN** и **KLHST\_WKS\_HOSTNAME**.
4. Используя метод [PackagesApi.GetPackages](#), найти идентификатор инсталляционного пакета **KLPKG\_NPI\_PKGID** для Агента администрирования **KLPKG\_NPI\_PRODUCT\_NAME = 1103** и версии **KLPKG\_NPI\_PRODUCT\_VERSION = 1.0.0.0** (см таблицу [PackagesIds](#)).
5. Используя метод [PackagesApi.GetPackages](#), найти идентификатор инсталляционного пакета **KLPKG\_NPI\_PKGID** для программы Kaspersky Security для виртуальных сред 5.2 Легкий агент **KLPKG\_NPI\_PRODUCT\_NAME = KSVLA** и версии **KLPKG\_NPI\_PRODUCT\_VERSION = 5.2.0.0** (см таблицу [PackagesIds](#)).
6. Используя метод [HostGroup.FindGroups](#) или [HostGroup.GroupIdGroups](#), указать группу администрирования, в которую надо переместить виртуальную машину после установки программы.
7. Создать [задачу удаленной установки](#) и запомнить ее идентификатор **taskId**:
  - a. Присвоить **KINagentPackageId** значение **KLPKG\_NPI\_PKGID**, полученное на шаге 4.
  - b. Присвоить **ProductPackageId** значение **KLPKG\_NPI\_PKGID**, полученное на шаге 5.
  - c. Присвоить значения, полученные на шаге 3: **HostDispName = KLHST\_WKS\_DN** и **HostName = KLHST\_WKS\_HOSTNAME**.

- d. Указать в полях **HostOsUserLogin** и **HostOsUserPassword** имя пользователя и пароль пользователя, от имени которого будет выполняться установка.
  - e. Указать в поле **GroupToMoveHostId**, группу администрирования, в которую нужно переместить виртуальную машину, полученную на шаге 6.
  - f. Указать отображаемое название задачи в поле **DisplayName**.
8. [Запустить задачу](#), указав ее идентификатор **taskId**, полученный на предыдущем шаге.
  9. Дождаться завершения задачи.

## Удаление программы с виртуальной машины

Чтобы удалить программу Kaspersky Security для виртуальных сред 5.2 Легкий агент с помощью Агента администрирования, нужно выполнить следующие действия:

1. Авторизоваться на Сервере администрирования, на котором будет запускаться задача удаленной установки программы. Для авторизации на главном Сервере администрирования используется метод [Login](#), для авторизации на виртуальном Сервере администрирования используется метод [Login-VirtualServer](#).
2. Получить список виртуальных машин вызовом метода [HostGroup.FindHosts](#) с полями:
  - **KLHST\_WKS\_DN** – отображаемое имя виртуальной машины.
  - **KLHST\_WKS\_HOSTNAME** – уникальный идентификатор виртуальной машины.
  - **KLHST\_WKS\_IP\_LONG** – IP адрес виртуальной машины.
3. В полученном списке найти нужную виртуальную машину по IP-адресу в поле **KLHST\_WKS\_IP\_LONG** и запомнить значение полей **KLHST\_WKS\_DN** и **KLHST\_WKS\_HOSTNAME**.
4. Создать [задачу удаленной деинсталляции](#) и запоминать ее идентификатор **taskId**
  - a. Присвоить значения, полученные на шаге 3: **HostDispName** = **KLHST\_WKS\_DN** и **HostName** = **KLHST\_WKS\_HOSTNAME**.
  - b. Указать в полях **HostOsUserLogin** и **HostOsUserPassword** имя пользователя и пароль пользователя, от имени которого будет выполняться удаление программы.
  - c. Указать информацию об удаляемой программе и ее версии в полях **ProductName** (например, KSVLA) и **ProductVersion** (например, 5.2.0.0).
  - d. Указать отображаемое название задачи в поле **DisplayName**.
5. [Запустить задачу](#), указав ее идентификатор **taskId**, полученный на предыдущем шаге.
6. Дождаться завершения задачи.

## Удаление виртуальной машины из защищаемой инфраструктуры

Перед удалением виртуальной машины вам нужно [удалить программу, установленную на виртуальной машине](#).

Чтобы удалить виртуальную машину из защищаемой инфраструктуры, нужно выполнить следующие действия:

1. Авторизоваться на виртуальном Сервере администрирования Kaspersky Security Center, на котором располагается нужная виртуальная машина, используя метод [Login-VirtualServer](#).
2. Получить список виртуальных машин вызовом метода [HostGroup.FindHosts](#) с полями:
  - **KLHST\_WKS\_DN** – отображаемое имя виртуальной машины.

