

# Руководство по использованию REST API

Интерфейс платформы **Invarivision** предоставляет возможность внешним разработчикам программными средствами получать и записывать данные с использованием REST API.

## Параметры запроса

В каждом запросе должен присутствовать набор обязательных параметров.

Имя	Тип	Описание
method	string	название вызываемого метода, например, video.list; обязательный параметр
session_key	string	сессия текущего пользователя

В ответе в поле "status" возвращается статус выполнения операции, и если произошла ошибка (status = error), то дополнительную информацию можно получить из полей "description", "err\_signature", "err\_type" (см. [таблицу 1](#)).

## Авторизация запроса

Сессия (session\_key) получается при каждом новом сеансе работы пользователя с вашим приложением или сайтом. При последующих заходах того же пользователя это значение будет другим, поэтому сохранять его не надо. Значение session\_key получается в процессе логина.

## Получение сессии приложений

Для получения сессии приложение должно сделать POST-запрос на адрес <https://tracker2.invarivision.com/api.php>. Например:

```
> POST /oauth/token HTTP/1.1
> Host: tracker2.invarivision.com
> Content-Type: application/x-www-form-urlencoded
>
> grant_type=password&username=test@mail.ru&password=qwerty
```

- grant\_type — строка «password»
- username — email пользователя
- password — пароль пользователя

## Формат ответа

В случае, если все параметры указаны правильно, сервер возвращает сессию (access\_token) и вспомогательные параметры:

```
{
  "method": "login",
```

```
"status": "finished",
"description": "Authorization is successful.",
"access_token": "e5dd32be039e48c08e0331f917421cdb",
"expires_in": "71603",
"user_id": "55",
"refresh_token": "0349e84853427020297183421e3d6e5d"
}
```

- access\_token — сессия приложения
- expires\_in — срок годности access\_token'a в секундах
- refresh\_token — refresh\_token, позволяющий совершить повторную авторизацию без логина и пароля пользователя
- user\_id — id пользователя, которому принадлежат данные логин и пароль; используется при подписи запросов и может использоваться везде, где необходимо указать пользователя

## Использование refresh\_token

Приложениям запрещается сохранять пароль пользователя. Для того, чтобы обеспечить повторное получение сессии без запроса пароля, используется механизм refresh\_token. При успешной авторизации с помощью логина и пароля пользователя сервер возвращает значение refresh\_token, которое можно сохранить на клиенте и использовать при последующих запусках программы вместо логина и пароля для получения новой сессии.

```
> POST /oauth/token HTTP/1.1
> Host: tracker2.invarivision.com
> Content-Type: application/x-www-form-urlencoded
>
> grant_type=refresh_token&
  refresh_token=0349e84853427020297183421e3d6e5d

< HTTP/1.1 200 OK
< Content-Type: application/json
<
< { "method": "refresh_token",
    "status": "finished",
    "description": "Authorization is successful.",
    "access_token": "e5dd32be039e48c08e0331f917421cdb",
    "expires_in": "62911",
    "user_id": "55",
    "refresh_token": "0349e84853427020297183421e3d6e5d" }
```

Как видно из примера, ответ на запрос с использованием refresh\_token'a возвращается такого же вида, как и при авторизации с помощью логина и пароля.

# Функции REST API

## video.add

Добавляет в систему исходный видеоролик для дальнейшего поиска совпадений по содержимому видео.

### Параметры

Имя	Тип	Описание
file_url	string	URL файла видеоролика.

### Результат

Формат ответа для json-выдачи:

```
{
  "method": название метода "video.add"
  "file": имя файла видеоролика
  "status": статус операции (finished, error)
  "description": текстовое описание результата выполнения операции
  "film_id": идентификационный номер видеоролика
}
```

### Пример вызова

```
http://tracker2.invarivision.com/api.php?method=video.add&
file_url=http://tracker.invarivision.com/reg/tv/sv1.avi&
session_key=e5dd32be039e48c08e0331f917421cdb
```

### Пример ответа в формате JSON

```
{
  "method": "video.add",
  "file": "sv1.avi",
  "status": "finished",
  "description": "Your video <./upload/sv1.avi> was successfully added to the searching
system.",
  "film_id": 1053
}
```

## video.scan

Выполняет сканирование видеоролика на совпадения его содержимого с ранее загруженным в систему исходным видео.

