

Antilopaу API: Описание интерфейса для мерчантов

Оглавление

1. История изменений документа	2
2. Термины	3
3. Общие сведения	4
3.1. Описание	4
3.2. Операции	4
3.3. Точка входа API	5
3.4. Подпись запроса	5
3.5. Примеры создания подписи запроса с алгоритмом SHA256WithRSA.	6
4. Службное API.	8
4.1. Проверка подписи	8
5. Платежное API.	9
5.1. Создание платежа	9
5.2. Проверка статуса платежа	12
6. Выплатное API.	15
6.1. Создание выплаты.	15
6.2. Проверка статуса выплаты	17
7. API для возвратов	19
7.1. Создание возврата	19
7.2. Проверка статуса возврата.	21
8. API для проектов	23
8.1. Запрос баланса проекта(-ов)	23
9. Отправка callback сообщений	25
9.1. Проверка подписи callback	25
9.2. Callback для платежа	26
9.3. Callback для выплаты	27
9.4. Callback для возврата средства	28

1. История изменений документа

Дата изменений	Версия документа	Описание изменений
01.12.2023	1.0	Первоначальный документ. Общие сведения. Описание API для платежей, выплат, возвратов средств. Описание callback запросов.
27.12.2023	1.01	Обновление информации в секции "Отправка callback сообщений", проверка подписи callback, примеры callback.
11.01.2024	1.02	Запрос "проверка статуса платежа" и платежный callback возвращают данные покупателя, указанные при создании платежа.
12.01.2024	1.03	Обновление запроса "Создание платежа"
19.01.2024	1.04	Примеры создания подписи для API запросов
29.01.2024	1.05	Новый код ошибки для запросов "Создание платежа" и "Создание выплаты"
29.01.2024	1.06	добавлен тип callback
06.02.2024	1.07	Новый код ошибки для запросов "Создание выплаты"
26.02.2024	1.08	Дополнение описания callback'ов
01.03.2024	1.09	Возможность выбора источника средств для комиссии при выплате (с баланса или с суммы выплаты)
05.03.2024	1.10	Запрос "проверка статуса платежа" и платежный callback возвращают данные оплаты
22.05.2024	1.11	Обновление параметров запроса "Создание возврата" и "Проверка возврата", добавлено описание возможной ошибки при API запросе
06.05.2024	1.12	Данные оплаты выдаются также для неуспешных платежей
24.07.2024	1.13	Эндпоинт для проверки подписи, обновление создания подписи, обновление кодов ошибки для запроса "Создание выплаты"
14.08.2024	1.14	Обновление периодов отправки callback
03.09.2024	1.15	Добавление технической информации о покупателе

2. Термины

Система — Платежная система, позволяющая магазинам принимать электронные платежи (карты, электронные деньги) за товары/услуги и выводить их.

Мерчант, Пользователь — физическое или юридическое лицо, принимающее платежи с помощью системы.

Магазин, проект — сайт, принимающий электронные платежи за предлагаемые товары/услуги. Принадлежит мерчанту. Каждый мерчант может иметь несколько магазинов/проектов.

Поставщик, банк — банк-эквайер, обрабатывающий карточные или платежи электронными деньгами, поступающие в адрес магазинов.

Покупатель — клиент магазина, оплачивающий товар/услуги с помощью системы.

Банк-эмитент — банк выпустивший карту, которой расплачивается покупатель.

Личный кабинет — графический интерфейс мерчанта, в котором он взаимодействует с Системой.

3. Общие сведения

3.1. Описание

С помощью данного API вы сможете взаимодействовать с системой Antilopay: принимать платежи, проводить выплаты и выполнять другие операции, поддерживаемые платформой.

API работает поверх HTTPS протокола и использует JSON-формат для обмена данными.

Для получения доступа к API каждый запрос должен содержать идентификатор мерчанта. API идентификатор мерчанта передаётся в HTTP-заголовке: **X-Apay-Secret-Id**

Все запросы к API должны использовать кодировку UTF-8 и указывать HTTP-заголовок **Content-Type: application/json**. Каждый запрос должен быть подписан, процесс формирования подписи описан ниже в разделе **Подпись запроса**.

Регистр символов параметров важен. Порядок следования параметров в запросах не важен.

Для времени, которое выдаётся в ответах на запросы, используется часовой пояс **UTC+03:00** (Москва).

Параметры запроса, представляющие собою денежные суммы, являются числами с плавающей запятой. Количество цифр после запятой не превышает количество установленное для валюты. Сумма может иметь меньше установленного или не иметь вовсе цифр после запятой.