- **KLHST\_WKS\_HOSTNAME** – уникальный идентификатор виртуальной машины.
  - **KLHST\_WKS\_IP\_LONG** – IP адрес виртуальной машины.
3. В полученном списке найти нужную виртуальную машину по IP-адресу в поле **KLHST\_WKS\_IP\_LONG** и запомнить значение полей **KLHST\_WKS\_DN** и **KLHST\_WKS\_HOSTNAME**.
  4. Если вы хотите удалить виртуальную машину из группы Управляемые устройства и перенести ее в группу Нераспределенные устройства, вам нужно выполнить следующие действия:
    - a. Получить идентификатор **id** группы Нераспределенные устройства вызовом метода [HostGroup.GroupIdUnassigned](#).
    - b. Вызвать [метод перемещения виртуальной машины](#) в группу администрирования, задав в поле **hostId** значение **KLHST\_WKS\_HOSTNAME**, полученное на шаге 3, и в поле **groupId** значение **id**, полученное на шаге 4а.
  5. Если вы хотите полностью удалить виртуальную машину, вам нужно вызвать [метод удаления виртуальной машины](#), передав в качестве идентификатора **hostId** виртуальной машины значение **KLHST\_WKS\_HOSTNAME**, полученное на шаге 3.

## Используемые методы OpenAPI Kaspersky Security Center

### Авторизация на главном Сервере администрирования Kaspersky Security Center

Для работы с OpenAPI Kaspersky Security Center необходимо авторизоваться на Сервере администрирования Kaspersky Security Center вызовом метода **login**:

```
POST https://MainKscServerIpAddress:Port/api/v1.0/login HTTP/1.1
Authorization: KSCBasic user="Base64UserKscLogin",
pass="Base64UserKscPassword", internal="0"
Content-Type: application/json
Content-Length: 2

{ }
```

где:

- **MainKscServerIpAddress** – IP-адрес в формате IPv4 или полное доменное имя (FQDN) главного Сервера администрирования Kaspersky Security Center.
- **Port** – порт OpenAPI Kaspersky Security Center на главном Сервере администрирования. По умолчанию это порт **13299**.
- **Base64UserKscLogin** – имя пользователя главного Сервера администрирования в кодировке Base64.
- **Base64UserKscPassword** – пароль пользователя главного Сервера администрирования в кодировке Base64.

### Авторизация на виртуальном Сервере администрирования Kaspersky Security Center

```
POST https://MainKscServerIpAddress:Port/api/v1.0/login HTTP/1.1
Authorization: KSCBasic user="Base64UserVirtualKscLogin",
pass="Base64UserVirtualKscPassword", internal="0"
X-KSC-VServer: Base64VirtualKscName
Content-Type: application/json
```

```
Content-Length: 2
{ }
```

где:

- **MainKscServerIpAddress** – IP-адрес в формате IPv4 или полное доменное имя (FQDN) главного Сервера администрирования Kaspersky Security Center.
- **Port** – порт OpenAPI Kaspersky Security Center на главном Сервере администрирования. По умолчанию это порт **13299**.
- **Base64UserVirtualKscLogin** – имя пользователя виртуального Сервера администрирования в кодировке Base64.
- **Base64UserVirtualKscPassword** – пароль пользователя виртуального Сервера администрирования в кодировке Base64.
- **Base64VirtualKscName** – имя виртуального Сервера администрирования в кодировке Base64.

## Работа с виртуальными машинами

### Поиск виртуальных машин на Сервере администрирования Kaspersky Security Center

```
POST
https://MainKscServerIpAddress:Port/api/v1.0/HostGroup.FindHosts
X-KSC-VServer: Base64VirtualKscName
Content-Type: application/json
Content-Length: BodyLength
{
  "wstrFilter" : "",
  "vecFieldsToReturn" : [
    "KLHST_WKS_DN",
    "KLHST_WKS_HOSTNAME",
    "KLHST_WKS_IP_LONG"
  ],
  "vecFieldsToOrder" : [],
  "pParams" : {"KLSRVH_SLAVE_REC_DEPTH" : 128,
    "KLGRP_FIND_FROM_CUR_VS_ONLY" : true },
  "lMaxLifeTime" : 600
}
```

где:

- **MainKscServerIpAddress** – IP-адрес в формате IPv4 или полное доменное имя (FQDN) главного Сервера администрирования Kaspersky Security Center.
- **Port** – порт OpenAPI Kaspersky Security Center на главном Сервере администрирования. По умолчанию это порт **13299**.
- **Base64VirtualKscName** – имя виртуального Сервера администрирования в кодировке Base64. Заголовок `X-KSC-VServer` требуется, если метод вызывается для виртуального Сервера администрирования.
- **BodyLength** – длина JSON тела запроса в байтах.
- **vecFieldsToReturn** – массив названий свойств виртуальной машины, которые нужно вернуть. Полный список полей указан в документации к методу.
  - **KLHST\_WKS\_DN** – отображаемое имя виртуальной машины.
  - **KLHST\_WKS\_HOSTNAME** – уникальный идентификатор виртуальной машины.
  - **KLHST\_WKS\_IP\_LONG** – IP адрес виртуальной машины.

Метод возвращает идентификатор объекта перечислителя [ChunkAccessor](#) на Сервере администрирования в виде:

```
{"strAccessor" : "iteratorId"}
```

По этому идентификатору вы можете получить информацию о виртуальных машинах, используя [методы работы с перечислителем](#). После завершения работы с перечислителем необходимо освободить перечислитель, вызвав метод [ChunkAccessor.Release](#).

