

Creating and Managing Webhooks via OnceHub [New]

Webhooks enable real-time data synchronization from OnceHub to your external applications. When an event occurs in OnceHub such as a new booking, a data payload is automatically sent to your designated URL. This allows you to automate workflows by receiving instant updates, eliminating the need for your system to constantly check OnceHub for new data.

While webhooks can be managed via the API, you can also configure them directly from your account interface.

How to Create a Webhook

To create a webhook:

1. Navigate to the **gear icon** in the top-right corner of the page.
2. Select **Account Integrations** from the dropdown menu.
3. Select **APIs & Webhooks** tile.
4. In the **Webhooks** section, click the **Create Webhook** button.
5. In the pop-up, provide the following details:
 - **Webhook Name:** A unique, descriptive name for your integration.
 - **Webhook URL:** The destination address (starting with https://) where the data will be sent.
6. Click **Next**.
7. **Select the checkboxes** for the specific events that should trigger the webhook.
8. Click **Create**.
9. A **Webhook Secret** will be displayed. Click **Copy & close** to save this secret immediately. It is required to verify that the received data is authentic.

The webhook is now active and will appear in your **Webhooks** list.

Understanding Supported Webhook Event Triggers

When a webhook is triggered, OnceHub sends a data payload containing either a **booking** or **conversation object**. These objects are generated by specific user actions.

The Booking Object

The **booking object** represents a scheduled meeting. It is produced when:

- A guest schedules a meeting through a **Booking Calendar** or **Booking Hub**.
- A meeting is scheduled via a **Chatbot** or a **Routing Form** interaction.

The Conversation Object

The **conversation object** represents a digital interaction session. It is produced when:

- A visitor interacts with an automated **Chatbot** or a **Routing Form**.
- A **live engagement** session is initiated with an agent.

Event Trigger Descriptions

The following table describes the specific **Webhook event** triggers available in the OnceHub interface:

Event Trigger	Description
Booking Scheduled Event	Triggers when a customer schedules a new booking.
Booking Rescheduled Event	Triggers when an existing booking is moved to a new time.
Booking Reassigned Event	Triggers when a booking is moved from one user to another.
Booking Canceled Then Rescheduled Event	Triggers when a booking is canceled and immediately rescheduled.
Booking Canceled Reschedule Requested Event	Triggers when a booking is canceled with a request for the customer to reschedule.
Booking Canceled Event	Triggers when a booking is officially canceled.
Booking Completed Event	Triggers once the scheduled booking end time has passed.
Booking No Show Event	Triggers when a user manually marks a completed booking as a No-show.
Conversation Started Event	Triggers when a new chatbot or live engagement session begins.
Conversation Closed Event	Triggers when a conversation is successfully concluded. This occurs when: <ul style="list-style-type: none"> • The visitor reaches the end of a conversation flow. • The visitor starts a new conversation with a different chatbot.
Conversation Abandoned Event	Triggers when a customer leaves a conversation before it is completed or closed.

How to Manage Existing Webhooks

You can manage active webhook subscriptions directly from the **APIs & Webhooks** page using the following actions:

- **View Secret:** Click to retrieve the signing secret. Your system uses this secret to **verify that incoming data is authentic** and was sent by OnceHub, preventing unauthorized messages.
 - **Delete:** If a webhook is no longer needed, click **Delete** to permanently remove the webhook subscription.
-

Troubleshooting Common Issues

If you are unable to complete the setup, verify the following:

- **URL Format:** Ensure the Webhook URL starts with **https://**
 - **Alphanumeric Name:** The Webhook name must contain at least one alphanumeric character.
-