

[Documentation](#)[API Reference](#)[Knowledge Base](#)

## Documentation

[Introduction](#)

## Quickstart

[Node.js](#)[Serverless](#)[PHP](#)[Ruby](#)[Python](#)[Go](#)[Rust](#)[Elixir](#)[Java](#)[.NET](#)[SMTP](#)

## Learn

[Sending](#)[Receiving](#)[Introduction](#)[Custom Domains](#)[Attachments](#)[Email Content](#)[Forward Emails](#)[Reply to Emails](#)[Audience](#)[Domains](#)[API Keys](#)[Webhooks](#)[Broadcasts](#)[Templates](#)[Settings](#)

## Resources

[Examples](#)[SDKs](#)[Security](#)[Integrations](#)[Receiving](#)

## Receiving Emails

[Learn how to receive emails via webhooks.](#)

Resend supports receiving emails (commonly called inbound) in addition to sending emails.

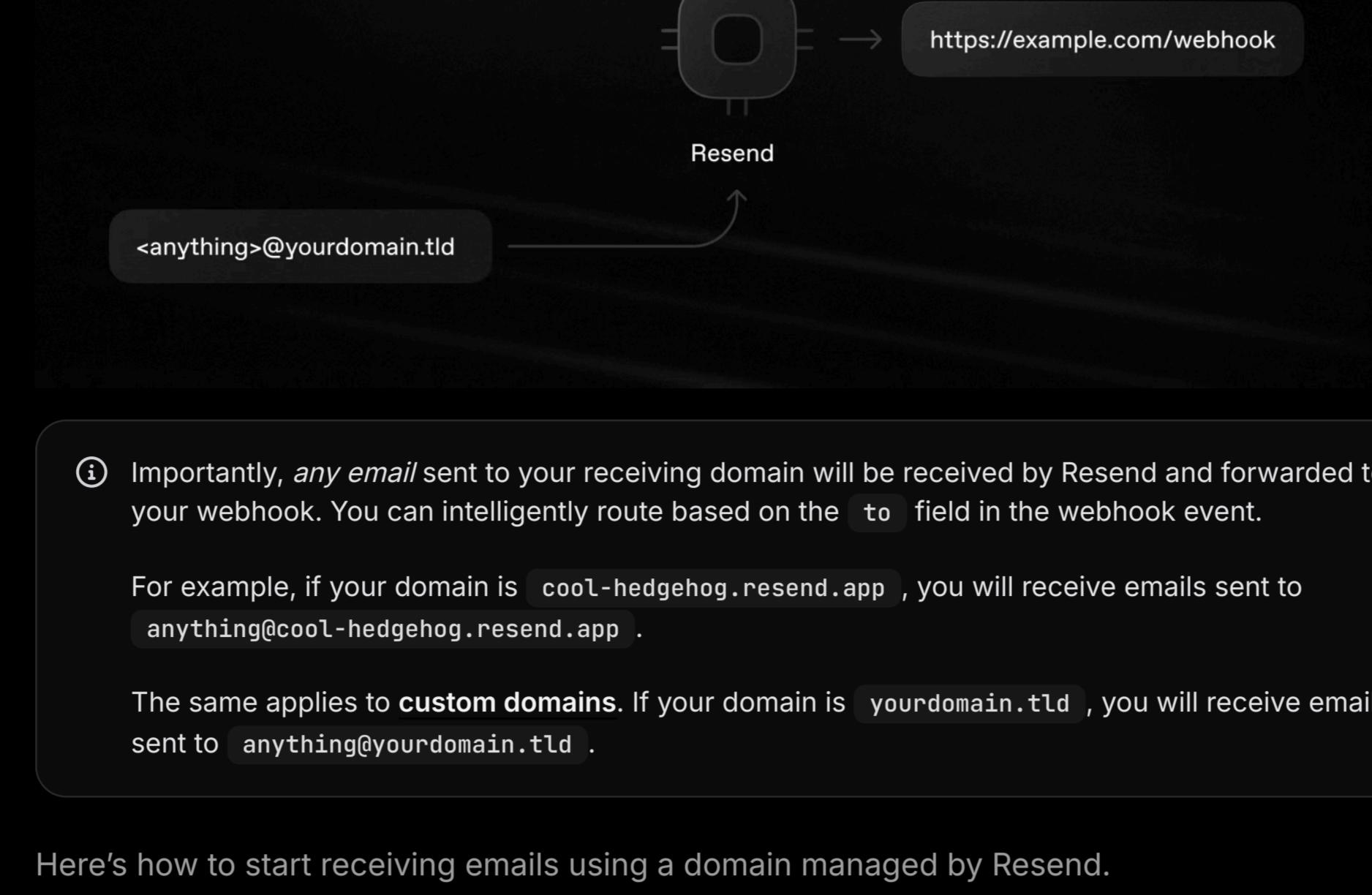
This is useful for:

- Receiving support emails from users
- Processing forwarded attachments
- Replying to emails from customers

### How does it work

Resend processes all incoming emails for your receiving domain, parses the contents and attachments, and then sends a `POST` request to an endpoint that you choose.