## Перемещение виртуальной машины в группу администрирования

```
POST
https://MainKscServerIpAddress:Port/api/v1.0/HostGroup.MoveHostsToGroup
X-KSC-VServer: Base64VirtualKscName
Content-Type: application/json
Content-Length: BodyLength
{"pHostNames":
  [
    "hostId"
  ],
  "nGroup": groupId
}
```

где:

- **MainKscServerIpAddress** – IP-адрес в формате IPv4 или полное доменное имя (FQDN) главного Сервера администрирования Kaspersky Security Center.
- **Port** – порт OpenAPI Kaspersky Security Center на главном Сервере администрирования. По умолчанию это порт **13299**.
- **Base64VirtualKscName** – имя виртуального Сервера администрирования в кодировке Base64. Заголовок `X-KSC-VServer` требуется, если метод вызывается для виртуального Сервера администрирования.
- **BodyLength** – длина JSON тела запроса в байтах.
- **hostId** – идентификатор виртуальной машины, которую требуется переместить. Значение **KLHST\_WKS\_HOSTNAME**, полученное вызовом [HostGroup.FindHosts](#).
- **groupId** – идентификатор группы администрирования, в которую требуется переместить виртуальную машину. Значение **id**, полученное вызовом [HostGroup.FindGroups](#).

## Удаление виртуальной машины

```
POST
https://MainKscServerIpAddress:Port/api/v1.0/HostGroup.RemoveHost
X-KSC-VServer: Base64VirtualKscName
Content-Type: application/json
Content-Length: BodyLength
{"strHostName": "hostId"}
```

где:

- **MainKscServerIpAddress** – IP-адрес в формате IPv4 или полное доменное имя (FQDN) главного Сервера администрирования Kaspersky Security Center.
- **Port** – порт OpenAPI Kaspersky Security Center на главном сервере администрирования Kaspersky Security Center. По умолчанию это порт **13299**.
- **Base64VirtualKscName** – имя виртуального Сервера администрирования в кодировке Base64. Заголовок `X-KSC-VServer` требуется, если метод вызывается для виртуального Сервера администрирования.
- **BodyLength** – длина JSON тела запроса в байтах.
- **hostId** – идентификатор виртуальной машины, которую требуется удалить. Значение **KLHST\_WKS\_HOSTNAME**, полученное вызовом [HostGroup.FindHosts](#).

## Работа с перечислителем

### Получение количества элементов перечислителя

```
POST
https://MainKscServerIpAddress:Port/api/v1.0/ChunkAccessor.GetItemSc
ount
X-KSC-VServer: Base64VirtualKscName
Content-Type: application/json
Content-Length: BodyLength
{"strAccessor" : "iteratorId"}
```

где:

- **MainKscServerIpAddress** – IP-адрес в формате IPv4 или полное доменное имя (FQDN) главного Сервера администрирования Kaspersky Security Center.
- **Port** – порт OpenAPI Kaspersky Security Center на главном сервере администрирования. По умолчанию это порт **13299**.
- **Base64VirtualKscName** – имя виртуального Сервера администрирования в кодировке Base64. Заголовок `X-KSC-VServer` требуется, если метод вызывается для виртуального Сервера администрирования.
- **BodyLength** – длина JSON тела запроса в байтах.
- **iteratorId** – идентификатор перечислителя, полученный в поле **strAccessor** ответа метода [HostGroup.FindHosts](#).

Метод возвращает количество элементов перечислителя в виде:

```
{"PkgRetVal" : itemCount}
```

### Получение элементов перечислителя

```
POST
https://MainKscServerIpAddress:Port/api/v1.0/ChunkAccessor.GetItemSc
hunk
X-KSC-VServer: Base64VirtualKscName
Content-Type: application/json
Content-Length: BodyLength
{
  "strAccessor" : "iteratorId",
  "nStart" : from,
  "nCount" : itemsCount
}
```

где:

- **MainKscServerIpAddress** – IP-адрес в формате IPv4 или полное доменное имя (FQDN) главного Сервера администрирования Kaspersky Security Center.
- **Port** – Порт OpenAPI Kaspersky Security Center на главном сервере администрирования. По умолчанию это порт **13299**.
- **Base64VirtualKscName** – имя виртуального Сервера администрирования в кодировке Base64. Заголовок `X-KSC-VServer` требуется, если метод вызывается для виртуального Сервера администрирования.
- **BodyLength** – длина JSON тела запроса в байтах.
- **iteratorId** – идентификатор перечислителя, полученный в поле **strAccessor** ответа метода [HostGroup.FindHosts](#).
- **from** – индекс элемента коллекции, начиная с которого необходимо возвращать элементы. Отсчет с 0 индекса.
- **itemsCount** – количество элементов, которое нужно вернуть. Если необходимо вернуть все элементы, то можно использовать значение **itemsCount**, возвращенное при вызове метода [ChunkAccessor.GetItemScount](#).