## Параметры

Имя	Тип	Описание
file_url	string	URL файла видеоролика.

## Результат

Формат ответа для json-выдачи:

```
{
  "method": название метода "video.scan"
  "file": имя файла видеоролика
  "status": статус операции (finished, error)
  "intersections": коэффициент совпадений содержимого видео с ранее загруженным
  "description": текстовое описание результата выполнения операции
  "results": массив описателей совпавших фрагментов
  [
    {
      "film_name": название исходного фильма, содержимое которого найдено
      "film_id": идентификационный номер исходного фильма
      "film_url": URL исходного фильма
      "film_length": длина исходного видеоролика (часы:минуты:секунды)
      "channel_name": название сканируемого видеоролика, в котором найдены совпадения
      "channel_url": URL сканируемого видео
      "channel_position": позиция найденного фрагмента в сканируемом видео (Ч:М:С)
      "film_position": позиция найденного фрагмента в исходном видео (Ч:М:С)
      "fragment_length": длина найденного совпавшего фрагмента видео (Ч:М:С)
      "found_date_time": локальное дата и время обнаружения фрагмента
    }, ...
  ],
  "gathered": массив альтернативных результатов с объединёнными видеофрагментами
  [
    "gathered_fragments": количество объединённых фрагментов
    "intersections": коэффициент совпадений содержимого видео с ранее загруженным
    "results": массив описателей совпавших фрагментов
    [
      { ... }, ...
    ]
  ],
  "compared_films": список оригинальных фильмов, которые имеют совпадения со сканированным файлом
  [
    {
```

"film\_name": название исходного фильма  
"film\_id": идентификационный номер исходного фильма  
"film\_length": длина исходного видеоролика (часы:минуты:секунды)  
"film\_url": URL исходного фильма  
"added\_date\_time": дата и время добавления фильма в систему  
"intersections": коэффициент совпадений сканированного ролика с исходным фильмом  
},  
...  
]  
}

### Пример вызова

[http://tracker2.invarivision.com/api.php?method=video.scan&file\\_url=http://tracker.invarivision.com/reg/tv/sv\\_test.avi&session\\_key=e5dd32be039e48c08e0331f917421cdb](http://tracker2.invarivision.com/api.php?method=video.scan&file_url=http://tracker.invarivision.com/reg/tv/sv_test.avi&session_key=e5dd32be039e48c08e0331f917421cdb)

### Пример ответа в формате JSON

```
{
  "method": "video.scan",
  "file": "sv_test.avi",
  "status": "finished",
  "intersections": "0.654218",
  "description": "Intersections with other video content were found: 65.4218 %.",
  "results":
  [
    {
      "film_name": "sv2_avi",
      "film_id": "1061",
      "film_url": "http://tracker2.invarivision.com/tv/sv2_avi",
      "film_length": "00:03:04",
      "channel_name": "sv_test_avi",
      "channel_url": "http://tracker2.invarivision.com/tv/sv_test_avi",
      "channel_position": "0:01:52",
      "film_position": "00:01:26",
      "fragment_length": "00:00:05",
      "found_date_time": "2015-03-31 14:13:09"
    }, ...
  ],
  "gathered":
  [
    "gathered_fragments": 0,
    "intersections": "0.654218",
    "results":
```

```
[...
{
  "film_name": "sv2_avi",
  "film_id": "1061",
  "film_url": "http://tracker2.invarivision.com/tv/sv2_avi",
  "film_length": "00:03:04",
  "channel_name": "sv_test_avi",
  "channel_url": "http://tracker2.invarivision.com/tv/sv_test_avi",
  "channel_position": "0:01:52",
  "film_position": "00:01:26",
  "fragment_length": "00:00:05",
  "found_date_time": "2015-03-31 14:13:09"
}, ...
]
],
"compared_films":
[
{
  "film_name": "sv2_avi",
  "film_id": 1061,
  "film_length": "00:03:04",
  "film_url": "http://tracker2.invarivision.com/tv/sv2_avi",
  "added_date_time": "2015-03-20 11:21:55",
  "intersections": 0.227
},
...
]
}
```

## video.delete

Удаляет из системы исходный видеоролик, используемый для поиска совпадений содержимого видео.

