



## Documentation

### Getting Started

What is Segment?  
[How Segment Works](#)  
Getting Started Guide  
A Basic Segment Installation  
Planning a Full Installation  
A Full Segment Installation  
Sending Data to Destinations  
Testing and Debugging  
What's Next  
Use Cases

### Guides

### Connections

### Unify

### Engage

### Privacy

### Protocols

### Segment App

### API

### Partners

### Glossary

### Config API

### Help

Server

Server

[Sentry](#) is open-source error tracking that helps developers monitor and fix crashes in real time. Iterate continuously. Boost efficiency. Improve user experience. The `analytics.js` Sentry Destination is open-source. You can browse the code [on GitHub](#).

## Getting Started

- 1 From the Segment web app, click **Catalog**.
- 2 Search for "Sentry" in the Catalog, select it, and choose which of your sources to connect the destination to.
- 3 In the destination settings, enter your "Public DSN".
- 4 Segment automatically initializes Sentry with your "Public DSN" upon loading `analytics.js`.
- 5 Sentry will automatically start tracking errors in your app's JavaScript.

## Identify

If you're not familiar with the Segment Specs, take a look to understand what the [Identify method](#) does. An example call would look like:

```
analytics.identify('userId123');
```

When you call `identify` Segment calls `Sentry.setUser` by passing in the traits you provided. Segment maps the `userId` you provide as `traits.id`.

## Troubleshooting

Since Sentry is an error tracking application, Segment needs to send something broken from the website to see the event appear in the Sentry dashboard. For example, firing the following error event from the web console, `Sentry.captureException(new Error("Some error"))`; Segment can see the relevant event payload with the `event_id` information.

The screenshot shows the Chrome DevTools interface. The top panel displays a network request to `?sentry_key=502a86940c7845ee9b60ee524a...`. The 'Payload' tab is selected, showing the 'Request Payload' which contains an exception object. The `event_id` is highlighted in green and pointed to by a red arrow. The bottom panel shows the console log with the `Sentry.captureException(new Error("some Error"))` call and the returned `event_id` value, which is also highlighted with a red box.

This error will appear in Sentry with the same `event_id` displayed in developer tools. This helps verify that Segment has the correct implementation of the destination.

## Settings

Segment lets you change these destination settings from the Segment app without having to touch any code.

SETTING	DESCRIPTION
Public DSN (required)	<code>string</code> . You can find your Public DSN by going to Sentry and clicking Projects & Teams > Any Project > Settings > Client Keys (DSN). This should be the PUBLIC DSN since it does not contain a secret.

SETTING	DESCRIPTION
Set Release by Property	<code>string</code> . Dynamically track the version of your application in Sentry. If the property you set here is on the global window object (example: you input 'my_custom_version', and window.my_custom_version = '2.4.5') then we will use that version. If not, we will default to the manually-set Release property. We recommend that this value is unique to prevent it from being overwritten by other integrations.
Debug Mode	<code>boolean</code> , defaults to <code>FALSE</code> .  Toggle this setting to "true" to enable Sentry's debug mode. In debug mode, the Sentry client will log activity to your browser's console. No events will be sent to your Sentry instance when this setting is enabled.
Environment	<code>string</code> . Segment will set the value you enter here as the value of Sentry's <code>environment</code> configuration option.
Ignore Errors	<code>array</code> , defaults to .  A list of error messages to be filtered out before being sent to Sentry. You can add either regular expressions or strings.
Ignore Urls	<code>array</code> , defaults to .  The inverse of whitelistUrls and similar to ignoreErrors, but will ignore errors from whole urls matching a regex pattern or an exact string.
Include Paths	<code>array</code> , defaults to .  An array of regex patterns to indicate which urls are a part of your app in the stack trace. All other frames will appear collapsed in Sentry's dashboard to make it easier to discern between frames that happened in your code versus other code. By default, Sentry displays all frames as expanded. Once you add values to this setting, all frames excepts for those you've added here will appear as collapsed in Sentry. You can still manually expand all frames from a stack trace in your Sentry dashboard. It would be suggested to add the current page url, and the host for your CDN here.
Logger	<code>string</code> . The name of the logger used by Sentry.
Release	<code>string</code> . Track the version of your application in Sentry.
Server Name	<code>string</code> . Typically this would be the server name, but that doesn't exist on all platforms. Instead you may use something like the device ID, as it indicates the host which the client is running on.
Whitelist Urls	<code>array</code> , defaults to .  The inverse of ignoreUrls. Only report errors from whole urls matching a regex pattern or an exact string. whitelistUrls should match the url of your actual JavaScript files. It should match the url of your site if and only if you are inlining code inside <code>&lt;script&gt;</code> tags. Not setting this value is equivalent to a catch-all and will not filter out any values.

This page was last modified: 27 Oct 2023

## Need support?

Questions? Problems? Need more info? Contact Segment Support for assistance!

[Visit our Support page](#)

## Help improve these docs!

 [Edit this page](#)

 [Request docs change](#)

## Was this page helpful?

 [Yes](#)

 No

## Get started with Segment

Segment is the easiest way to integrate your websites & mobile apps data to over 300 analytics and growth tools.

[Request Demo](#)

or

[Create free account](#)

© 2025 Segment.io, Inc.

[Privacy](#)

[Terms](#)

[Website Data Collection Preferences](#)

