Configuring network settings for GitHub Copilot

In this article

You can connect to Copilot through an HTTP proxy and use custom certificates.

JetBrains IDEs Visual Studio Visual Studio Code

GitHub Copilot can be managed through personal accounts with GitHub Copilot for Individuals or through organization or enterprise accounts with GitHub Copilot for Business.

GitHub Copilot is free to use for verified students, teachers, and maintainers of popular open source projects. For more information, see "About billing for GitHub Copilot."

Note: GitHub Copilot is not currently available for use with Visual Studio for Mac.

Introduction

By default, GitHub Copilot connects to GitHub's server directly from your environment, via a secure HTTPS connection. You don't necessarily need to configure any additional network settings to use Copilot.

Some networks use an HTTP proxy server to intercept Internet traffic before sending it to its intended location. Companies often use an HTTP proxy to detect suspicious traffic or restrict the content entering their networks. If you're working on a corporate network, you may need to configure Copilot to connect via an HTTP proxy.

If you have a license for Copilot for Business, Copilot can read custom SSL certificates installed on a user's machine. This allows a proxy server to be identified as the intended recipient of Copilot's secure connection so network traffic can be inspected. Without a custom certificate, a company can use an HTTP proxy to monitor, route, and terminate Copilot's connection, but cannot inspect the contents of the traffic.

Configuring proxy settings for GitHub Copilot

GitHub Copilot supports basic HTTP proxy setups. If you need to authenticate to a proxy, GitHub Copilot supports basic authentication or authentication with Kerberos. If the proxy URL starts `https://`, the proxy is not currently supported.

You can configure an HTTP proxy for GitHub Copilot in your chosen editor. To view instructions for your editor, use the tabs at the top of this article.

If you don't configure a proxy directly in your editor, GitHub Copilot checks if a proxy URL is set in any of the following environment variables, listed from highest to lowest priority.

- `HTTPS_PROXY`
- `https_proxy`
- `HTTP_PROXY`
- `http_proxy`

> **Note:** You can use any of these variables to store the URL of a standard HTTP proxy. In standard usage, the `http` and `https` portions of these variables refer to the type of request being made, not the URL of the proxy itself. GitHub Copilot does not follow this convention, and uses the URL stored in the variable with the highest priority as the proxy for both HTTP and HTTPS requests.

If you have configured a proxy but are still encountering connection errors, see "Troubleshooting network errors for GitHub Copilot."

Configuring a proxy in a JetBrains IDE

1. In your JetBrains IDE, click the **File** menu (Windows) or the name of the application in the menu bar (macOS), then click **Settings**.
2. Under **Appearance & Behavior**, click **System Settings** and then click **HTTP Proxy**.
3. Select **Manual proxy configuration**, and then select **HTTP**.
4. In the "Host name" field, enter the hostname of your proxy server, and in the "Port number" field, enter the port number of your proxy server.
5. Optionally, to configure Copilot to ignore certificate errors, in the left sidebar, click **Tools**, click **Server Certificates**, then select or deselect **Accept non-trusted certificates automatically**.

   > **Warning:** Ignoring certificate errors can cause security issues and is not recommended.