Метод возвращает элементы перечислителя в виде (на примере вызова метода [HostGroup.FindHosts](#)):

```
{
  "pChunk": {
    "KLCSP_ITERATOR_ARRAY": [
      {
        "type": "params",
        "value": {
          "KLHST_WKS_DN": "Host1",
          "KLHST_WKS_HOSTNAME": "7ad995e2-eb62-40e5-9c7e-5abae19979a0",
          "KLHST_WKS_IP_LONG": {
            "type": "long",
            "value": 2130706433
          }
        }
      },
      {
        "type": "params",
        "value": {
          "KLHST_WKS_DN": "Host2",
          "KLHST_WKS_HOSTNAME": "5fb6a90c-d054-4f9b-a342-2a62949ad899",
          "KLHST_WKS_IP_LONG": {
            "type": "long",
            "value": 172052763
          }
        }
      }
    ]
  },
  "PkgRetVal": 2
}
```

В этом примере возвращается два элемента ("PkgRetVal" : 2) с полями, указанными в параметре **vecFieldsToReturn** запроса [HostGroup.FindHosts](#).

## Освобождение перечислителя

```
POST
https://MainKscServerIpAddress:Port/api/v1.0/ChunkAccessor.Release
X-KSC-VServer: Base64VirtualKscName
Content-Type: application/json
Content-Length: BodyLength
{"strAccessor" : "iteratorId"}
```

где:

- **MainKscServerIpAddress** - IP-адрес в формате IPv4 или полное доменное имя (FQDN) главного Сервера администрирования Kaspersky Security Center.
- **Port** - Порт OpenAPI Kaspersky Security Center на главном сервере администрирования. По умолчанию это порт **13299**.
- **Base64VirtualKscName** - имя виртуального Сервера администрирования в кодировке Base64. Заголовок X-KSC-VServer требуется, если метод вызывается для виртуального Сервера администрирования.
- **BodyLength** - длина JSON тела запроса в байтах.
- **iteratorId** - идентификатор перечислителя, полученный в поле **strAccessor** ответа метода [HostGroup.FindHosts](#).

## Работа с группами администрирования

### Поиск групп администрирования

```
POST
https://MainKscServerIpAddress:Port/api/v1.0/HostGroup.FindGroups
X-KSC-VServer: Base64VirtualKscName
Content-Type: application/json
Content-Length: BodyLength
{
  "wstrFilter" : "",
  "vecFieldsToReturn" : [
    "id",
    "name"  ],
  "vecFieldsToOrder" : [],
  "pParams" : {},
  "lMaxLifeTime" : 600
}
```

где:

- **MainKscServerIpAddress** – IP-адрес в формате IPv4 или полное доменное имя (FQDN) главного Сервера администрирования Kaspersky Security Center.
- **Port** – порт OpenAPI Kaspersky Security Center на главном сервере администрирования. По умолчанию это порт **13299**.
- **Base64VirtualKscName** – имя виртуального Сервера администрирования в кодировке Base64. Заголовок `X-KSC-VServer` требуется, если метод вызывается для виртуального Сервера администрирования.
- **BodyLength** - длина JSON тела запроса в байтах.
- **vecFieldsToReturn** – массив названий свойств группы администрирования, которые нужно вернуть. Полный список полей указан в документации к методу.
  - **id** – идентификатор группы.
  - **name** – имя группы.

Метод возвращает идентификатор объекта перечислителя [ChunkAccessor](#) на Сервере администрирования в виде:

```
{"strAccessor" : "iteratorId"}
```

По этому идентификатору вы можете получить информацию о группах администрирования, используя [методы работы с перечислителем](#). После завершения работы с перечислителем необходимо освободить перечислитель, вызвав метод [ChunkAccessor.Release](#).

### Получение идентификатора группы Управляемые устройства

```
POST
https://MainKscServerIpAddress:Port/api/v1.0/HostGroup.GroupIdGroups
X-KSC-VServer: Base64VirtualKscName
Content-Type: application/json
Content-Length: 2
{}
```

где:

- **MainKscServerIpAddress** – IP-адрес в формате IPv4 или полное доменное имя (FQDN) главного Сервера администрирования Kaspersky Security Center.
- **Port** – порт OpenAPI Kaspersky Security Center на главном Сервере администрирования. По умолчанию это порт **13299**.

- **Base64VirtualKscName** – имя виртуального Сервера администрирования в кодировке Base64. Заголовок `X-KSC-VServer` требуется, если метод вызывается для виртуального Сервера администрирования.

