SENTRY

Admin Quick Reference Guide

ORGANIZATION LEVEL CONFIGURATIONS

Implementing Single Sign On

SSO allows you to manage your organization's entire membership via a third-party provider. If your SSO solution also supports SCIM, you can provision and deprovision users and assign them to teams through your identity provider and avoid the hassle of doing it manually.

The alternative to SSO and SCIM is to manually create users and teams and assign organizational-level and team-level roles.

Key Integrations with Sentry

There are three key types of integrations that have the largest impact on developer productivity and experience when using Sentry.

Source Code Management (SCM)

Integrating your SCM solution with Sentry helps hone in on problematic commits. Commit tracking allows users to see the most recent changes to files found in the issue's stack trace with Suspect Commits.

Collaboration and Notification

You can customize alerts based on type (critical vs. non-critical) and hours (business vs. non-business). Sentry supports a number of collaboration platforms.

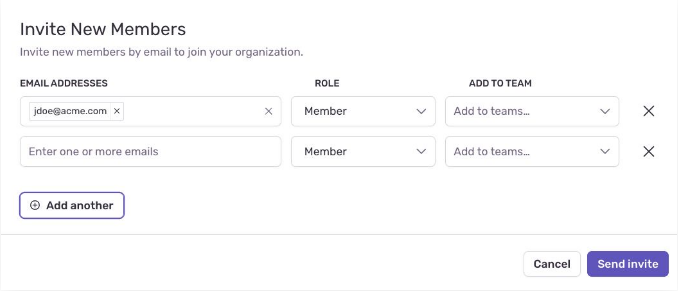
Issue and Bug Tracking (aka Ticketing)

Sentry can automate manual tickets and updates across other solutions for your organization. Save time keeping track of issues in multiple locations without the hassle of manual updates.

PROJECT LEVEL CONFIGURATION

We'll start with the basics

1. If you implemented SSO you can skip this step, if not, the first thing to do is to create users and teams manually. Be sure to pay attention to the different roles when creating the users, this is critical to ensure users will only perform tasks that they are allowed to.

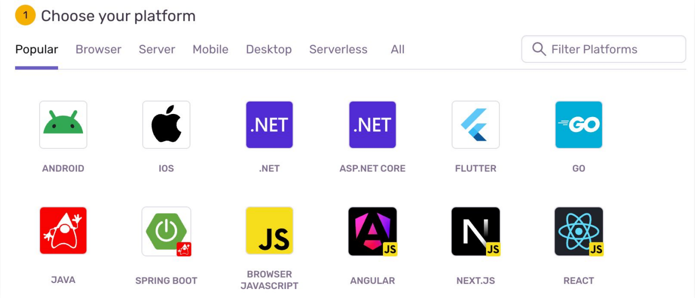


Invite New Members

Invite new members by email to join your organization.

EMAIL ADDRESSES	ROLE	ADD TO TEAM
jdoe@acme.com	Member	Add to teams...
Enter one or more emails	Member	Add to teams...

2. Now it is time to create your projects. Pro Tip: Try to create a naming convention that is meaningful for your entire organization (not only for your team). Oftentimes organizations have conventions for their code repositories and use that for their Sentry projects.



1 Choose your platform

Don't forget to...

If you have a JavaScript based project, upload your source maps. This will help developers debug issues faster by seeing their source code in their stack trace. For code languages and frameworks that don't support source maps, consider using debug files, Sentry supports a multitude of debug files such as: dSYM files, ELF symbols, PDB files, among others.

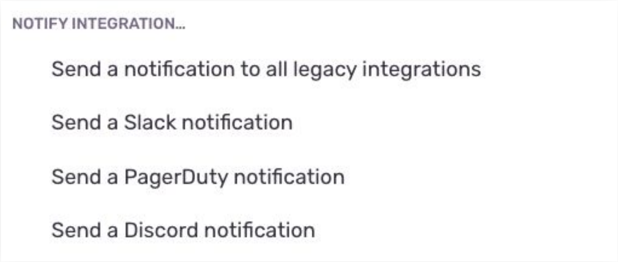
Also, be sure to set up your environments so developers can have a better understanding of which environments (Dev, Test, QA, Production) are being impacted. Be sure to set up Releases as well to trace issues back to specific software releases.