To receive emails, you can either use a domain managed by Resend, or [set up a custom domain](#).



ⓘ Importantly, any email sent to your receiving domain will be received by Resend and forwarded to your webhook. You can intelligently route based on the `to` field in the webhook event.

For example, if your domain is `cool-hedgehog.resend.app`, you will receive emails sent to `anything@cool-hedgehog.resend.app`.

The same applies to custom domains. If your domain is `yourdomain.tld`, you will receive emails sent to `anything@yourdomain.tld`.

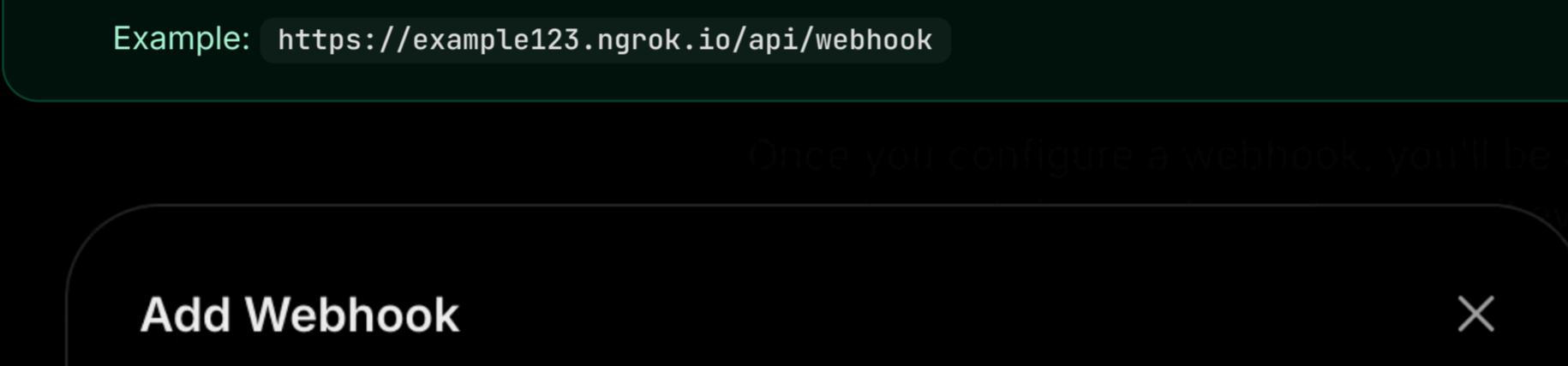
### 1. Get your .resend.app domain

Any emails sent to an `<anything>@<id>.resend.app` address will be received by Resend and forwarded to your webhook.

To see your Resend domain:

1. Go to the [Emails page](#).
2. Select the “Receiving” tab.
3. Click the three dots button and select “Receiving address.”