### Параметры

Имя	Тип	Описание
file_url	string	URL или имя файла видеоролика;
file_id	uint	ID файла видеоролика (нужен URL либо ID файла)

### Результат

Формат ответа для json-выдачи:

```
{  
  "method": название метода "video.delete"  
  "file": имя файла видеоролика  
  "status": статус операции (finished, error)  
  "description": текстовое описание результата выполнения операции  
}
```

### Пример вызова #1

```
http://tracker2.invarivision.com/api.php?method=video.delete&  
file_url=http://tracker.invarivision.com/reg/tv/sv1.avi&  
session_key=e5dd32be039e48c08e0331f917421cdb
```

### Пример вызова #2

```
http://tracker2.invarivision.com/api.php?method=video.delete&  
file_id=1072&session_key=e5dd32be039e48c08e0331f917421cdb
```

### Пример ответа в формате JSON

```
{  
  "method": "video.delete",  
  "file": "sv1.avi",  
  "status": "finished",  
  "description": "The film is deleted."  
}
```

## video.list

Возвращает список загруженных в систему исходных видеороликов.

### Результат

Формат ответа для json-выдачи:

```
{  
  "method": название метода "video.list"  
  "total": общее количество видеороликов в списке  
  "list": массив описателей загруженных видеороликов  
  [  
    {  
      "film_name": название исходного видеоролика  
      "film_link": настоящее имя файла видеоролика  
      "length": длина исходного видеоролика (часы:минуты:секунды)  
      "date_time": дата и время добавления видеоролика  
    }, ...  
  ]  
}
```

```
]
}
```

### Пример вызова

[http://tracker2.invarivision.com/api.php?method=video.list&session\\_key=e5dd32be039e48c08e0331f917421cdb](http://tracker2.invarivision.com/api.php?method=video.list&session_key=e5dd32be039e48c08e0331f917421cdb)

### Пример ответа в формате JSON

```
{
  "method": "video.list",
  "user_id": "55"
  "total": 4,
  "list":
  [
    {"film_name": "sv1_avi", "film_link": "sv1.avi", "length": "0:03:06", "date_time": "2015-03-31 13:58:42"},
    {"film_name": "sv2_avi", "film_link": "sv2.avi", "length": "0:03:04", "date_time": "2015-03-31 14:07:40"},
    {"film_name": "sv3_avi", "film_link": "sv3.avi", "length": "0:03:24", "date_time": "2015-03-31 14:10:06"},
    {"film_name": "sv4_avi", "film_link": "sv4.avi", "length": "0:03:01", "date_time": "2015-03-31 14:11:50"}
  ]
}
```

## logout

Завершение работы сессии. После выполнения этой операции параметры `access_token` и `refresh_token`, которые были получены после логина становятся недействительными.

### Пример вызова

[http://tracker2.invarivision.com/api.php?method=logout&session\\_key=e5dd32be039e48c08e0331f917421cdb](http://tracker2.invarivision.com/api.php?method=logout&session_key=e5dd32be039e48c08e0331f917421cdb)

### Пример ответа в формате JSON

```
{
  "method": "logout",
  "status": "finished",
  "description": "The session has been closed."
}
```

## Асинхронный API

### video.add.start

Запускает фоновый процесс добавления в систему исходного видеоролика для дальнейшего поиска совпадений по содержимому видео.



## Параметры

Имя	Тип	Описание
file_url	string	URL файла видеоролика.
streaming	uint	обработка в режим HLS потока (по умолчанию - отключено "0")
callback	string	URL скрипта, который должен быть вызван по завершению процесса (необязательный параметр)

## Результат

Формат ответа для json-выдачи:

```
{
  "method": название метода "video.add.start"
  "file": имя файла видеоролика
  "status": статус операции (running)
  "handle": идентификатор запущенного процесса
}
```

## Пример вызова

[http://tracker2.invarivision.com/api.php?method=video.add.start&file\\_url=http://tracker.invarivision.com/reg/tv/sv1.avi&session\\_key=e5dd32be039e48c08e0331f917421cdb](http://tracker2.invarivision.com/api.php?method=video.add.start&file_url=http://tracker.invarivision.com/reg/tv/sv1.avi&session_key=e5dd32be039e48c08e0331f917421cdb)

## Пример ответа в формате JSON

```
{
  "method": "video.add.start",
  "file": "sv1.avi",
  "status": "running",
  "handle": "774cafcaa6cb370f8be28401e6ed1d3e"
}
```

## video.scan.start

Запускает фоновый процесс сканирования видеоролика на совпадения его содержимого с ранее загруженным в систему исходным видео.