Метод возвращает идентификатор группы в виде:

```
{"PxcRetVal" : groupId}
```

## Получение идентификатора группы Нераспределенные устройства

```
POST
https://MainKscServerIpAddress:Port/api/v1.0/HostGroup.GroupIdUnassigned
X-KSC-VServer: Base64VirtualKscName
Content-Type: application/json
Content-Length: 2
{}
```

где:

- **MainKscServerIpAddress** – IP-адрес в формате IPv4 или полное доменное имя (FQDN) главного Сервера администрирования Kaspersky Security Center.
- **Port** – порт OpenAPI Kaspersky Security Center на главном сервере администрирования. По умолчанию это порт **13299**.
- **Base64VirtualKscName** – имя виртуального сервера администрирования в кодировке Base64. Заголовок `X-KSC-VServer` требуется, если метод вызывается для виртуального Сервера администрирования.

Метод возвращает идентификатор группы в виде:

```
{"PxcRetVal" : groupId}
```

## Получение списка виртуальных Серверов администрирования

```
POST
https://MainKscServerIpAddress:Port/api/v1.0/VServers.GetVServers
Content-Type: application/json
Content-Length: 2
{}
```

где:

- **MainKscServerIpAddress** – IP-адрес в формате IPv4 или полное доменное имя (FQDN) главного Сервера администрирования Kaspersky Security Center.
- **Port** – порт OpenAPI Kaspersky Security Center на главном Сервере администрирования. По умолчанию это порт **13299**.

В случае успеха метод возвращает массив виртуальных Серверов администрирования в виде:

```
{
  "PxcRetVal": [
    {
      "type": "params",
      "value": {
        "KLVSrv_CREATED": {
          "type": "datetime",
          "value": "2020-06-15T09:41:18Z"
        }
      }
    }
  ]
}
```

```

        "KLVSrv_DN": "SomeVirtualServer",
        "KLVSrv_ENABLED": true,
        "KLVSrv_GROUPS": 8,
        "KLVSrv_GRP": 0,
        "KLVSrv_HST_UID": "VSRV08172e1f-4057-4579-89c4-
d5e6256d8ad2",
        "KLVSrv_ID": 1,
        "KLVSrv_LIC_ENABLED": true,
        "KLVSrv_SUPER": 7,
        "KLVSrv_UID": "VSRV08172e1f-4057-4579-89c4-d5e6256d8ad2",
        "KLVSrv_UNASSIGNED": 11
    }
}
]
}

```

Важные параметры:

- **KLVSrv\_DN** – имя виртуального Сервера администрирования.
- **KLVSrv\_ID** – идентификатор виртуального Сервера администрирования.

## Работа с инсталляционными пакетами

### Получение списка инсталляционных пакетов

```

POST
https://MainKscServerIpAddress:Port/api/v1.0/PackagesApi.GetPackages
X-KSC-VServer: Base64VirtualKscName
Content-Type: application/json
Content-Length: 2
{}

```

где:

- **MainKscServerIpAddress** – IP-адрес в формате IPv4 или полное доменное имя (FQDN) главного Сервера администрирования Kaspersky Security Center.
- **Port** – порт OpenAPI Kaspersky Security Center на главном Сервере администрирования. По умолчанию это порт **13299**.
- **Base64VirtualKscName** – имя виртуального Сервера администрирования в кодировке Base64. Заголовок `X-KSC-VServer` требуется, если метод вызывается для виртуального Сервера администрирования.

В случае успеха метод возвращает массив пакетов в виде:

```

{
  "PkgRetVal": [
    {
      "type": "params",
      "value": {
        "KLPKG_NPI_CREATION_TIME": {
          "type": "datetime",
          "value": "2019-10-08T17:30:40Z"
        },
        "KLPKG_NPI_MODIF_TIME": {
          "type": "datetime",
          "value": "2019-10-08T17:30:40Z"
        },
        "KLPKG_NPI_NAME": "Kaspersky Security для виртуальных сред
5.2 Легкий агент",
        "KLPKG_NPI_PACKAGE_PATH": "\\\\"DESKTOP-

```

```

MI1CJOJ\\KLSHARE\\Packages\\KSVLA_5.2.27.1202",
  "KLPKG_NPI_PKGID": 1,
  "KLPKG_NPI_PRODUCT_DISPL_NAME": "Kaspersky Security для
виртуальных сред 5.2 Легкий агент",
  "KLPKG_NPI_PRODUCT_DISPL_VERSION": "5.2.27.1202",
  "KLPKG_NPI_PRODUCT_NAME": "KSVLA",
  "KLPKG_NPI_PRODUCT_VERSION": "5.2.0.0",
  "KLPKG_NPI_SIZE": {
    "type": "long",
    "value": 217427607
  },
  "KLPKG_NPI_SS_DESCR":
"KSVLA_5.2.27.1202\\exec\\ss_install.xml|1"
}
},
....
]
}

```

Важные параметры:

- **KLPKG\_NPI\_PKGID** – идентификатор инсталляционного пакета.
- **KLPKG\_NPI\_PRODUCT\_NAME** – название программы, которая устанавливается с помощью инсталляционного пакета.
- **KLPKG\_NPI\_PRODUCT\_VERSION** – версия программы, которая устанавливается с помощью инсталляционного пакета.