Important Note: Some of these tasks are performed after the project is created in the SDK, so work with your development team to make it happen.

3. Now it is time to create your alerts. Keep in mind that you have different types of alerts. Issue Alerts will trigger whenever any issue in a project matches a specified criteria.

Metric Alerts help reduce notification noise by setting a standard threshold for when an issue needs your attention.

Don't forget to send alerts to Notification tools like: Slack, Microsoft Teams, PagerDuty, Opsgenie, Discord and others.



NOTIFY INTEGRATION...

Send a notification to all legacy integrations

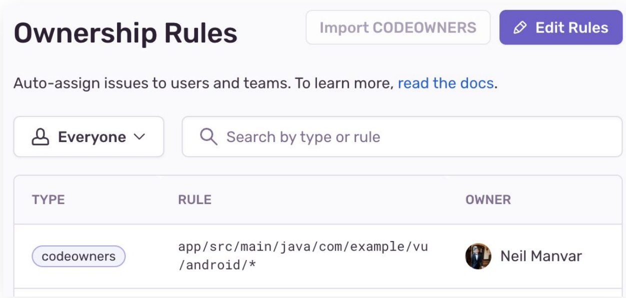
Send a Slack notification

Send a PagerDuty notification

Send a Discord notification

Take your setup to the next level

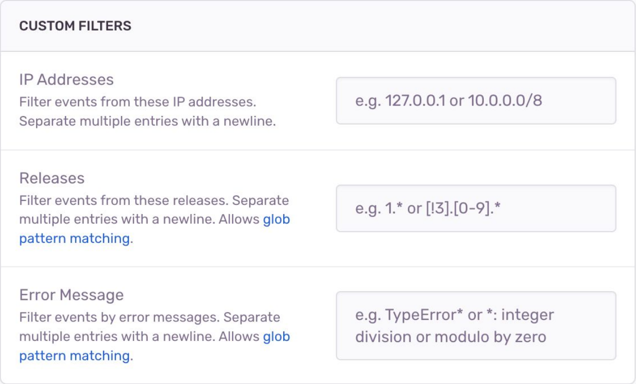
1. By configuring Automatic Issue Assignment you can get the issue in front of the right person fast. That can be done either through [Ownership Rules]() or [CODEOWNER]() files from GitHub or GitLab.



Auto-assign issues to users and teams. To learn more, read the docs.

TYPE	RULE	OWNER
codeowners	app/src/main/java/com/example/vu /android/*	Neil Manvar

2. Setting up filtering will help keep noise to a minimum. Consider implementing [client-side]() and [server-side]() filters for your noisey projects. Keep in mind that client-side filters are set up via the SDK.



CUSTOM FILTERS

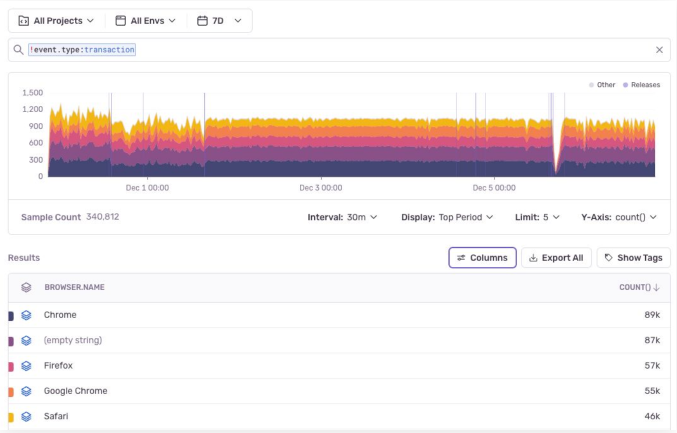
IP Addresses
Filter events from these IP addresses. Separate multiple entries with a newline.
e.g. 127.0.0.1 or 10.0.0.0/8

Releases
Filter events from these releases. Separate multiple entries with a newline. Allows glob pattern matching.
e.g. 1.* or [!3].[0-9].*

Error Message
Filter events by error messages. Separate multiple entries with a newline. Allows glob pattern matching.
e.g. TypeError* or *: integer division or modulo by zero

Issue [auto-resolution]() helps keep your Issues Stream clean so you only focus on actionable issues by closing the ones you don't care about automatically.

