



АО «КАЛУГА АСТРАЛ»

**Инструкция по эксплуатации
ПП «Астрал. Сервер электронной отчетности»**

Калуга

СОДЕРЖАНИЕ

1. АРІ ОБРАБОТЧИКА ФНС	3
1.1. ПРИЕМ ПАКЕТА ОТ АБОНЕНТА.....	3
1.2. ПОЛУЧЕНИЕ СПИСКА ПАКЕТОВ.....	4
1.3. СКАЧИВАНИЕ ПАКЕТА.....	5
1.4. УДАЛЕНИЕ ПАКЕТА.....	5
1.5. ПОЛУЧЕНИЕ СПИСКА МАССОВЫХ РАССЫЛОК.....	6
2. АРІ ОБРАБОТЧИКА АИС ПФР 2.....	8
2.1. ПОЛУЧЕНИЕ СПИСКА ПАКЕТОВ.....	8
2.2. ПРИЕМ ПАКЕТА ОТ АБОНЕНТА.....	8
2.3. СКАЧИВАНИЕ ПАКЕТА.....	9
2.4. УДАЛЕНИЕ ПАКЕТА.....	9
2.5. ОПИСАНИЕ ФОРМАТА ОШИБОК.....	9
3. АРІ ОБРАБОТЧИКА РОССТАТ	11
3.1. ПРИЕМ ПАКЕТА ОТ АБОНЕНТА.....	11
3.2. ПОЛУЧЕНИЕ СПИСКА ПАКЕТОВ.....	11
3.3. ПОЛУЧЕНИЕ СПИСКА РАССЫЛОК.....	12
3.4. СКАЧИВАНИЕ ПАКЕТА.....	12
3.5. УДАЛЕНИЕ ПАКЕТА.....	13
3.6. ОПИСАНИЕ ФОРМАТА ОШИБОК.....	13

1. API обработчика ФНС

1.1. Прием пакета от абонента

Тип: POST

Путь: /packages

Заголовки:

- X-Real-IP: <ip-адрес отправителя пакета>
- X-Send-Date: <дата и время отправки запроса на сервер оператора>
- Content-Type: multipart/form-data

Тело:

Имя	Тип	Описание	Обязательность
file	file	пакет	+

При получении контейнера сервер ожидает в заголовках запроса получить атрибуты X-Real-IP и X-Send-Date. Если таковых нет в запросе, то они сформируются автоматически: дата и время берется текущая серверная на момент приема пакета, ip-адрес отправителя берется из стандартного заголовка X-Forwarded-For, но это не гарантирует, что сервером оператора будет получен корректный ip-адрес, с которого клиент действительно отправлял отчет.

Ответ: JSON-объект с данными о принятом сервером пакете

Имя	Описание
id	Идентификатор пакета, который оператор присвоил ему при приеме.
filename	Имя принятого пакета
client_ip	ip-адрес, с которого клиент отправил пакет в сторону оператора
date_in	Дата приема пакета оператором

Пример:

```
1{
2  "id": "7f281c1a-6113-49c8-a86e-51fab9102687",
3  "filename": "FNS_9965_1AE71D20EE1-D9F0-4C9F-920F-8A8E705A2A3D_d7ab32e90692dece7392c63f6c754c34_07_01_01.zip",
4  "client_ip": "82.113.1.187",
5  "date_in": "2022-03-12 14:56:17"
6}
```

1.2. Получение списка пакетов

Тип: GET

Путь: /packages

Параметры запроса:

Имя	Описание	Обязательный
type	тип продукта (может принимать значения 1с, ao5, ao4 и platform)	+
guid	гуид абонента (<префикс СОС><гуид продукта>)	-
gate	гейт (один гуид продукта может принадлежать только единственному гейту в рамках направления)	-

При запросе всегда ожидается наличие параметра type. Если параметр type имеет значение ao4 и при этом параметр guid не имеет значения, то сервис вернет ошибку.

Ответ: JSON-объект, представляющий из себя список объектов с информацией о доступных к получению контейнеров абонентов.