#### Emails

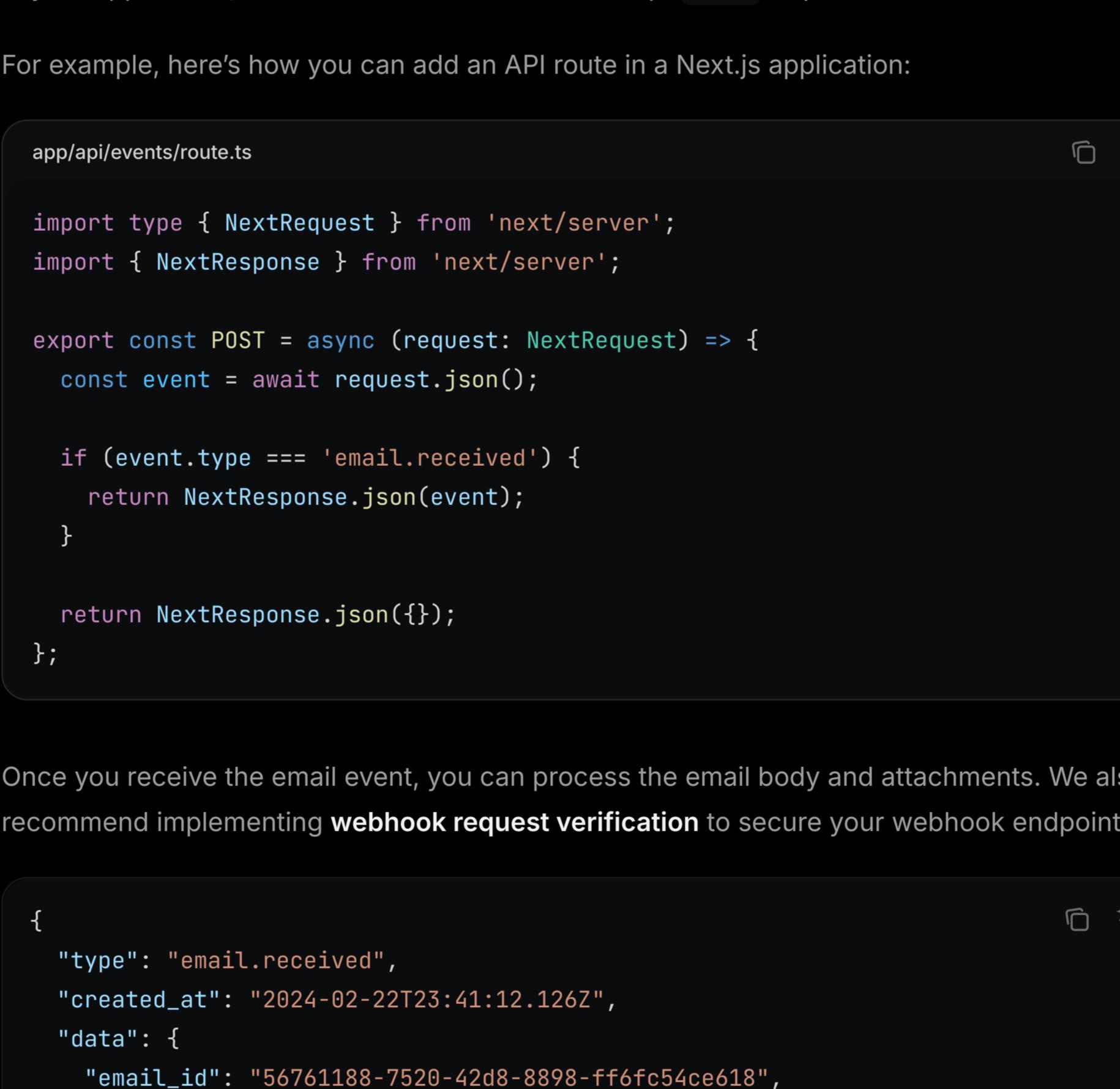


### 2. Configure webhooks

1. Go to the [Webhooks](#) page.
2. Click [Add Webhook](#).
3. Enter the URL of your webhook endpoint.
4. Select the event type `email.received`.
5. Click [Add](#).

ⓘ For development, you can create a tunnel to your localhost server using a tool like `ngrok` or `Code Port Forwarding`. These tools serve your local dev environment at a public URL you can use to test your local webhook endpoint.

Example: `https://example123.ngrok.io/api/webhook`



### 3. Receive email events

In your application, create a new route that can accept `POST` requests.

For example, here's how you can add an API route in a Next.js application:

```
app/api/events/route.ts
import type { NextRequest } from 'next/server';
import { NextResponse } from 'next/server';

export const POST = async (request: NextRequest) => {
  const event = await request.json();

  if (event.type === 'email.received') {
    return NextResponse.json(event);
  }

  return NextResponse.json({});
};
```

Once you receive the email event, you can process the email body and attachments. We also recommend implementing [webhook request verification](#) to secure your webhook endpoint.

```
{
  "type": "email.received",
  "created_at": "2024-02-22T23:41:12.126Z",
  "data": {
    "email_id": "56761188-7520-42d8-8898-ff6fc54ce618",
    "created_at": "2024-02-22T23:41:11.894719+00:00",
    "from": "Alice <onboarding@resend.dev>",
    "to": ["delivered@resend.dev"],
    "bcc": [],
    "cc": [],
    "message_id": "<example+123>",
    "subject": "Sending this example",
    "attachments": [
      {
        "id": "2a0c9ce0-3112-4728-976e-47ddcd16a318",
        "filename": "avatar.png",
        "content_type": "image/png",
        "content_disposition": "inline",
        "content_id": "img001"
      }
    ]
  }
}
```

### What can you do with Receiving emails

Once you receive an email, you can process it in a variety of ways. Here are some common actions you can take:

- Get email content
- Process attachments
- Forward emails to another address
- Reply to emails in the same thread

ⓘ Webhooks do not include the email body, headers, or attachments, only their metadata. You must call the [Received emails API](#) or the [Attachments API](#) to retrieve them. This design choice supports large attachments in serverless environments that have limited request body sizes.

## FAQ

ⓘ Will I receive emails for any address at my domain?

ⓘ Can I receive emails on a subdomain?

ⓘ Should I add the 'MX' records for my root domain or a subdomain?

ⓘ Will I lose my emails if my webhook endpoint is down?

ⓘ How can I make sure that it's Resend who's sending webhooks?

Was this page helpful? Yes No

Tags

Custom Domains

Ask a question...

...