Валюта	Количество цифр после запятой
RUB	2

Пример для Российского рубля (RUB):

- Сумма имеет 2 цифры после запятой: 100.01 = 100 руб. 1 коп.
- Сумма имеет 1 цифру после запятой: 100.1 = 100 руб. 10 коп.
- Сумма является целым числом: 100 = 100 руб. 00 коп.

3.2. Операции

Платеж

Операция по переводу средств со счёта покупателя на счёт мерчанта. Покупатель оставляет за собой право выбора метода оплаты.

Выплата

Операция по переводу средств с доступного баланса мерчанта на указанный счёт.

Возврат

Операция по возврату потраченных средств на оплату платежа на счёт покупателя.

3.3. Точка входа API

URL для обращения к API: <https://lk.antilopay.com/api/v1/>

3.4. Подпись запроса

Каждый запрос к API должен быть подписан секретным ключом, чтобы система смогла идентифицировать мерчанта и убедиться, что запрос не был изменён после отправки.

Подпись передаётся в HTTP-заголовке: **X-Apay-Sign**

Версия алгоритма подписи передаётся в HTTP-заголовке: **X-Apay-Sign-Version**



Для подписи запросов на выплату используется отдельный секретный ключ!

На данный момент в системе используется только одна версия подписи:

X-Apay-Sign-Version	Алгоритм подписи
1	SHA256WithRSA

Подпись SHA256WithRSA формируется следующим образом:

1. Тело запроса представляется в виде JSON строки, без переносов строк и пробелов между ключом и значением.

Пример:

```
{"project_identificator":"PE8BED46C045139256","amount":100,"order_id":"test_payment_0001","currency":"rub","product_name":"Test coupon","description":"this is test payment"}
```

2. Из строки с помощью секретного ключа генерируется подпись по алгоритму **SHA256WithRSA**.

Секретный ключ представляет собою 512-битный RSA ключ.

Пример ключа:

```
MIIBVAIBADANBgkqhkiG9w0BAQEFAASCAT4wggE6AgEAAkEAiuIjIP1l2as57VabZaj2xfjevMJO3RKJlBqJG69ZirHxOQkq+xCHEQ3aYlcujmhbvfcZP228yZIQZbkyJAoaaQIDAQABAKABm3egMX044/aICMMvDbcqaB84HxsPdLVq23+X8XsZwuH7M1/lvYuG2fQ/dSFB6tc4VvxpLM3P0TLfQ/Da5bkBAiEA5TwxwIe/poNpc416JBkD3c3UpkRB0sC5DDhoz3QNfWkCIQCbGVc/xU/qQBdHQqg6zrEFnn78q1wxCyL0hbJXEbsVAQIhALv0QfrRkyNNUQy2uKn2VMQ9axk0p6Mrt848RjuqtRDZaiA6qu077BD8lN25UNd91y1S6M80GEW5L3M7d08sbEKOAQIgJH032eDzMaJc0/8arwbI6jhBvPg4QvqcVofcdUG1rq0=
```

3. Полученная подпись передаётся в HTTP-заголовок запроса

Пример подписи:

```
H2zlp7GrbMwG6i0lFWWh0eFerqZ+gRlw9l2G8wrp3qzzHJXg+smNX2wg1G8MR+AFdQ2ivasfq5SFtQpu+34yCoA==
```



Тело запроса, которое использовалось для формирования подписи, должно быть идентично тому, которое передается непосредственно в запросе к API, без лишних пробелов, символов, переносов строки.

3.5. Примеры создания подписи запроса с алгоритмом SHA256WithRSA

3.5.1. Пример на Java

```
String key = "<Ваш приватный ключ в строковом представлении>"
String data = "<Нагрузка запроса в текстовом представлении>"

byte[] privateKeyBytes = Base64.getDecoder().decode(key);
KeyFactory pkeyFactory = KeyFactory.getInstance("RSA");
EncodedKeySpec privateKeySpec = new PKCS8EncodedKeySpec(privateKeyBytes);
PrivateKey pkey = pkeyFactory.generatePrivate(privateKeySpec);

Signature privateSignature = Signature.getInstance("SHA256withRSA");
privateSignature.initSign(pkey);
privateSignature.update(data.getBytes(StandardCharsets.UTF_8));
String sign = Base64.getEncoder().encodeToString(privateSignature.sign());
```

Значение переменной `sign` передаете в заголовке `X-Api-Sign`.

3.5.2. Пример на PHP

```
$private_key = "-----BEGIN RSA PRIVATE KEY-----\n<Ваш приватный ключ в строковом\nпредставлении>\n-----END RSA PRIVATE KEY-----";
$data = "<Нагрузка запроса в текстовом представлении>";

$raw_sign = "";
openssl_sign($data, $raw_sign, $private_key, OPENSSL_ALGO_SHA256);
$sign = base64_encode($raw_sign);
```

Значение переменной `sign` передаете в заголовке `X-Api-Sign`.