## Инсталляционные пакеты Агента администрирования Kaspersky Security Center и программы Kaspersky Security

Для идентификации пакетов Агента администрирования и программы Kaspersky Security для виртуальных сред 5.2 Легкий агент при вызове методов OpenAPI Kaspersky Security Center можно воспользоваться следующей таблицей:

Название свойства пакета	Агент администрирования	KSVLA 5.2
<b>KLPKG_NPI_PRODUCT_NAME</b>	1103	KSVLA
<b>KLPKG_NPI_PRODUCT_VERSION</b>	1.0.0.0	5.2.0.0

## Распространение инсталляционного пакета на виртуальный Сервер администрирования

```

POST
https://MainKscServerIpAddress:Port/api/v1.0/PackagesApi.Retranslate
ToVServerAsync
Content-Type: application/json
Content-Length: BodyLength
{
  "nPackageId": packageId,
  "nVServerId": VirtualServerId,
  "pOptions":
  {
    "KLPKG_CREATE_STANDALONE_PRODS" : true,
    "KLPKG_CREATE_STANDALONE_NAGT" : true,
    "KLPKG_USE_LANGUAGE_TAG" : true,
    "KLPKG_TYPE" : 1,
    "KLPKG_LAZY_RETRANSLATION" : false
  }
}

```

}

где:

**MainKscServerIpAddress** – IP-адрес в формате IPv4 или полное доменное имя (FQDN) главного Сервера администрирования Kaspersky Security Center.

**Port** – порт OpenAPI Kaspersky Security Center на главном Сервере администрирования. По умолчанию это порт **13299**.

**BodyLength** – длина JSON тела запроса в байтах.

**packageld** – идентификатор инсталляционного пакета, который необходимо распространить на виртуальный Сервер. Значение **KLPKG\_NPI\_PKGID**, полученное вызовом метода [PackagesApi.GetPackages](#).

**VirtualServerId** – идентификатор виртуального Сервера администрирования на который нужно распространить пакет. Значение **KLVSrv\_ID**, полученное вызовом метода [VServers.GetVServers](#).

В случае успеха метод возвращает идентификатор асинхронной операции в виде:

```
{"PkgRetVal" : "asyncActionId"}
```

Для проверки завершения асинхронной операции передачи пакета нужно вызвать метод [AsyncActionStateChecker.CheckActionState](#).

## Проверка состояния асинхронной операции

```
POST
https://MainKscServerIpAddress:Port/api/v1.0/AsyncActionStateChecker
.CheckActionState
Content-Type: application/json
Content-Length: BodyLength
{"wstrActionGuid": "asyncActionId"}
```

где:

**MainKscServerIpAddress** – IP-адрес в формате IPv4 или полное доменное имя (FQDN) главного Сервера администрирования Kaspersky Security Center.

**Port** – порт OpenAPI Kaspersky Security Center на главном Сервере администрирования. По умолчанию это порт **13299**.

**BodyLength** – длина JSON тела запроса в байтах.

**asyncActionId** – идентификатор асинхронной операции, полученный при вызове метода [PackagesApi.RetranslateToVServerAsync](#).

В случае успеха метод возвращает состояние асинхронной операции в виде:

```
{
  "bFinalized" : true
  "bSucceededFinalized" : true
  "lStateCode" : 0
  "pStateData": {}
  "lNextCheckDelay" : 0
}
```

где:

**bFinalized** – состояние операции. Если значение `true`, операция завершена.

**bSucceededFinalized** – признак успешного завершения операции. Если значение `true`, операция завершена успешно.

## Работа с задачами

### Создание задачи Смена Сервера администрирования

Задача позволяет сменить Сервер администрирования, под управлением которого находится виртуальная машина.

```
POST https://MainKscServerIpAddress:Port/api/v1.0/Tasks.AddTask
X-KSC-VServer: Base64VirtualKscName
Content-Type: application/json
Content-Length: BodyLength
{"pData" : taskParams}
```

где:

- **MainKscServerIpAddress** – IP-адрес в формате IPv4 или полное доменное имя (FQDN) главного Сервера администрирования Kaspersky Security Center.
- **Port** – порт OpenAPI Kaspersky Security Center на главном сервере администрирования Kaspersky Security Center. По умолчанию это порт **13299**.
- **Base64VirtualKscName** – имя виртуального Сервера администрирования в кодировке Base64. Заголовок X-KSC-VServer требуется, если метод вызывается для виртуального Сервера администрирования.
- **BodyLength** – длина JSON тела запроса в байтах.
- **taskParams** – параметры задачи Смены Сервера администрирования:

```
{ "pData" : {
  "PRTS_TASK_ENABLED": true,
  "TASKID_PRODUCT_NAME": "1103",
  "TASK_ADDITIONAL_PARAMS": {
    "type": "params",
    "value": {
      "ServerSslPorts": [ 13000 ],
      "Serverports": [ 14000 ],
      "ServerAddress": "NewKscServerAddress"
    }
  },
  "TASK_INFO_PARAMS": {
    "type": "params",
    "value": {
      "DisplayName": "Reconnect to vKSC",
      "HostList": [ {
        "type": "params",
        "value": {
          "HostDispName": "Host1",
          "HostName": "7ad995e2-eb62-40e5-9c7e-5abae19979a0"
        }
      }
    ]
  }
},
"TASKID_VERSION": "1.0.0.0",
"TASKSCH_TYPE": 0,
"TASK_NAME": "KLNAG_TASK_RECONNECTION",
"TASKID_COMPONENT_NAME": "86"
} }
```

где:

- **ServerSslPorts** – массив SSL-портов нового Сервера администрирования. По умолчанию один порт **13000**.

