

## □ Documentation

Web Service

# App Store Server API

Manage your customers' App Store transactions from your server.

App Store Server API 1.0+

## Overview

The App Store Server API is a REST API that you call from your server to request and provide information about your customers' In-App Purchases. The App Store signs the transaction and subscription renewal information that this API returns using the [JSON Web Signature \(JWS\)](#) specification. Most endpoints return data for a single customer of your app, indicated by a transaction identifier that you provide.

The App Store Server API is independent of the app's installation status on the customers' devices. The App Store server returns information based on a customer's In-App Purchase history regardless of whether the customer installs, removes, or reinstalls the app on their devices.

This API provides the following functionality:

- **Transactions and auto-renewable subscription status.** Get information for single transactions by calling [Get Transaction Info](#) or a customer's entire transaction history using [Get Transaction History](#). Call [Get All Subscription Statuses](#) for up-to-date subscription status. Use this information to keep your customers' purchase information current on your server.
- **Refund information.** Call [Get Refund History](#) to get a customer's refund history. Use the [Send Consumption Information](#) endpoint to send information to the App Store when customers request a refund for a consumable In-App Purchase, after you receive the [CONSUMPTION\\_REQUEST notificationType](#) from [App Store Server Notifications V2](#). Your data helps inform refund decisions.
- **App Store Server Notifications history and testing.** Call [Get Notification History](#) to request the notifications your server may have missed in the past 180 days (or 30 days in the sandbox environment). Call [Request a Test Notification](#) and [Get Test](#)

[Notification Status](#) to test if your server is successfully receiving notifications at its [App Store Server Notifications V2](#) endpoint.

- **Subscription renewal date extensions.** Call [Extend a Subscription Renewal Date](#) and related endpoints to compensate your customers for temporary service outages, canceled events, or interruptions to live-streamed events by extending the renewal date of their paid, active subscription. For more information, see [Extending the renewal date for auto-renewable subscriptions](#).
- **Order information lookup.** Call [Look Up Order ID](#) to get In-App Purchase information based on a customer's order ID, found on the App Store receipt that customers receive in email.
- **App transaction information and setting an app account token.** Call [Get App Transaction Info](#) to get details about the customer's purchase of your app, such as the original purchase date and version. Use [Set App Account Token](#) to set an app account token when your customer makes an In-App Purchase outside your app, or to update its value.

Your server must support the Transport Layer Security (TLS) protocol 1.2 or later to use the App Store Server API.

Check the [App Store Server API changelog](#) to learn about the latest changes to this API. Look for videos about the App Store Server API on the [Apple Developer website](#).

## Authorize your API calls

Calls to the API require JSON Web Tokens (JWTs) for authorization; you obtain keys to create the tokens from your organization's App Store Connect account. See [Creating API keys to authorize API requests](#) to create your keys. See [Generating JSON Web Tokens for API requests](#) to generate tokens using your keys, and send API requests.

After you have a complete and signed token, provide the token in the request's authorization header as a bearer token. Generate a new token for each new API request, or reuse tokens until they expire.

## Create JWTs, verify transactions, and more using the App Store Server Library

The App Store Server Library is an open source library from Apple, available in four languages. It provides a client that make it easier to adopt the App Store Server APIs, including creating the JWTs to authorize calls. For more information, see [Simplifying your implementation by using the App Store Server Library](#) and the WWDC23 session [Meet the App Store Server Library](#).

## Test using the sandbox environment

All App Store Server API endpoints are available for testing in the sandbox environment, except [Look Up Order ID](#). Access the sandbox environment by sending requests to the endpoints using the following base URL:

`https://api.storekit-sandbox.itunes.apple.com/`

For example, to call [Get Transaction History](#) in the sandbox environment, send a request using the sandbox URL:

`https://api.storekit-sandbox.itunes.apple.com/inApps/v2/history/{transactionId}`

Note that /inApps in the path is case-sensitive.

For endpoints that take a [transactionId](#) as a parameter, be sure to call the endpoint using the same environment that creates the transaction identifier. Environment information is present in the [environment](#) property of the [JWSTransactionDecodedPayload](#).

If you don't have environment information, follow these steps:

1. Call the endpoint using the production URL. If the call succeeds, the transaction identifier belongs to the production environment.
2. If you receive an error code [4040010 TransactionIdNotFoundError](#), call the endpoint using the sandbox environment.
3. If the call succeeds, the transaction identifier belongs to the sandbox environment. If the call fails with the [4040010](#) error code, the transaction identifier isn't present in either environment.

## Topics

### Essentials

-  [Simplifying your implementation by using the App Store Server Library](#)  
Use Apple's open source library to create JSON Web Tokens (JWT) to authorize your calls, verify transactions, extract transaction identifiers from receipts, and more.
-  [Creating API keys to authorize API requests](#)  
Create API keys you use to sign JSON Web Tokens and authorize API requests.
-  [Generating JSON Web Tokens for API requests](#)  
Create JSON Web Tokens signed with your private key to authorize requests for App Store Server API and External Purchase Server API.

## Identifying rate limits

Recognize the rate limits that apply to App Store Server API endpoints and handle them in your code.