Примеры:

- /packages?type=1с
1 [
2 {
3 "guid": "1AE71D20EE1-D9F0-4C9F-920F-8A8E705A2A3D",
4 "uid": "632464bdf97e4e1aa7308c88e6e566fd",
5 "filename": "FNS_9965_1AE71D20EE1-D9F0-4C9F-920F-
8A8E705A2A3D_d7ab32e90692dece7392c63f6c754c34_07_01_01.zip",
6 "link": "files/gpr/71D20EE1-D9F0-4C9F-920F-
8A8E705A2A3D/632464bdf97e4e1aa7308c88e6e566fd/FNS_9965_1AE71D20EE1-
D9F0-4C9F-920F-
8A8E705A2A3D_d7ab32e90692dece7392c63f6c754c34_07_01_01.zip"
7 },
8 {
9 "guid": "1AE37D52A72-7B0E-45C5-93DE-4C7F313E6C56",
10 "uid": "55b078415bc74b04941f5358f741dc7d",
11 "filename": "FNS_9965_1AE37D52A72-7B0E-45C5-93DE-
4C7F313E6C56_d5df2b51270104dbbf61bc4aa718fc4d_07_01_01.zip",
12 "link": "files/gpr/37D52A72-7B0E-45C5-93DE-
4C7F313E6C56/55b078415bc74b04941f5358f741dc7d/FNS_9965_1AE37D52A72-
7B0E-45C5-93DE-
4C7F313E6C56_d5df2b51270104dbbf61bc4aa718fc4d_07_01_01.zip"
13 }
14]

- /packages?type=ao4&guid=1AE71D20EE1-D9F0-4C9F-920F-8A8E705A2A3D
1[
2 {
3 "guid": "1AE71D20EE1-D9F0-4C9F-920F-8A8E705A2A3D",
4 "uid": "632464bdf97e4e1aa7308c88e6e566fd",
5 "filename": "FNS_9965_1AE71D20EE1-D9F0-4C9F-920F-
8A8E705A2A3D_d7ab32e90692dece7392c63f6c754c34_07_01_01.zip",
6 "link": "files/gpr/71D20EE1-D9F0-4C9F-920F-
8A8E705A2A3D/632464bdf97e4e1aa7308c88e6e566fd/FNS_9965_1AE71D20EE1-
D9F0-4C9F-920F-
8A8E705A2A3D_d7ab32e90692dece7392c63f6c754c34_07_01_01.zip"
7 },
8 {
9 "guid": "1AE71D20EE1-D9F0-4C9F-920F-8A8E705A2A3D",
10 "uid": "632464bdf97e4e1aa7308c88e6e566fd",
11 "filename": "FNS_9965_1AE71D20EE1-D9F0-4C9F-920F-
8A8E705A2A3D_d7ab32e90692dece7392c63f6c754c34_07_01_01.zip",
12 "link": "files/gpr/71D20EE1-D9F0-4C9F-920F-
8A8E705A2A3D/632464bdf97e4e1aa7308c88e6e566fd/FNS_9965_1AE71D20EE1-
D9F0-4C9F-920F-
8A8E705A2A3D_d7ab32e90692dece7392c63f6c754c34_07_01_01.zip"
13 }
14]
- /packages?type=ao4.5
1{
2 "errors": [
3 "\"type\" type must be one of this list: ['lc', 'iotchet',
'platform', 'ao4', 'ao5']"
4],
5 "status": "Bad Request"
6}

1.3. Скачивание пакета

Тип: GET

Путь: /[[link]]

Описание:

В методах получения списка пакетов (/packages, /massreport) возвращаются пути (поле link), по которым можно контейнеры забрать. Эти пути универсальны и их нужно просто присоединить к адресу сервера. Балансировщик сам найдет контейнер и отдаст его содержимое.

Примеры:

- files/gpr/71D20EE1-D9F0-4C9F-920F-
8A8E705A2A3D/632464bdf97e4e1aa7308c88e6e566fd/FNS_9965_1AE71D20EE1-
D9F0-4C9F-920F-
8A8E705A2A3D_d7ab32e90692dece7392c63f6c754c34_07_01_01.zip

Результат: диалог скачивания пакета

1.4. Удаление пакета

Тип: DELETE

Описание:

После успешного получения пакета, следует воспользоваться отдельным методом для удаления его из списка и проставления даты.

Примеры:

- files/gpr/71D20EE1-D9F0-4C9F-920F-8A8E705A2A3D/632464bdf97e4e1aa7308c88e6e566fd/FNS_9965_1AE71D20EE1-D9F0-4C9F-920F-8A8E705A2A3D_d7ab32e90692dece7392c63f6c754c34_07_01_01.zip

Результат: статус 204

1.5. Получение списка массовых рассылок

Тип: GET

Путь: /massreport

Параметры запроса:

Имя	Описание	Обязательный
date_start	дата\время начала выборки (Формат: "ГГГГ-ММ-ДД чч:мм:сс"). По умолчанию текущая дата.	–
date_end	дата\время окончания выборки (Формат: "ГГГГ-ММ-ДД чч:мм:сс"). По умолчанию текущая дата.	–
code	код ИФНС (4 цифры)	–

Ответ: JSON-объект, представляющий из себя список объектов с информацией о доступных к получению контейнеров с массовой рассылкой.