- **Serverports** – массив портов нового Сервера администрирования. По умолчанию один порт **14000**.
- **ServerAddress** – IP-адрес в формате IPv4 или полное доменное имя (FQDN) нового Сервера администрирования. Если виртуальная машина будет находиться под управлением виртуального Сервера администрирования, то значение должно иметь вид **MainKscServerAddress/VirtualServerName**.
- **DisplayName** – отображаемое имя задачи.
- **HostDispName** – имя виртуальной машины, которую нужно переместить на другой Сервер администрирования. Значение **KLHST\_WKS\_DN**, полученное вызовом метода [HostGroup.FindHosts](#).
- **HostName** – уникальный идентификатор виртуальной машины, которую нужно переместить на другой Сервер администрирования. Значение **KLHST\_WKS\_HOSTNAME**, полученное вызовом метода [HostGroup.FindHosts](#).

В случае успеха метод возвращает идентификатор созданной задачи в виде:

```
{"PkgRetVal" : "taskId"}
```

## Создание задачи удаленной установки программы

```
POST https://MainKscServerIpAddress:Port/api/v1.0/Tasks.AddTask
X-KSC-VServer: Base64VirtualKscName
Content-Type: application/json
Content-Length: BodyLength
{"pData" : taskParams}
```

где:

- **MainKscServerIpAddress** – IP-адрес в формате IPv4 или полное доменное имя (FQDN) главного Сервера администрирования Kaspersky Security Center.
- **Port** – порт OpenAPI Kaspersky Security Center на главном Сервере администрирования. По умолчанию это порт **13299**.
- **Base64VirtualKscName** – имя виртуального Сервера администрирования в кодировке Base64. Заголовок `X-KSC-VServer` требуется, если метод вызывается для виртуального Сервера администрирования.
- **BodyLength** – длина JSON тела запроса в байтах.
- **taskParams** – параметры задачи удаленной установки программы:

```
{"pData" : {
  "PRTS_TASK_ENABLED": true,
  "TASKID_PRODUCT_NAME": "1093",
  "TASK_ADDITIONAL_PARAMS": {
    "type": "params",
    "value": {
      "KLTSK_RI_USE_SHARE": true,
      "KLTSK_RI_USE_SHARE_SRV": true,
      "KLTSK_RI_PACKAGES_IDS": [
        KlnAgentPackageId,
        ProductPackageId
      ],
      "KLTSK_RI_USE_NAGENT": true,
      "KLTSK_RI_GROUP_TO_MOVE_HOST": GroupToMoveHostId,
      "klprts-TaskAccounts": [
        {
          "type": "params",
          "value": {
            "klprts-TaskAccountAuthType": 1
          }
        }
      ],
      "type": "params",
```



```

        "value": {
            "klprts-TaskAccountUser": "HostOsUserLogin"
        }
    },
    {
        "type": "params",
        "value": {
            "klprts-TaskAccountPassword": "HostOsUserPassword"
        }
    }
]
},
"TASK_INFO_PARAMS": {
    "type": "params",
    "value": {
        "DisplayName": "Install KSVLA 5.2.0.0 on host",
        "HostList": [
            {
                "type": "params",
                "value": {
                    "HostDispName": "Host1",
                    "HostName": "7ad995e2-eb62-40e5-9c7e-
5abae19979a0"
                }
            }
        ]
    }
},
"TASKID_VERSION": "1.0.0.0",
"TASKSCH_TYPE": 0,
"TASK_NAME": "Remote Installation",
"TASKID_COMPONENT_NAME": "87"
} }

```

где:

- **KINagentPackageld** – идентификатор инсталляционного пакета Агента администрирования, найденный с помощью вызова метода [PackagesApi.GetPackages](#). Пакет должен находиться на Сервере администрирования, на котором находится виртуальная машина. Для передачи пакетов на нужный Сервер администрирования используйте метод [PackagesApi.RetranslateToVServerAsync](#).
- **ProductPackageld** – идентификатор инсталляционного пакета Kaspersky Security для виртуальных сред 5.2 Легкий агент, найденный с помощью вызова метода [PackagesApi.GetPackages](#). Пакет должен находиться на Сервере администрирования, на котором находится виртуальная машина. Для передачи пакетов используйте метод [PackagesApi.RetranslateToVServerAsync](#).
- **GroupToMoveHostId** – идентификатор группы администрирования, в которую нужно переместить виртуальную машину после установки программы. Для поиска группы используйте метод [HostGroup.FindGroups](#) или [HostGroups.GroupIdGroups](#).
- **HostOsUserLogin** – имя пользователя, от имени которого будет осуществляться установка программы.
- **HostOsUserPassword** – пароль пользователя, от имени которого будет осуществляться установка программы.
- **DisplayName** – отображаемое имя задачи.
- **HostDispName** – имя виртуальной машины, на которую надо установить программу. Значение **KLHST\_WKS\_DN**, полученное вызовом метода [HostGroup.FindHosts](#).
- **HostName** – уникальный идентификатор виртуальной машины, на которую надо установить программу. Значение **KLHST\_WKS\_HOSTNAME**, полученное вызовом метода [HostGroup.FindHosts](#).

