Trust Center

Cursor

Modern engineering organizations of all sizes use Cursor to accelerate and reshape the way code is written, read, and edited. Over 30,000 companies trust Cursor as their IDE of choice.

Cursor operates a continuously monitored and 3rd-party audited security program. This page provides access to trust resources including policies, audits, pentests, and real-time status against security controls.

Cursor Security Page

Overview Resources Controls Subprocessors FAQ

FAQ

Security

How does Cursor process AI requests?

To provide its features, Cursor makes AI requests to our server. This happens for many different reasons. For example, we send AI requests when you ask questions in chat, we send AI requests on every keystroke so that Cursor Tab can make suggestions for you, and we may also send AI requests in the background for building up context or looking for bugs to show you.

An AI request generally includes context such as your recently viewed files, your conversation history, and relevant pieces of code based on language server information. This code data is sent to our infrastructure on AWS, and then to the appropriate language model inference provider (Fireworks/OpenAI/Anthropic/Google). Note that the requests always hit our infrastructure on AWS even if you have configured your own API key for OpenAI in the settings.

We currently do not have the ability to direct-route from the Cursor app to your enterprise deployment of OpenAI/Azure/Anthropic, as our prompt-building happens on our server, and

Top

our custom models on Fireworks are critical in providing a good user experience. We do not yet have a self-hosted server deployment option.

Does Cursor use infrastructure in China?

None of our infrastructure is in China. We do not directly use any Chinese company as a subprocessor, and to our knowledge none of our subprocessors do either.

How is my code data protected?

Privacy Mode is enforced by default for all Business plan users, ensuring that no raw code data is stored or used for training purposes. We maintain zero-data retention agreements with our vendors and only store essential user information, specifically names and email addresses. All data is protected with TLS 1.2+ encryption during transit and AES-256 encryption while at rest.

Our code embeddings architecture is designed with security at its core, storing no raw code in the cloud. Instead, the system maintains vector embeddings, semi-encrypted file pathways, and code location pointers. The process begins with local file and project indexing, followed by the creation of vector embeddings. These embeddings generate pointers that map to specific code locations. When a query is made, the system retrieves the top 50-100 relevant pointers, which the local client then translates. The server uses this information to create prompts with appropriate context.

Several security measures are implemented throughout this process. Each embedding consists of 512 tokens and generates 4-5 kilobyte vectors. File paths are semi-encrypted, allowing visibility of the directory hierarchy while keeping file and folder names encrypted. The system also respects both .gitignore and .cursorignore files to maintain user privacy preferences.

What authentication and access controls exist?

Users can authenticate through email or GitHub login by default, while Business plan customers have access to SAML 2.0 SSO integration. Just-in-Time authentication is available, though we currently do not support SCIM protocols.

Our tenant structure operates with separate tenants for privacy and non-privacy modes within a multi-tenant environment that uses logical segregation. We do not currently offer individually isolated tenants for customers.

Top

What compliance and monitoring is in place?

We maintain and operate mature internal controls, including annual security training for all employees and regular account access reviews that comply with SOC2 standards. We maintain custom enterprise contracts with our AI vendors, which undergo annual reviews.

For system monitoring, we implement basic logging to track system performance and collect limited telemetry data. While we conduct regular security scanning, the specific frequency is still being determined. Currently, we do not expose audit logging capabilities to clients.

Do you support detailed logging?

Currently, we don't support detailed logging. Admins can access basic log and usage information in the Admin Dashboard.

Do you encrypt data at rest and in transit?

Cursor encrypts data in transit with a minimum of TLS 1.2, while all data at rest is secured using AES 256 encryption.

What deployment options are available?

Our solution is currently available only through cloud deployment, though we do support installation on local machines. While we don't offer a sandbox environment, we do provide a free tier for testing and evaluation.

Cursor infrastructure runs on AWS, with our primary region in the United States and secondary regions in Tokyo and London. At present, we don't support hybrid deployments, on-premises installations, or VPC configurations.

Privacy

How does enabling Privacy Mode change the data Cursor processes and stores?

Top

Privacy mode can be enabled during onboarding or in settings. When it is enabled, we guarantee that code data is not stored in plaintext at our servers or by our subprocessors. Privacy mode can be enabled by anyone (free or Pro user), and is by default forcibly enabled for any user that is a member of a team.