Примеры:

- ```
/massreport?date_start=2022-01-01
1[
2 {
3 "fns": "9999",
4 "uid": "632464bdf97e4e1aa7308c88e6e566fd",
5 "filename":
6 "FNS_9999_1AE_d7ab32e90692dece7392c63f6c754c34_04_01_01.zip",
7 "link":
8 "files/gpr/9999/632464bdf97e4e1aa7308c88e6e566fd/FNS_9999_1AE_d7ab32e
9 90692dece7392c63f6c754c34_04_01_01.zip"
10 },
11 {
12 "fns": "0000",
13 "uid": "55b078415bc74b04941f5358f741dc7d",
14 "filename":
15 "FNS_0000_1AE_d7ab32e90692dece7392c63f6c754c34_04_01_01.zip",
```

```
12 "link":
 "files/gpr/0000/632464bdf97e4e1aa7308c88e6e566fd/FNS_000_1AE_d7ab32e9
 0692dece7392c63f6c754c34_04_01_01.zip"
13 }
14]
```

- /massreport?date\_start=2022-01-01 12:56:23GMT&code=123456  
1{  
2 "errors": [  
3 "\"code\" string does not match regex \"^\\d{4}\$\"",  
4 "\"date\_start\" must match pattern \"%Y-%m-%d %H:%M:%S\""  
5 ],  
6 "status": "Bad Request"  
7}

## 2. API обработчика АИС ПФР 2

### 2.1. Получение списка пакетов

**Тип:** GET

**Адрес:**

**Параметры:**

| Имя  | Тип                                 | Описание                      | Обязательность |
|------|-------------------------------------|-------------------------------|----------------|
| type | str ('1c', 'iotchet', 'ao4', 'ao5') | тип абонента                  | +              |
| guid | uuid                                | идентификатор абонента        | -              |
| gate | uuid                                | идентификатор группы абонента | -              |

**Результат:** список словарей с ссылкой на файл и мета-данными

```
1[
2 {
3 "uid": "52fbabca-7a27-446a-9481-ae3d3da027fe",
4 "guid": "6593062e-9030-b2bc-e63a-25fbb4723ecc",
5 "regnum": "099-099-999999",
6 "filename": "szv_td.zip",
7 "processed_at": "2021-07-28 15:05:53.426449",
8 "link": "/099-099-999999/52fbabca-7a27-446a-9481-ae3d3da027fe/szv_td.zip"
9 },
10 {
11 "uid": "52fbabca-7a27-446a-9481-ae3d3da027fe",
12 "guid": "6593062e-9030-b2bc-e63a-25fbb4723ecc",
13 "regnum": "099-099-999999",
14 "filename": "uod.zip",
15 "processed_at": "2021-07-28 15:05:53.426449",
16 "link": "/099-099-999999/52fbabca-7a27-446a-9481-ae3d3da027fe/uod.zip"
17 }
18]
```

### 2.2. Прием пакета от абонента

**Тип:** POST

**Адрес:**

**Заголовки:**

- Content-Type: multipart/form-data

**Тело:**

| Имя  | Тип                                 | Описание                      | Обязательность |
|------|-------------------------------------|-------------------------------|----------------|
| file | file                                | пакет                         | +              |
| type | str ('1c', 'iotchet', 'ao4', 'ao5') | тип абонента                  | +              |
| guid | uuid                                | идентификатор абонента        | +              |
| gate | uuid                                | идентификатор группы абонента | -              |

**Результат:** статус 200 при успешном приеме файла

### 2.3. Скачивание пакета

**Тип:** GET

**Адрес:**

**Описание:**

После успешного получения пакета следует воспользоваться отдельным методом для удаления его из списка и проставления даты.

**Результат:** диалог скачивания пакета

### 2.4. Удаление пакета

**Тип:** DELETE

**Адрес:**

**Результат:** статус 204 при успешном удалении файла из списка или файла уже нет в списке

### 2.5. Описание формата ошибок

Если параметры запроса не удовлетворяют требованиям, вернется ошибка Bad Request со статусом 400. В теле ответа будет JSON, представляющий собой список словарей. Каждый словарь содержит 3 поля:

- "loc" - местоположение элемента, вызвавшего ошибку. Это всегда список с одним или двумя значениями. Первое значение – имя параметра/поля. Второе – порядковый номер элемента если параметром передан список;
- "msg" - сообщение об ошибке;
- "type" - тип ошибки.