3.5.3. Пример на NodeJs

```
const crypto = require('crypto');

const privateKey = "-----BEGIN RSA PRIVATE KEY-----\n<Ваш приватный ключ в строковом
```

```
представлении>\n-----END RSA PRIVATE KEY-----";
const data = Buffer.from("<Нагрузка запроса в текстовом представлении>");
const sign = crypto.sign('RSA-SHA256', data, privateKey).toString("base64");
```

Значение переменной `sign` передаете в заголовке `X-Apay-Sign`.

3.5.4. Пример на Python

```
pip install pycryptodome
```

```
import requests
import base64
from Crypto.Hash import SHA256
from Crypto.PublicKey import RSA
from Crypto.Signature import pkcs1_15

str_payload = '<Нагрузка запроса в текстовом представлении>'
secret_key = "<Ваш приватный ключ в формате Base64>"

rsa_key = RSA.importKey(base64.b64decode(secret_key))
payload = bytes(str_payload, 'UTF-8')
hash = SHA256.new(payload)
signature = base64.b64encode(pkcs1_15.new(rsa_key).sign(hash))

# Отправка запроса
url = "https://lk.antilopay.com/api/v1/payment/create"
headers = {"X-Apay-Secret-Id": "<Ваш secret ID>",
           "X-Apay-Sign": signature}
resp = requests.post(url, data=payload, headers=headers)
```

Значение переменной `signature` передаете в заголовке `X-Apay-Sign`.

3.5.5. Пример на Go

```
jsonBody := []byte('<Нагрузка запроса в текстовом представлении>')
bodyReader := bytes.NewReader(jsonBody)

req, err := http.NewRequest(http.MethodPost, url, bodyReader)

key := '<Ваш приватный ключ в формате Base64>'
pk, err := base64.StdEncoding.DecodeString(key)
privateKey, err := x509.ParsePKCS8PrivateKey(pk)
hash := sha256.New()
hash.Write(jsonBody)
```

```
d := hash.Sum(nil)
s, err := rsa.SignPKCS1v15(rand.Reader, privateKey.(*rsa.PrivateKey), crypto.SHA256, d)
sign := base64.StdEncoding.EncodeToString(s)

req.Header.Set("X-Apay-Sign", sign)
```

3.5.6. Пример на C#

```
var strPayload = "<Нагрузка запроса в текстовом представлении>"
var base64Key = "<Ваш приватный ключ в формате Base64>";
byte[] key = Convert.FromBase64String(base64Key);

var rsa = RSA.Create();
rsa.ImportPkcs8PrivateKey(key, out int _);
byte[] signBytes = rsa.SignData(Encoding.UTF8.GetBytes(strPayload), HashAlgorithmName.SHA256,
RSASignaturePadding.Pkcs1);

var sign = Convert.ToBase64String(signBytes);
```

Значение переменной `sign` передаете в заголовке `X-Apay-Sign`.

4. Службное API

4.1. Проверка подписи

URL: <https://lk.antilopay.com/api/v1/signature/check>

Метод запроса: POST

Параметры запроса: без параметров

Пример ответа на успешный запрос:

```
{ "status": "ok" }
```

Пример ответа с ошибкой:

```
{
  "code": 3,
  "error": "Invalid sign"
}
```


5. Платежное API

5.1. Создание платежа

URL: <https://lk.antilopay.com/api/v1/payment/create>

Метод запроса: POST

Параметры запроса:

Название	Тип	Обязательно	Описание
project_identificator	строка	да	Идентификатор проекта мерчанта
amount	число	да	Сумма платежа
order_id	строка	да	Идентификатор платежа на стороне мерчанта. Должен быть уникальным
currency	строка	да	Валюта. Только "RUB"
product_name	строка	да	Название товара/услуги
product_type	строка	да	Принимает значения: "goods" (товары), "services" (услуги)
product_quantity	число	нет	Количество товара/услуги, значение по умолчанию - 1
vat	число	нет	Ставка НДС, возможные значения: 10, 20. Поле обязательное, если сн - ОСНО
description	строка	да	Описание платежа
success_url	строка	нет	URL переадресации покупателя после успешной оплаты
fail_url	строка	нет	URL переадресации покупателя после неуспешной оплаты
customer	объект	да	Данные покупателя

Параметры customer:

Название	Тип	Обязательно	Описание
email	строка	да/нет	Электронная почта покупателя. Если указан phone , то можно пропустить
phone	строка	да/нет	Номер телефона покупателя. Если указан email , то можно пропустить
address	строка	нет	Адрес покупателя
ip	строка	нет	IP адрес покупателя
fullname	строка	нет	ФИО покупателя

Параметры ответа:

Название	Тип	Описание
code	число	Код ответа на запрос: 0 – запрос успешный; иначе – запрос завершился с ошибкой
payment_id	строка	Идентификатор платежа в системе Antilopay
payment_url	строка	Ссылка на форму оплаты платежа
error	строка	Описание ошибки. Отображается, если code не равен 0

Коды ошибок для запроса “Создание платежа”:

Код ошибки	Описание
1	Не предоставлен идентификатор мерчанта (X-Apay-Secret-Id)
2	Неправильный идентификатор мерчанта
3	Неправильная подпись запроса
4	Проект не найден
5	Предоставленный идентификатор платежа от мерчанта уже использован
6	Проект не подтверждён
7	Нет доступных платежных шлюзов, чтобы произвести платеж
9	Мерчант не имеет разрешения на выполнение этого запроса
10	Не предоставлены данные покупателя
11	Данные покупателя должны содержать phone или email
12	Данные покупателя содержат ошибку
13	Неправильная сумма платежа
14	success_url или fail_url является некорректным HTTP-адресом
15	Ошибка в параметрах запроса
16	Ошибка в составлении JSON
17	Не предоставлена ставка НДС или недопустимое значение
18	Предоставлен недопустимый тип объекта платежа
19	Предоставлено неверное количество товара/услуг
20	Сумма должна быть больше чем указано в сообщении error
21	Проекту не назначены шлюзы
22	Данный IP адрес не имеет разрешения на обращение к API
1000	Неизвестная ошибка, обратитесь в тех. поддержку

Пример ответа на успешный запрос:

```
{
  "code": 0,
  "payment_id": "APAY4AA6BB4B1701155257296",
  "payment_url": "https://gate.antilopay.com/#payment/APAY4AA6BB4B1701155257296"
}
```

Пример ответа с ошибкой:

```
{
  "code": 3,
  "error": "Invalid sign"
}
```

5.2. Проверка статуса платежа

URL: <https://lk.antilopay.com/api/v1/payment/check>

Метод запроса: POST

Параметры запроса:

Название	Тип	Обязательно	Описание
project_identificator	строка	да	Идентификатор проекта мерчанта
order_id	строка	да	Идентификатор платежа со стороны мерчанта

Параметры ответа:

Название	Тип	Описание
code	число	Код ответа на запрос: 0 – запрос успешный; иначе – запрос завершился с ошибкой
payment_id	строка	Идентификатор платежа в системе Antilopay
order_id	строка	Идентификатор платежа со стороны мерчанта
ctime	строка	Время создания платежа
amount	число	Сумма платежа, которая зачислится на баланс
original_amount	число	Сумма платежа, указанная при создании
fee	число	Сумма комиссии
status	строка	Статус платежа
currency	строка	Валюта платежа
product_name	строка	Наименование продукта/услуги
description	строка	Описание платежа
pay_method	строка	Способ оплаты. Отображается, если оплата прошла успешно или завершилась с ошибкой
pay_data	строка	Данные оплаты. Отображается, если оплата прошла успешно или завершилась с ошибкой
customer_ip	строка	IP адрес покупателя. Отображается, если оплата прошла успешно или завершилась с ошибкой
customer_useragent	строка	UserAgent покупателя. Отображается, если оплата прошла успешно или завершилась с ошибкой
customer	объект	Данные покупателя, указанные при создании платежа
error	строка	Описание ошибки. Отображается, если code не равен 0

Способы оплаты:

Название	Описание	Данные оплаты
CARD_RU	Банковская карта РФ банка	Номер банковской карты (pan)
QIWI	Кошелек QIWI	Номер телефона
SBER_PAY	Оплата через SberPay	Номер телефона

Параметры customer:

Название	Тип	Описание
email	строка	Электронная почта покупателя. Если указан phone , то можно пропустить
phone	строка	Номер телефона покупателя. Если указан email , то можно пропустить
address	строка	Адрес покупателя
ip	строка	IP адрес покупателя
fullname	строка	ФИО покупателя

Статусы платежа:

Статус	Описание
PENDING	Ожидание оплаты
SUCCESS	Платеж успешно оплачен
FAIL	Платеж не был оплачен из-за ошибки
CANCEL	Платеж был отменен покупателем
EXPIRED	Не было попыток произвести оплату платежа, платеж более нельзя оплатить

Коды ошибок для запроса “Проверка статуса платежа”:

Код ошибки	Описание
1	Не предоставлен идентификатор мерчанта (X-Apay-Secret-Id)
2	Неправильный идентификатор мерчанта
3	Неправильная подпись запроса
4	Проект не найден
8	Платеж с указанным идентификатором не найден
15	Ошибка в параметрах запроса
16	Ошибка в составлении JSON
22	Данный IP адрес не имеет разрешения на обращение к API
1000	Неизвестная ошибка, обратитесь в тех. поддержку