3. Consider creating [Discover Queries]() and [Dashboards]() for application insights. [Discover]() allows you to dig into Error and Performance Issues with all the flexibility you need to slice and dice the data any way you want.



!event.type:transaction

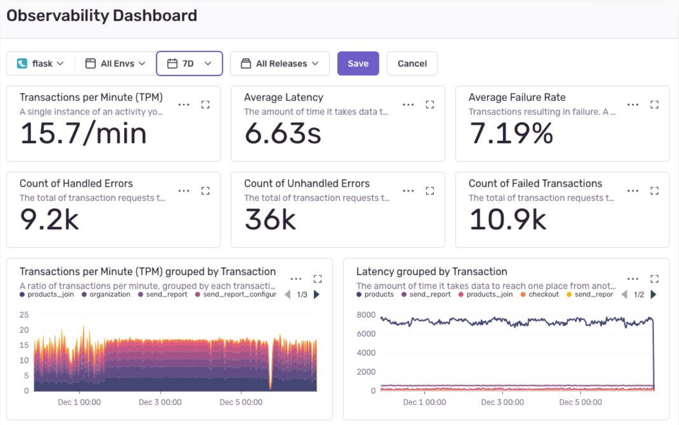
Sample Count 340,812

Interval: 30m
Display: Top Period
Limit: 5
Y-Axis: count()

BROWSER.NAME	COUNT() ↓
Chrome	89k
(empty string)	87k
Firefox	57k
Google Chrome	55k
Safari	46k

Dashboards can provide additional insights that can help with your continuous improvement processes. Pro Tip: Do not start a dashboard from scratch, there are several [Dashboard Templates]() to pick from.

As an Admin, you can create templates for your teams to start with (e.g. Custom Web Template). With a starter template, your teams can just copy and modify based on their own needs.



Observability Dashboard

Transactions per Minute (TPM)
A single instance of an activity yo...
15.7/min

Average Latency
The amount of time it takes data t...
6.63s

Average Failure Rate
Transactions resulting in failure. A ...
7.19%

Count of Handled Errors
The total of transaction requests t...
9.2k

Count of Unhandled Errors
The total of transaction requests t...
36k

Count of Failed Transactions
The total of transaction requests t...
10.9k

Transactions per Minute (TPM) grouped by Transaction
A ratio of transactions per minute, grouped by each transacti...

Latency grouped by Transaction
The amount of time it takes data to reach one place from anot...

Keep your quota under control

There are multiple strategies for managing your event quota. Here are some tips to keep noise to a minimum and your quota under control.

Strategy

- [Spike Protection]() protects your projects from an anomaly in project event volume.
- [Spend Allocation]() allows you to reserve a given amount of volume for specific projects.
- Filters are one of the best ways to avoid noisey issues you don't care about.
- [Metric Alerts]() also help you know when there's a sudden spike in events for an issue or project impacting your quota.
- [Rate Limiting]()* can stop a single project from depleting your entire error event quota by allowing you to set a maximum number of error events a project can receive over a period of time.

**Not applicable to Transactions.*

To learn more about [quota management best practices]() and [what counts towards your quota](), visit our documentation.

BE IN THE KNOW

Found a [bug](), have a [feature request]() or question? Let us know on GitHub and join the Sentry team on [Discord]().

Subscribe to [status updates]() and learn about the latest product updates on our [blog]() and the [Sentry changelog]().

To get access to our full documentation go to [docs.sentry.io]()