## Параметры

Имя	Тип	Описание
file_url	string	URL файла видеоролика.

streaming	uint	обработка в режим HLS потока (по умолчанию - отключено "0")
callback	string	URL скрипта, который должен быть вызван по завершению процесса (необязательный параметр)

## Результат

Формат ответа для json-выдачи:

```
{
  "method": название метода "video.scan.start"
  "file": имя файла видеоролика
  "status": статус операции (running)
  "handle": идентификатор запущенного процесса
}
```

## Пример вызова

[http://tracker2.invarivision.com/api.php?method=video.scan.start&file\\_url=http://tracker.invarivision.com/reg/tv/sv\\_test.avi&session\\_key=e5dd32be039e48c08e0331f917421cdb](http://tracker2.invarivision.com/api.php?method=video.scan.start&file_url=http://tracker.invarivision.com/reg/tv/sv_test.avi&session_key=e5dd32be039e48c08e0331f917421cdb)

## Пример ответа в формате JSON

```
{
  "method": "video.scan.start",
  "file": "sv1.avi",
  "status": "running",
  "handle": "0f955a699f725f57458b7bded06d1bcb"
}
```

## get.status

Возвращает текущее состояние или результат выполнения запущенного фонового процесса аналогичный тому, который возвращает метод [video.add](#) и [video.scan](#).

## Параметры

Имя	Тип	Описание
handle	string	идентификатор процесса

## Результат

Формат ответа для json-выдачи (текущее состояние процесса):

```
{
  "method": название метода выполняемого процесса (video.add или video.scan)
```

"status": статус (waiting, running, downloading, launching, scanning, adding, preparing, finished, error)

"task\_progress": прогресс выполнения в процентах

// Если streaming = 1, то здесь добавляются дополнительные поля:

"intersections": коэффициент совпадений содержимого видео с ранее загруженным

"fragment": описатель совпадающего фрагмента

```
{
  "film_id": идентификационный номер исходного фильма
  "channel_position": позиция найденного фрагмента в сканируемом видео (Ч:М:С)
  "film_position": позиция найденного фрагмента в исходном видео (Ч:М:С)
  "fragment_length": длина найденного совпавшего фрагмента видео (Ч:М:С)
}
```

## Пример вызова

```
http://tracker2.invarivision.com/api.php?method=get.status&
handle=24c3adb754f1ed3385c899b0e250188c&
session_key=e5dd32be039e48c08e0331f917421cdb
```

## Пример ответа в формате JSON

```
{
  "method": "video.scan",
  "status": "downloading",
  "task_progress": "2"
}
```

## get.results

Возвращает результат сканирования видеофайла аналогичный тому, который возвращает метод [video.add](#) и [video.scan](#).

## Параметры

Имя	Тип	Описание
file_url	string	имя файла или его URL

## Результат

Формат ответа для json-выдачи:

```
{
  "method": название метода "get.results"
  "file": имя файла видеоролика
  "status": статус операции (finished, error)
```

"intersections": коэффициент совпадений содержимого видео с ранее загруженным

"description": текстовое описание результата выполнения операции

"results": массив описателей совпавших фрагментов

[

{

"film\_name": название исходного фильма, содержимое которого найдено

"film\_id": идентификационный номер исходного фильма

"film\_url": URL исходного фильма

"film\_length": длина исходного видеоролика (часы:минуты:секунды)

"channel\_name": название сканируемого видеоролика, в котором найдены совпадения

"channel\_url": URL сканируемого видео

"channel\_position": позиция найденного фрагмента в сканируемом видео (Ч:М:С)

"film\_position": позиция найденного фрагмента в исходном видео (Ч:М:С)

"fragment\_length": длина найденного совпавшего фрагмента видео (Ч:М:С)

"found\_date\_time": локальная дата и время обнаружения фрагмента

}, ...

],

"gathered": массив альтернативных результатов с объединёнными видеофрагментами

[

"gathered\_fragments": количество объединённых фрагментов

"intersections": коэффициент совпадений содержимого видео с ранее загруженным

"results": массив описателей совпавших фрагментов

[

{ ... }, ...