Пример ответа на успешный запрос:

```
{
  "code": 0,
  "payment_id": "APAYE98274D61702540483325",
  "order_id": "MERCHANT_ORDER_ID_0001",
  "ctime": "2023-11-29 14:30:00.012345",
  "amount": 990.00,
  "original_amount": 1000.00,
  "fee": 10.00,
  "status": "SUCCESS",
  "currency": "RUB",
  "product_name": "your product name",
  "description": "your description",
  "pay_method": "CARD_RU",
  "pay_data": "411111*****1111",
  "customer": {
    "email": "test@mail.com",
    "phone": "+7 9008007000",
    "fullname": null,
    "ip": null,
    "address": null
  }
}
```

Пример ответа с ошибкой:

```
{
  "code": 8,
  "error": "Payment not found"
}
```

6. Выплатное API

6.1. Создание выплаты

URL: <https://lk.antilopay.com/api/v1/withdraw/create>

Метод запроса: POST

Параметры запроса:

Название	Тип	Обязательно	Описание
project_identificator	строка	да	Идентификатор проекта мерчанта
amount	число	да	Сумма выплаты
method	строка	да	Способ выплаты
account	строка	да	Адрес вывода средств. Зависит от указанного method
fee_type	строка	нет	Источник средств для выплаты комиссии: 'BALANCE' - баланс проекта, 'ORDER' - выплата. По умолчанию 'ORDER'

Поддерживаемые методы:

Название	Описание	Значение account
CARD_RU	Банковская карта РФ банка	Номер банковской карты (pan)
QIWI	Кошелек QIWI	Номер телефона
CRYPTO_TRON_USDT	Перевод USDT на кошелек в блокчейне TRON	Адрес кошелька



Доступный перечень методов выплаты для проекта можете уточнить у менеджера

Параметры ответа:

Название	Тип	Описание
code	число	Код ответа на запрос: 0 – запрос успешный; иначе – запрос завершился с ошибкой
withdraw_id	строка	Идентификатор выплаты в системе Antilopay
error	строка	Описание ошибки. Отображается, если code не равен 0

Коды ошибок для запроса “Создание выплаты”:

Код ошибки	Описание
1	Не предоставлен идентификатор мерчанта (X-Apay-Secret-Id)

Код ошибки	Описание
2	Неправильный идентификатор мерчанта
3	Неправильная подпись запроса
4	Проект не найден
5	Отсутствует доступный шлюз
6	Проект не подтверждён
8	Недостаточно средств на балансе
9	Мерчант не имеет разрешения на выполнение этого запроса
10	Некорректная сумма выплаты
11	Ошибка в параметрах запроса
12	Ошибка в составлении JSON
13	Проекту не назначены шлюзы
14	Данная сумма не соответствует выплатам лимитам
15	Неправильное значение источника средств для выплаты комиссии
17	Данный IP адрес не имеет разрешения на обращение к API
1000	Неизвестная ошибка, обратитесь в тех. поддержку

Пример ответа на успешный запрос:

```
{
  "code": 0,
  "withdraw_id": "WDAPAY3184A312C99012138990011"
}
```

Пример ответа с ошибкой:

```
{
  "code": 8,
  "error": "Balance issue"
}
```


6.2. Проверка статуса выплаты

URL: <https://lk.antilopay.com/api/v1/withdraw/check>

Метод запроса: POST

Параметры запроса:

Название	Тип	Обязательно	Описание
project_identificator	строка	да	Идентификатор проекта мерчанта
withdraw_id	строка	да	Идентификатор выплаты

Параметры ответа:

Название	Тип	Описание
code	число	Код ответа на запрос: 0 – запрос успешный; иначе – запрос завершился с ошибкой
withdraw_id	строка	Идентификатор выплаты
ctime	строка	Время создания выплаты
status	строка	Статус выплаты
amount	число	Сумма выплаты
fee	число	Комиссия выплаты
fee_type	строка	Источник средств для выплаты комиссии: 'BALANCE' - баланс проекта, 'ORDER' - выплата
currency	строка	Валюта выплаты
provide_method	строка	Метод выплаты
error	строка	Описание ошибки. Отображается, если code не равен 0

Статусы выплаты:

Статус	Описание
PENDING	Ожидание проведения выплаты
WAIT_ANSWER	Выплата в процессе
FAIL	Выплата не была произведена по причине ошибки
COMPLETE	Выплата успешно завершена

Коды ошибок для запроса “Проверка статуса выплаты”:

Код ошибки	Описание
1	Не предоставлен идентификатор мерчанта (X-Apay-Secret-Id)

