Chatbase

API

Webhooks

Webhook API Guide

The Webhook API guide allows you to set-up webhooks to receive a `POST` request on when an event or more is triggered.

Payload

Key	Type	Description
eventType	string	**Event type**
chatbotId	string	Chatbot ID
payload	Object	Payload of the event. **Learn more**

Event types

These are the list of events supported in webhooks:

- `leads.submit` : When a customer submit his info (Name, Email, and Phone) to your chatbot.

Event payload

The payload of each event:

```
  customerEmail: string,
  customerName: string,
  customerPhone: string
}
```

Receiving the request

You can receive the payload by accessing the body same as any request. But it is recommended to to check the request header `x-chatbase-signature` for securing your endpoint from spam from anyone knows your endpoint.

You can achieve this by using SHA-1 (Secure Hash Algorithm 1) function to generate a signature for the request and compare it with `x-chatbase-signature` found in the request headers. If the are identical then the request is from Chatbase.

Next.js

```
import crypto from 'crypto'
import {NextApiRequest, NextApiResponse} from 'next'
import getRawBody from 'raw-body'

// Raw body is required.
export const config = {
  api: {
    bodyParser: false,
  },
}

async function webhookHandler(req: NextApiRequest, res: NextApiResponse) {
  if (req.method === 'POST') {
    const {SECRET_KEY} = process.env
    const requestBodySignature = sha1(rawBody, SECRET_KEY)

    if (requestBodySignature !== req.headers['x-chatbase-signature']) {
      return res.status(400).json({message: "Signature didn't match"})
    }

    const receivedJson = await JSON.parse(rawBody.toString())

    console.log('Received:', receivedJson)

    /*
    Body example for leads.submit event
    {
      eventType: 'leads.submit',
      chatbotId: 'xxxxxxxx',
      payload: {
        conversationId: 'xxxxxxxx',
        customerEmail: 'example@chatbase.co',
        customerName: 'Example',
        customerPhone: '123'
      }
    }
    */

    res.status(200).end('OK')
  } else {
    res.setHeader('Allow', 'POST')
    res.status(405).end('Method Not Allowed')
  }
}

function sha1(data: Buffer, secret: string): string {
  return crypto.createHmac('sha1', secret).update(data).digest('hex')
}
```

Node.js

```
import crypto from 'crypto'
import {Request, Response} from 'express'

// Note: In this example json body parser is enabled in the app

export async function webhookHandler(req: Request, res: Response) {
  if (req.method === 'POST') {
    const {SECRET_KEY} = process.env

    if (typeof SECRET_KEY != 'string') {
      throw new Error('No secret key found')
    }

    const receivedJson = req.body

    const rawBody = Buffer.from(JSON.stringify(receivedJson))

    const bodySignature = sha1(rawBody, secretKey)

    if (requestBodySignature !== req.headers['x-chatbase-signature']) {
      return res.status(400).json({message: "Signature didn't match"})
    }

    console.log('Received:', receivedJson)

    /*
    Body example for leads.submit event
    {
      eventType: 'leads.submit',
      chatbotId: 'xxxxxxxx',
      payload: {
        conversationId: 'xxxxxxxx',
        customerEmail: 'example@chatbase.co',
        customerName: 'Example',
    res.status(200).end('OK')
  } else {
    res.setHeader('Allow', 'POST')
    res.status(405).end('Method Not Allowed')
  }
}

function sha1(data: Buffer, secret: string): string {
  return crypto.createHmac('sha1', secret).update(data).digest('hex')
}
```

‹ Get leads