]

],

"compared\_films": список оригинальных фильмов, которые имеют совпадения со сканированным файлом

[

{

"film\_name": название исходного фильма

"film\_id": идентификационный номер исходного фильма

"film\_length": длина исходного видеоролика (часы:минуты:секунды)

"film\_url": URL исходного фильма

"added\_date\_time": дата и время добавления фильма в систему

"intersections": коэффициент совпадений сканированного ролика с исходным фильмом

},

...

]

}

## Пример вызова

<http://tracker2.invarivision.com/api.php?method=get.results&>

file\_url=http://tracker.invarivision.com/reg/tv/sv\_test.avi&  
session\_key=e5dd32be039e48c08e0331f917421cdb

## Пример ответа в формате JSON

```
{
  "method": "get.results",
  "file": "sv_test.avi",
  "status": "finished",
  "intersections": "0.654218",
  "description": "Intersections with other video content were found: 65.4218 %.",
  "results":
  [...
  {
    "film_name": "sv2_avi",
    "film_id": "1061",
    "film_url": "http://tracker2.invarivision.com/tv/sv2_avi",
    "film_length": "00:03:04",
    "channel_name": "sv_test_avi",
    "channel_url": "http://tracker2.invarivision.com/tv/sv_test_avi",
    "channel_position": "0:01:52",
    "film_position": "00:01:26",
    "fragment_length": "00:00:05",
    "found_date_time": "2015-03-31 14:13:09"
  }, ...
  ]
  "gathered":
  [
    "gathered_fragments": 0,
    "intersections": "0.654218",
    "results":
    [...
    {
      "film_name": "sv2_avi",
      "film_id": "1061",
      "film_url": "http://tracker2.invarivision.com/tv/sv2_avi",
      "film_length": "00:03:04",
      "channel_name": "sv_test_avi",
      "channel_url": "http://tracker2.invarivision.com/tv/sv_test_avi",
      "channel_position": "0:01:52",
      "film_position": "00:01:26",
      "fragment_length": "00:00:05",
      "found_date_time": "2015-03-31 14:13:09"
    }, ...
  ]
}
```

```
]
],
"compared_films":
[
{
"film_name": "sv2_avi",
"film_id": 1061,
"film_length": "00:03:04",
"film_url": "http://tracker2.invarivision.com/tv/sv2_avi",
"added_date_time": "2015-03-20 11:21:55",
"intersections": 0.227
},
...
]
}
```

## close

Закрывает описатель процесса, удаляя при этом с сервера всю информацию о результатах завершения данного процесса.

### Параметры

Имя	Тип	Описание
handle	string	идентификатор процесса

### Пример вызова

```
http://tracker2.invarivision.com/api.php?method=close&
handle=24c3adb754f1ed3385c899b0e250188c&
session_key=e5dd32be039e48c08e0331f917421cdb
```

### Пример ответа в формате JSON

```
{
"method": "close",
"status": "finished",
"description": "The handle is closed."
}
```

## callback

Если при запуске фоновых процессов был указан URL адрес обратного вызова, то по завершению - процесс обратится к скрипту указанному в URL с параметрами **handle** и **session\_key**. Далее вызванный скрипт может получить результаты завершения процесса с помощью метода **get.status**. После получения результатов необходимо вызвать метод **close** для удаления информации о завершившемся процессе с сервера.

## Таблица кодов ошибок

Таблица 1

#	err_signature	err_type	description
1	operation_failed	retry_operation	Processing of < <i>filename</i> > was failed
2	cannot_open_file	permanent	Can't open file < <i>filename</i> >
3	too_small_duration	permanent	Too small video file, length is XXXX ms
4	download_error	retry_operation	Download error> fopen( <i>URL</i> ): failed to open stream
5	login_error	retry_login	Sorry, this user is not found.
6	token_not_exist	retry_login	This refresh token is not exist.
7	session_expired	retry_login	The session time is expired.
8	session_not_valid	retry_login	The session key is not valid.
9	session_absent	permanent	The session key is absent.
10	empty_url	permanent	Sorry but your URL is empty.
11	film_not_found	permanent	The film < <i>filename</i> > was not found.
12	handle_not_exist	permanent	This handle is not exist.
13	method_not_defined	permanent	The method < <i>method</i> > is not defined.