Код ошибки	Описание
2	Неправильный идентификатор мерчанта
3	Неправильная подпись запроса
7	Выплата не найдена
11	Ошибка в параметрах запроса
12	Ошибка в составлении JSON
17	Данный IP адрес не имеет разрешения на обращение к API
1000	Неизвестная ошибка, обратитесь в тех. поддержку

Пример ответа на успешный запрос:

```
{
  "code": 0,
  "withdraw_id": "WDAPAY3184A312C99012138990011",
  "ctime": "2023-11-30 15:30:00.012345",
  "status": "WAIT_ANSWER",
  "amount": 1000.00,
  "fee": 10.00,
  "fee_type": "ORDER",
  "currency": "RUB",
  "provide_method": "CARD_RU",
}
```

Пример ответа с ошибкой:

```
{
  "code": 7,
  "error": "Withdraw not found"
}
```

7. API для возвратов

7.1. Создание возврата

URL: <https://lk.antilopay.com/api/v1/refund/create>

Метод запроса: POST

Параметры запроса:

Название	Тип	Обязательно	Описание
project_identificator	строка	да	Идентификатор проекта мерчанта
transaction_id	строка	да	Идентификатор платежа в системе Antilopay
amount	число	да	Сумма возврата

Параметры ответа:

Название	Тип	Описание
code	число	Код ответа на запрос: 0 – запрос успешный; иначе – запрос завершился с ошибкой
refund_id	строка	Идентификатор возврата в системе Antilopay
error	строка	Описание ошибки. Отображается, если code не равен 0

Коды ошибок для запроса “Создание возврата”:

Код ошибки	Описание
1	Не предоставлен идентификатор мерчанта (X-Apay-Secret-Id)
2	Неправильный идентификатор мерчанта
3	Неправильная подпись запроса
8	Платеж для возврата средств не найден
9	Для этого платежа нельзя создать возврат средств
11	Несоответствующий статус мерчанта или нет разрешения на выполнение этой операции
12	Некорректная сумма возврата
13	Ошибка в параметрах запроса
14	Ошибка в составлении JSON
15	Данный IP адрес не имеет разрешения на обращение к API
1000	Неизвестная ошибка, обратитесь в тех. поддержку

Пример ответа на успешный запрос:

```
{
  "code": 0,
  "refund_id": "RDAPAY4AA6BB4B1781155257296"
}
```

Пример запроса с ошибкой:

```
{
  "code": 3,
  "error": "Invalid sign"
}=== Отмена платежа
```

URL: <https://lk.antilopay.com/api/v1/reverse/create>

Метод запроса: POST

Параметры запроса:

Название	Тип	Обязательно	Описание
project_identificator	строка	да	Идентификатор проекта мерчанта
transaction_id	строка	да	Идентификатор платежа в системе Antilopay

Параметры ответа:

Название	Тип	Описание
code	число	Код ответа на запрос: 0 – запрос успешный; иначе – запрос завершился с ошибкой
refund_id	строка	Идентификатор возврата в системе Antilopay
error	строка	Описание ошибки. Отображается, если code не равен 0

Коды ошибок для запроса “Создание возврата”:

Код ошибки	Описание
1	Не предоставлен идентификатор мерчанта (X-Apay-Secret-Id)
2	Неправильный идентификатор мерчанта
3	Неправильная подпись запроса
8	Платеж для возврата средств не найден
9	Для этого платежа нельзя создать возврат средств
11	Несоответствующий статус мерчанта или нет разрешения на выполнение этой операции
12	Некорректная сумма возврата

Код ошибки	Описание
13	Ошибка в параметрах запроса
14	Ошибка в составлении JSON
15	Данный IP адрес не имеет разрешения на обращение к API
16	Этот платеж нельзя отменить
1000	Неизвестная ошибка, обратитесь в тех. поддержку

Пример ответа на успешный запрос:

```
{
  "code": 0,
  "refund_id": "RDAPAY4AA6BB4B1701155257296"
}
```

Пример запроса с ошибкой:

```
{
  "code": 3,
  "error": "Invalid sign"
}
```

7.2. Проверка статуса возврата

URL: <https://lk.antilopay.com/api/v1/refund/check>

Метод запроса: POST

Параметры запроса:

Название	Тип	Обязательно	Описание
project_identificator	строка	да	Идентификатор проекта мерчанта
refund_id	строка	да	Идентификатор возврата в системе Antilopay

Параметры ответа:

Название	Тип	Описание
code	число	Код ответа на запрос: 0 – запрос успешный; иначе – запрос завершился с ошибкой
refund_id	строка	Идентификатор возврата в системе Antilopay
status	строка	Статус возврата
amount	число	Сумма возврата