We take the privacy mode guarantee very seriously. About 50% of all Cursor users have privacy mode enabled. You can read more about the privacy guarantee in our Privacy Policy.

If you enable "Privacy Mode" in Cursor's settings, zero data retention will be enabled, and none of your code will ever be stored or trained on by us or any third-party.

If you choose to keep "Privacy Mode" off:

- We collect telemetry and usage data. This may include prompts, editor actions, code snippets, and edits made to this code. We use this data to evaluate and improve our AI features.
- If you use autocomplete, Fireworks, our inference provider, may also collect prompts to improve inference speed.

Who owns the code created while using Cursor?

Cursor customers own all the code generated by Cursor.

How long will you retain my data?

If you already have an account on the Apps, you may access, update, alter, or delete your basic user profile information by logging into your account and updating profile settings.

Anysphere will retain your information for as long as your account is active or as needed to perform our contractual obligations, provide you services through the App, to comply with legal obligations, resolve disputes, preserve legal rights, or enforce our agreements. Retention periods will be determined taking into account the type of information that is collected and the purpose for which it is collected, bearing in mind the requirements applicable to the situation and the need to destroy outdated, unused information at the earliest reasonable opportunity. For instance, in respect of data held for the management of customers and potential customers, we consider the lead time necessary to develop and maintain our commercial relationships and how recent our interactions are with you. We may rectify, update or remove incomplete or inaccurate information, at any time and at our own discretion. For more information on our retention periods you can contact us by email with the subject line "Privacy Concern" at hi@cursor.com.

Please note that due to the open source nature of our products, services, and community, we may retain limited personal information indefinitely in order to ensure transactional integrity and nonrepudiation. For example, if you provide your information in connection with a blog

Top

post, GitHub issue or comment, we may display that information even if you have deleted your account as we do not automatically delete community posts.

Does Cursor transfer data across borders?

The Apps are hosted in the United States and the personal information we collect will be stored and processed on our servers in the United States. Our employees, contractors and affiliated organizations that process information for us as described above may be located in the United States or in other countries outside of your home country which may have different data protection standards to those which apply in your home country.

Where your personal information is transferred outside of the EEA, Switzerland and UK and where this is to a country which is not subject to an adequacy decision by the EU Commission or considered adequate as determined by applicable data protection laws, we will take steps to ensure your personal information is adequately protected by safeguards such as Standard Contractual Clauses (“SCCs”) approved by the EU Commission or by the UK Government. A copy of the relevant mechanism can be obtained for your review on request by emailing us with the subject line "Data Transfers" at hi@cursor.com.

How do you handle the data we collect?

You may choose to interact with our Apps in ways that provide us with your personal information. In some instances, a User ID is generated for form and URL tracking, page views, page pings and usage counts in order to ascertain product performance and development. The amount and type of information that Anysphere gathers depends on the nature of your interaction with us, as well as the amount of information you choose to share. For example, we ask visitors who use our community Discord, or our forum, to provide a username and email address. We will also collect the information you provide with us in connection with creating an account on the App. In each case, Anysphere collects such personal information only insofar as is necessary or appropriate to fulfill the purpose of your interaction with or your request to Anysphere. We will not disclose your personal information other than as described in this Privacy Policy.

Like most website operators, Anysphere automatically collects (i) technical information about your device including your device's internet protocol (IP) address, device type (e.g., phone, tablet), unique identifiers (including identifiers used for advertising purposes), language settings, mobile device carrier, radio/network information (e.g., WiFi, LTE, 4G), and general location information such as city, state or geographic area; and (ii) information about your visit to our Apps and online activity data (such as the referral URL, the content viewed and the content interacted with). Some of this information is collected using cookies, web beacons and related local storage technologies. See below for further information on these technologies. We collect this information to better understand how visitors use our Apps, to improve our Apps and experience for visitors, and to monitor the security of the Apps.

Top

We may aggregate all information (including your personal information) collected from our Apps for our own statistical and analytics purposes and share such aggregated information with third parties for our own promotional purposes (e.g. by publishing a report on trends in the usage of our Apps).

What personal information does Cursor collect?

You may choose to interact with our Apps in ways that provide us with your personal information. In some instances, a User ID is generated for form and URL tracking, page views, page pings and usage counts in order to ascertain product performance and development. The amount and type of information that Anysphere gathers depends on the nature of your interaction with us, as well as the amount of information you choose to share. For example, we ask visitors who use our community Discord, or our forum, to provide a username and email address. We will also collect the information you provide with us in connection with creating an account on the App. In each case, Anysphere collects such personal information only insofar as is necessary or appropriate to fulfill the purpose of your interaction with or your request to Anysphere. We will not disclose your personal information other than as described in this Privacy Policy.