**Пример ошибки:**

```
1[
2 {
3 "loc": [
4 "guid"
5],
6 "msg": "field required",
7 "type": "value_error"
8 }
9]
```

В исключительных случаях может вернуться ошибка со статусом 5xx.  
Самая обычная со строковым значением в теле.

### 3. API обработчика Росстат

#### 3.1. Прием пакета от абонента

Тип: POST

Адрес:

Заголовки:

- Content-Type: multipart/form-data

Тело:

| Имя  | Тип                                 | Описание                      | Обязательность |
|------|-------------------------------------|-------------------------------|----------------|
| file | file                                | пакет                         | +              |
| type | str ('1c', 'iotchet', 'ao4', 'ao5') | тип абонента                  | +              |
| guid | uuid                                | идентификатор абонента        | +              |
| gate | uuid                                | идентификатор группы абонента | -              |

Результат: статус 200 при успешном приеме файла

#### 3.2. Получение списка пакетов

Тип: GET

Адрес:

Параметры:

| Имя  | Тип                                 | Описание               | Обязательность |
|------|-------------------------------------|------------------------|----------------|
| type | str ('1c', 'iotchet', 'ao4', 'ao5') | тип абонента           | +              |
| guid | uuid                                | идентификатор абонента | -*             |

|      |      |                               |   |
|------|------|-------------------------------|---|
| gate | uuid | идентификатор группы абонента | – |
|------|------|-------------------------------|---|

*\* Для ао4 обязательно!*

**Результат:** список словарей с ссылкой на файл и мета-данными

```

1[
2 {
3 "uid": "52fbabca-7a27-446a-9481-ae3d3da027fe",
4 "guid": "6593062e-9030-b2bc-e63a-25fbb4723ecc",
5 "окпо": "39650292",
6 "filename": "report_4_3.zip",
7 "link": "/39650292/52fbabca-7a27-446a-9481-ae3d3da027fe/report_4_3.zip"
8 },
9 {
10 "uid": "52fbabca-7a27-446a-9481-ae3d3da027fe",
11 "guid": "6593062e-9030-b2bc-e63a-25fbb4723ecc",
12 "окпо": "39650292",
13 "filename": "report_4_5.zip",
14 "link": "/39650292/52fbabca-7a27-446a-9481-ae3d3da027fe/report_4_5.zip"
15 }
16]

```

### 3.3. Получение списка рассылок

**Тип:** GET

**Адрес:**

**Параметры:**

| Имя        | Тип               | Описание                                 | Обязательность |
|------------|-------------------|------------------------------------------|----------------|
| code       | str               | код подразделения                        | +              |
| date_start | datetime,<br>date | дата начала выборки                      | –*             |
| date_end   | datetime,<br>date | дата окончания выборки<br>(включительно) | –*             |

*\* По умолчанию – текущая дата!*

**Примеры:**

**Результат запроса:** список путей к файлам

```

1[
2 "/99-99/a6e00570-3b1f-4a8d-81b2-f14e7b9508a1/massreport1_3_1.zip",
3 "/99-99/35d1cc65-d5ad-4ff9-821d-093293004a6b/massreport2_3_1.zip",
4]

```

### 3.4. Скачивание пакета

**Тип:** GET

**Адрес:**

**Описание:**

После успешного получения пакета следует воспользоваться отдельным методом для удаления его из списка и проставления даты.

**Примеры:**

**Результат:** диалог скачивания пакета

### 3.5. Удаление пакета

**Тип:** DELETE

**Адрес:**

**Примеры:**

**Результат:** статус 204 при успешном удалении файла из списка или файла уже нет в списке

### 3.6. Описание формата ошибок

Если параметры запроса не удовлетворяют требованиям, вернется ошибка Bad Request со статусом 400. В теле ответа будет JSON, представляющий собой список словарей. Каждый словарь содержит 3 поля:

- "loc" – местоположение элемента, вызвавшего ошибку. Это всегда список с одним или двумя значениями. Первое значение – имя параметра/поля. Второе – порядковый номер элемента если параметром передан список;
- "msg" – сообщение об ошибке;
- "type" – тип ошибки.

**Пример ошибки:**

```
1[
2 {
3 "loc": [
4 "guid"
5],
6 "msg": "field required",
7 "type": "value_error"
8 }
9]
```

В исключительных случаях может вернуться ошибка со статусом 5xx. Самая обычная со строковым значением в теле.