Название	Тип	Описание
error	строка	Описание ошибки. Отображается, если code не равен 0

Статусы возврата:

Статус	Описание
PENDING	Ожидание проведения возврата средств
WAIT_ANSWER	Возврат средств в процессе
FAIL	В процессе возврата средств произошла ошибка
COMPLETE	Возврат средств был проведён успешно

Коды ошибок для запроса “Проверка статуса возврата”:

Код ошибки	Описание
1	Не предоставлен идентификатор мерчанта (X-Apay-Secret-Id)
2	Неправильный идентификатор мерчанта
3	Неправильная подпись запроса
10	Возврат средств не найден
13	Ошибка в параметрах запроса
14	Ошибка в составлении JSON
15	Данный IP адрес не имеет разрешения на обращение к API
1000	Неизвестная ошибка, обратитесь в тех. поддержку

Пример ответа на успешный запрос:

```

{
  "code": 0,
  "refund_id": "RDAPAY4AA6BB4B1701155257296",
  "amount": 100.00,
  "status": "COMPLETE"
}

```

Пример запроса с ошибкой:

```

{
  "code": 10,
  "error": "Refund not found"
}

```

8. API для проектов

8.1. Запрос баланса проекта(-ов)

URL: <https://lk.antilopay.com/api/v1/project/balance>

Метод запроса: POST

Параметры запроса:

Название	Тип	Обязательно	Описание
project_identificator	строка	нет	Идентификатор проекта мерчанта

Параметры ответа в случае указания проекта:

Название	Тип	Описание
code	число	Код ответа на запрос: 0 – запрос успешный; иначе – запрос завершился с ошибкой
project_identificator	строка	Идентификатор проекта мерчанта
available	число	Доступный баланс
blocked	число	Заблокированный баланс
withdraw	число	Баланс, ожидающий вывода

Параметры ответа без указания проекта:

Название	Тип	Описание
code	число	Код ответа на запрос: 0 – запрос успешный; иначе – запрос завершился с ошибкой
balances	массив объектов	Массив балансов проектов

Содержимое массива balances

Название	Тип	Описание
project_identificator	строка	Идентификатор проекта мерчанта
available	число	Доступный баланс
blocked	число	Заблокированный баланс
withdraw	число	Баланс, ожидающий вывода

Коды ошибок для запроса “Запрос баланса проекта”:

Код ошибки	Описание
1	Не предоставлен идентификатор мерчанта (X-Api-Secret-Id)
2	Неправильный идентификатор мерчанта
4	Проект не найден
6	Проект не подтверждён
9	Мерчант не имеет разрешения на выполнение этого запроса
10	Нету подтвержденных проектов

Пример ответа на успешный запрос с указанием проекта:

```
{
  "available": 10.32,
  "blocked": 2123.47,
  "withdraw": 111.11,
  "project_identificator": "P00000002",
  "code": 0
}
```

Пример ответа на успешный запрос:

```
{
  "code": 0,
  "balances": [
    {
      "available": 10.32,
      "blocked": 2123.47,
      "withdraw": 111.11,
      "project_identificator": "P00000002"
    },
    {
      "available": 100.00,
      "blocked": 0,
      "withdraw": 0,
      "project_identificator": "P00000006"
    }
  ]
}
```

Пример запроса с ошибкой:

```
{
  "code": 10,
  "error": "There are no projects"
}
```



```
}
```

9. Отправка callback сообщений

На сервер мерчанта отправляются сообщения, информирующие об изменении статуса операции, которая была создана с помощью API. Учитывайте что будут отправляться сообщения и об неуспешных операциях. При изменении статуса операции сервер Antilopay отправляет **POST** запрос в формате **application/json** с кодировкой **UTF-8** на указанный в настройках проекта адрес для приема callback сообщений.

Сервер системы ждет ответ на запрос **HTTP 200 OK**, если сервер мерчанта получил данные.

Если в процессе обработки данных со стороны сервера мерчанта возникла ошибка, сервер всё равно должен ответить **HTTP 200 OK**.

В случае если сервер мерчанта не ответил на callback запрос ответом **HTTP 200 OK**, будет производиться повторная отправка запросов.

Сервер системы будет предпринимать повторные попытки отправить callback в течение 24 часов каждые 5 минут с момента первой отправки callback запроса.

Статус операции можно будет запросить с помощью API запроса.

В зависимости от операции содержимое запроса будет отличаться.



Отсутствие callback по операции не означает, что операция не выполнялась или статус остался в определенном значении. Рекомендуется проверить статус операции через API или обратиться в поддержку.

9.1. Проверка подписи callback

Callback запрос содержит заголовок **X-Apay-Callback** с подписью и заголовок **X-Apay-Callback-Version**, который определяет алгоритм подписи. Ключи для проверки подписи генерируются и отправляются мерчанту индивидуально для каждого его подтвержденного проекта.