## App Store Server API changelog

Learn about new features and updates in the App Store Server API.

# In-App Purchase history

## Get Transaction History

Get a customer's in-app purchase transaction history for your app.

### object HistoryResponse

A response that contains the customer's transaction history for an app.

# Transaction information

## Get Transaction Info

Get information about a single transaction for your app.

### object TransactionInfoResponse

A response that contains signed transaction information for a single transaction.

# App Transaction information

## Get App Transaction Info

Get a customer's app transaction information for your app.

### object AppTransactionInfoResponse

A response that contains signed app transaction information for a customer.

# Subscription status

## Get All Subscription Statuses

Get the statuses for all of a customer's auto-renewable subscriptions in your app.

### object StatusResponse

A response that contains status information for all of a customer's auto-renewable subscriptions in your app.

## App Account Token

### Set App Account Token

Sets the app account token value for a purchase the customer makes outside of your app, or updates its value in an existing transaction.

#### object UpdateAppAccountTokenRequest

The request body that contains an app account token value.

## Order ID lookup

### Look Up Order ID

Get a customer's in-app purchases from a receipt using the order ID.

#### type orderId

The customer's order ID from an App Store receipt for in-app purchases.

#### object OrderLookupResponse

A response that includes the order lookup status and an array of signed transactions for the in-app purchases in the order.

## Consumption information

### Send Consumption Information

Send consumption information about a consumable in-app purchase or auto-renewable subscription to the App Store after your server receives a consumption request notification.

#### object ConsumptionRequest

The request body containing consumption information.

## Refund lookup

### Get Refund History

Get a paginated list of all of a customer's refunded in-app purchases for your app.

#### object RefundHistoryResponse

A response that contains an array of signed JSON Web Signature (JWS) refunded transactions, and paging information.

## Subscription-renewal-date extension

Extending the renewal date for auto-renewable subscriptions

Compensate eligible active subscribers for service interruptions by extending a subscription's renewal date.

### Extend a Subscription Renewal Date

Extends the renewal date of a customer's active subscription using the original transaction identifier.

### Extend Subscription Renewal Dates for All Active Subscribers

Uses a subscription's product identifier to extend the renewal date for all of its eligible active subscribers.

### Get Status of Subscription Renewal Date Extensions

Checks whether a renewal date extension request completed, and provides the final count of successful or failed extensions.

#### object ExtendRenewalDateRequest

The request body that contains subscription-renewal-extension data for an individual subscription.

#### object ExtendRenewalDateResponse

A response that indicates whether an individual renewal-date extension succeeded, and related details.

#### object MassExtendRenewalDateRequest

The request body that contains subscription-renewal-extension data to apply for all eligible active subscribers.

#### object MassExtendRenewalDateResponse

A response that indicates the server successfully received the subscription-renewal-date extension request.

#### object MassExtendRenewalDateStatusResponse

A response that indicates the current status of a request to extend the subscription renewal date to all eligible subscribers.

## App Store Server Notifications history

## Get Notification History

Get a list of notifications that the App Store server attempted to send to your server.

**object NotificationHistoryRequest**

The request body for notification history.

**object NotificationHistoryResponse**

A response that contains the App Store Server Notifications history for your app.

**object notificationHistoryResponseItem**

The App Store server notification history record, including the signed notification payload and the result of the server's first send attempt.

## App Store Server Notifications testing

### Request a Test Notification

Ask App Store Server Notifications to send a test notification to your server.

### Get Test Notification Status

Check the status of the test App Store server notification sent to your server.

**object SendTestNotificationResponse**

A response that contains the test notification token.

**object CheckTestNotificationResponse**

A response that contains the contents of the App Store server's test notification and the result from your server.

## JWS headers and payloads

**object JWSDecodedHeader**

A decoded JSON Web Signature (JWS) header containing transaction or renewal information.

**type JWSAppTransaction**

App transaction information signed by the App Store, in JSON Web Signature (JWS) Compact Serialization format.

**object JWSAppTransactionDecodedPayload**

A decoded payload that contains app transaction information.

**type JWSTransaction**

Transaction information signed by the App Store, in JSON Web Signature (JWS) Compact Serialization format.

## object JWSTransactionDecodedPayload

A decoded payload that contains transaction information.

## type JWSRenewalInfo

Subscription renewal information, signed by the App Store, in JSON Web Signature (JWS) format.

## object JWSRenewalInfoDecodedPayload

A decoded payload containing subscription renewal information for an auto-renewable subscription.

### ☰ Data types

Refer to these data types for decoded transaction and renewal information payloads.

## Error information

### ☰ Error codes

Understand the error codes that App Store Server API responses return.

## Deprecated

### ~~Get Transaction History V1~~

Get a customer's in-app purchase transaction history for your app, except finished consumable in-app purchases.

Deprecated

### ~~Get Refund History V1~~

Get a list of up to 50 of a customer's refunded in-app purchases for your app.

Deprecated

### ~~object RefundLookupResponse~~

A response that contains an array of signed JSON Web Signature (JWS) transactions.

Deprecated