Like most website operators, Anysphere automatically collects (i) technical information about your device including your device's internet protocol (IP) address, device type (e.g., phone, tablet), unique identifiers (including identifiers used for advertising purposes), language settings, mobile device carrier, radio/network information (e.g., WiFi, LTE, 4G), and general location information such as city, state or geographic area; and (ii) information about your visit to our Apps and online activity data (such as the referral URL, the content viewed and the content interacted with). Some of this information is collected using cookies, web beacons and related local storage technologies. See below for further information on these technologies. We collect this information to better understand how visitors use our Apps, to improve our Apps and experience for visitors, and to monitor the security of the Apps.

We may aggregate all information (including your personal information) collected from our Apps for our own statistical and analytics purposes and share such aggregated information with third parties for our own promotional purposes (e.g. by publishing a report on trends in the usage of our Apps).

Does Cursor sign Business Associate Agreements (BAAs)?

No, Cursor does not currently sign BAAs.

Misc

Top

What customer support is available?

We provide extensive documentation including a comprehensive troubleshooting guide, a section covering common issues, detailed network configuration guides, and steps for troubleshooting VPN connectivity issues.

For support, users can reach out via email at hi@cursor.com.

Enterprise customers have access to additional support options including a shared Slack channel and dedicated support reps.

All documentation is readily accessible at docs.cursor.com.

How does Cursor index my codebase?

Cursor allows you to semantically index your codebase, which allows it to answer questions with the context of all of your code as well as write better code by referencing existing implementations. Codebase indexing is enabled by default, but can be turned off during onboarding or in the settings.

Our codebase indexing feature works as follows: when enabled, it scans the folder that you open in Cursor and computes a Merkle tree of hashes of all files. Files and subdirectories specified by `'.gitignore'` or `'.cursorignore'` are ignored. The Merkle tree is then synced to the server. Every 10 minutes, we check for hash mismatches, and use the Merkle tree to figure out which files have changed and only upload those.

At our server, we chunk and embed the files, and store the embeddings in Turbopuffer. To allow filtering vector search results by file path, we store with every vector an obfuscated relative file path, as well as the line range the chunk corresponds to. We also store the embedding in a cache in AWS, indexed by the hash of the chunk, to ensure that indexing the same codebase a second time is much faster (which is particularly useful for teams).

At inference time, we compute an embedding, let Turbopuffer do the nearest neighbor search, send back the obfuscated file path and line range to the client, and read those file chunks on the client locally. We then send those chunks back up to the server to answer the user's question. This means that no plaintext code is stored on our servers or in Turbopuffer.

Some notes:

-> While a `'.cursorignore'` file can prevent files from being indexed, those files may still be included in AI requests, such as if you recently viewed a file and then ask a question in the chat. We are considering adding a `'.cursorban'` file to address the use case of wanting to block files from being sent up in any request — please make a forum post or reach out at hi@cursor.com if this is a feature that would be interesting to you.

-> File path obfuscation details: the path is split by `'/'` and `'.'` and each segment is encrypted with a secret key stored on the client and a deterministic short 6-byte nonce. This

Top

leaks information about directory hierarchy, and will have some nonce collisions, but hides most information.

-> Embedding reversal: academic work has shown that reversing embeddings is possible in some cases. Current attacks rely on having access to the model and embedding short strings into big vectors, which makes us believe that the attack would be somewhat difficult to do here. That said, it is definitely possible for an adversary who breaks into our vector database to learn things about the indexed codebases.

-> When codebase indexing is enabled in a Git repo, we also index the Git history. Specifically, we store commit SHAs, parent information and obfuscated file names (same as above). To allow sharing the data structure for users in the same Git repo and on the same team, the secret key for obfuscating the file names is derived from hashes of recent commit contents. Commit messages and file contents or diffs are not indexed.

-> Our indexing feature often experiences heavy load, which can cause many requests to fail. This means that sometimes, files will need to be uploaded several times before they get fully indexed. One way this manifests is that if you check the network traffic to `‘repo42.cursor.sh’`, you may see more bandwidth used than expected.

Top