Значение X-Apay-Callback-Version	Алгоритм подписи
1	SHA256WithRSA

Текущая используемая версия: 1

В настройках проекта можно установить версию callback.

Подпись вычисляется на основе полного тела запроса callback, то есть берется весь запрос:

```
{"payment_id":"APAY269B723F1703648835542","order_id":"test_payment_133","ctime":"2023-12-27 10:47:15.550961","amount":77.51,"original_amount":77.51,"fee":0,"status":"PENDING","currency"
```

```
:"rub","product_name":"API Access","description":"X-Apay-Access"}
```

И генерируется подпись

```
BSFww9t2f8PyYjseg2Iili/KzC7WroQTtZMRGIHeKMPb1kaTaXig1/pytoiW43KpcDlMoi1lmuFthT5yhM2JuQ==
```

9.2. Callback для платежа

Возможные статусы: SUCCESS, FAIL

Параметры запроса:

Название	Тип	Описание
type	строка	Значение всегда равно payment для платежей
payment_id	строка	Идентификатор платежа в системе Antilopay
order_id	строка	Идентификатор платежа со стороны мерчанта
ctime	строка	Время создания платежа
amount	число	Сумма платежа, которая зачислится на баланс
original_amount	число	Сумма платежа, указанная при создании
fee	число	Сумма комиссии
status	строка	Статус платежа
currency	строка	Валюта платежа
product_name	строка	Наименование продукта/услуги
description	строка	Описание платежа
pay_method	строка	Способ оплаты. Отображается, если оплата прошла успешно или завершилась с ошибкой
pay_data	строка	Данные оплаты. Отображается, если оплата прошла успешно или завершилась с ошибкой
customer_ip	строка	IP адрес покупателя. Отображается, если оплата прошла успешно или завершилась с ошибкой
customer_useragent	строка	UserAgent покупателя. Отображается, если оплата прошла успешно или завершилась с ошибкой
customer	объект	Данные покупателя, указанные при создании платежа

Параметры customer:

Название	Тип	Описание
email	строка	Электронная почта покупателя. Если указан phone , то можно пропустить

Название	Тип	Описание
phone	строка	Номер телефона покупателя. Если указан email , то можно пропустить
address	строка	Адрес покупателя
ip	строка	IP адрес покупателя
fullname	строка	ФИО покупателя

Пример callback платежа

```
{
  "type": "payment",
  "payment_id": "APAYE98274D61782548483325",
  "order_id": "MERCHANT_ORDER_ID_0001",
  "ctime": "2023-11-29 14:30:00.012345",
  "amount": 1000.00,
  "original_amount": 1000.00,
  "fee": 0.00,
  "status": "PENDING",
  "currency": "RUB",
  "product_name": "your product name",
  "description": "your description",
  "pay_method": "CARD_RU",
  "pay_data": "411111*****1111",
  "customer": {
    "email": "test@mail.com",
    "phone": "+79587888885",
    "ip": "96.10.250.1",
    "address": "Moscow, st. Pushkina, 1",
    "fullname": "John Doe"
  }
}
```

9.3. Callback для выплаты

Возможные статусы: COMPLETE, FAIL

Параметры запроса:

Название	Тип	Описание
type	строка	Значение всегда равно withdraw для выплат
withdraw_id	строка	Идентификатор выплаты
ctime	строка	Время создания выплаты
status	строка	Статус выплаты

Название	Тип	Описание
amount	строка	Сумма выплаты
fee	число	Сумма комиссии
fee_type	строка	С чего списывается комиссия
currency	строка	Валюта выплаты
provide_method	строка	Метод выплаты

Пример callback выплаты

```
{
  "type": "withdraw",
  "withdraw_id": "WDAPAY3184A312C99012138990011",
  "ctime": "2023-11-30 15:30:00.012345",
  "status": "COMPLETE",
  "amount": 1000.00,
  "fee": 10.00,
  "fee_type": "ORDER",
  "currency": "RUB",
  "provide_method": "CARD_RU"
}
```

9.4. Callback для возврата средства

Возможные статусы: COMPLETE, FAIL

Параметры запроса:

Название	Тип	Описание
type	строка	Значение всегда равно refund для возвратов
refund_id	строка	Идентификатор возврата в системе Antilopay
status	строка	Статус возврата
ctime	строка	Время создания возврата
amount	число	Сумма возврата

Пример callback возврата

```
{
  "type": "refund",
  "refund_id": "RDAPAY17327AF61702552484490",
  "amount": 1000.00,
  "ctime": "2023-11-29 14:40:00.012345",
}
```

```
"status": "COMPLETE"
```

```
}
```