В случае успеха метод возвращает идентификатор созданной задачи в виде:

```
{"PkgRetVal" : "taskId"}
```

## Создание задачи удаленной деинсталляции программы

```
POST https://MainKscServerIpAddress:Port/api/v1.0/Tasks.AddTask
X-KSC-VServer: Base64VirtualKscName
Content-Type: application/json
Content-Length: BodyLength
{"pData" : taskParams}
```

где:

- **MainKscServerIpAddress** – IP-адрес в формате IPv4 или полное доменное имя (FQDN) главного Сервера администрирования Kaspersky Security Center.
- **Port** – порт OpenAPI Kaspersky Security Center на главном сервере администрирования. По умолчанию это порт **13299**.
- **Base64VirtualKscName** – имя виртуального сервера администрирования в кодировке Base64. Заголовок X-KSC-VServer требуется, если метод вызывается для виртуального сервера.
- **BodyLength** – длина JSON тела запроса в байтах.
- **taskParams** – параметры задачи удаленной деинсталляции программы:

```
{"pData" : {
  "PRTS_TASK_ENABLED": true,
  "TASKID_PRODUCT_NAME": "1093",
  "TASK_ADDITIONAL_PARAMS": {
    "type": "params",
    "value": {
      "klprts-TaskAccounts": [
        {
          "type": "params",
          "value": {
            "klprts-TaskAccountAuthType": 1
          }
        },
        {
          "type": "params",
          "value": {
            "klprts-TaskAccountUser":
"HostOsUserLogin"
          }
        },
        {
          "type": "params",
          "value": {
            "klprts-TaskAccountPassword":
"HostOsUserPassword"
          }
        }
      ],
      "KLTSK_RI_USE_SHARE_SRV": true,
      "KLTSK_RI_USE_SHARE": true,
      "ProductVersion": "5.2.0.0",
      "KLTSK_RI_USE_NAGENT": true,
      "ProductName": "KSVLA",
      "UninstallType": 0
    }
  },
  "TASK_INFO_PARAMS": {
    "type": "params",
```

```

    "value": {
      "DisplayName": "Deinstall KSVLA 5.2.0.0 on host",
      "HostList": [
        {
          "HostDispName": "Host1",
          "HostName": "7ad995e2-eb62-40e5-9c7e-
5abae19979a0"
        }
      ]
    },
    "TASKID_VERSION": "1.0.0.0",
    "TASKSCH_TYPE": 0,
    "TASK_NAME": "Remote Deinstallation",
    "TASKID_COMPONENT_NAME": "87"
  } }

```

где:

- **HostOsUserLogin** – имя пользователя, от имени которого будет осуществляться удаление программы.
- **HostOsUserPassword** – пароль пользователя, от имени которого будет осуществляться удаление программы.
- **ProductVersion** – название удаляемой программы.
- **ProductName** – версия удаляемой программы.
- **DisplayName** – отображаемое имя задачи.
- **HostDispName** – имя виртуальной машины, на которой надо удалить программу. Значение **KLHST\_WKS\_DN**, полученное вызовом метода [HostGroup.FindHosts](#).
- **HostName** – уникальный идентификатор виртуальной машины, на которой надо удалить программу. Значение **KLHST\_WKS\_HOSTNAME**, полученное вызовом метода [HostGroup.FindHosts](#).

В случае успеха метод возвращает идентификатор созданной задачи в виде:

```
{"PxcRetVal" : "taskId"}
```

## Запуск задачи

```

POST https://MainKscServerIpAddress:Port/api/v1.0/Tasks.RunTask
X-KSC-VServer: Base64VirtualKscName
Content-Type: application/json
Content-Length: BodyLength
{"strTask" : "taskId"}

```

где:

**MainKscServerIpAddress** – IP-адрес в формате IPv4 или полное доменное имя (FQDN) главного Сервера администрирования Kaspersky Security Center.

**Port** - Порт OpenAPI Kaspersky Security Center на главном сервере администрирования. По умолчанию это порт **13299**.

**Base64VirtualKscName** - имя виртуального сервера администрирования в кодировке Base64. Заголовок **X-KSC-VServer** требуется, если метод вызывается для виртуального Сервера администрирования.

**BodyLength** - длина JSON тела запроса в байтах.

**taskId** – идентификатор задачи, полученный после ее создания.