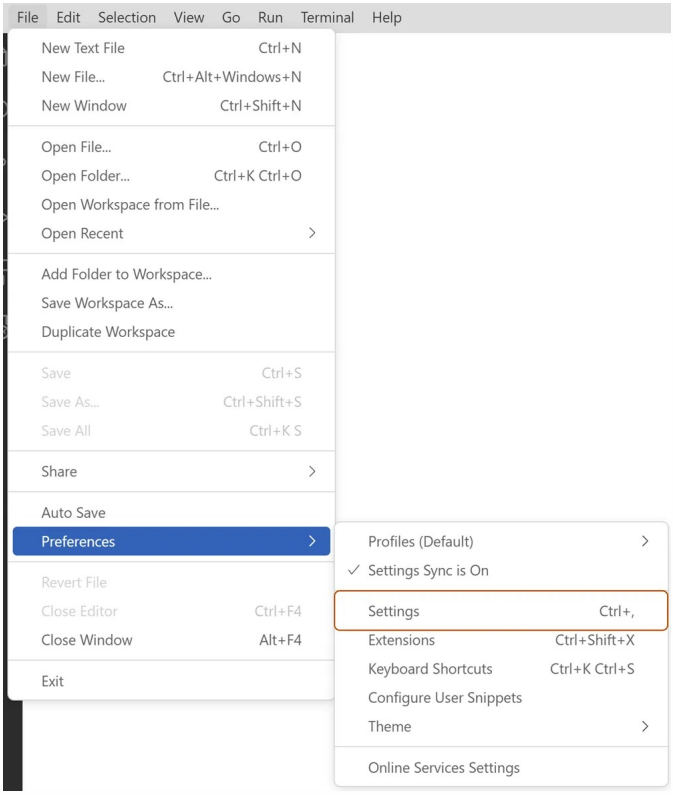
Basic authentication

GitHub Copilot for JetBrains supports basic authentication. To authenticate, you can select **Proxy authentication** on the "Manual proxy configuration" page, then enter your credentials.

This stores your credentials as plaintext in your editor's settings. Alternatively, you may prefer to include your credentials in the proxy URL (for example: `http://USERNAME:PASSWORD@10.203.0.1:5187/`), and then set this URL as one of the supported environment variables listed in "Configuring proxy settings for GitHub Copilot."

Configuring a proxy in Visual Studio Code

1. In the **File** menu, navigate to **Preferences** and click **Settings**.



2. In the left-side panel of the settings tab, click **Application** and then select **Proxy**.
3. In the text box under "Proxy", type the address of your proxy server, for example `http://localhost:3128`.
4. Optionally, to configure Copilot to ignore certificate errors, under "Proxy Strict SSL", select or deselect the checkbox.

   > **Warning:** Ignoring certificate errors can cause security issues and is not recommended.

Basic authentication

GitHub Copilot for VS Code supports basic authentication. To authenticate, you can include your credentials in the proxy URL, for example: `http://USERNAME:PASSWORD@10.203.0.1:5187/`. You can store this URL in your VS Code settings or in one of the environment variables listed in "Configuring proxy settings for GitHub Copilot."

Configuring a proxy in Visual Studio

GitHub Copilot for Visual Studio reads the proxy settings from Windows. For information about configuring proxy settings on Windows, see the instructions under "To set up a proxy server connection manually" in Use a proxy server in Windows in the Microsoft documentation.

Basic authentication

GitHub Copilot for Visual Studio does not retrieve authentication credentials from the Windows settings. If you need to authenticate to a proxy, you can include your credentials in the proxy URL (for example: `http://USERNAME:PASSWORD@10.203.0.1:5187/`), then set this URL as one of the supported environment variables listed in "Configuring proxy settings for GitHub Copilot."

Authentication with Kerberos

Kerberos is an authentication protocol that allows users and services to prove their identity to each other. When a user successfully authenticates, an authentication service grants the user a ticket that gives them access to a service for a period of time. Network administrators may prefer Kerberos to basic authentication because it is more secure and doesn't require sending unencrypted credentials.

GitHub Copilot supports authentication to a proxy with Kerberos. To use Kerberos, you must have the appropriate krb5 library for your operating system installed on your machine, and an active ticket for the proxy service (either created manually with the `kinit` command, or by another application). You can use the `klist` command to check if you have a ticket for the proxy service.

Kerberos uses a service principal name (SPN) to uniquely identify a service instance. By default, the SPN is derived from the proxy URL. For example, if the proxy URL is `http://proxy.example.com:3128`, the SPN is `HTTP/proxy.example.com`.

If the default SPN isn't correct for your proxy, you can override the SPN in VS Code and in JetBrains IDEs. You cannot currently override the default SPN in Visual Studio.

Overriding the default SPN in VS Code

1. Open the VS Code Command Palette by pressing `Shift` + `Command` + `P` (Mac) / `Ctrl` + `Shift` + `P` (Windows/Linux).
2. Type `settings`, then click **Preferences: Open User Settings (JSON)**.
3. In the JSON object, add the following top-level property, replacing `YOUR-SPN` with the correct SPN for your proxy service.

   JSON

   ```
   "http.proxyKerberosServicePrincipal": "YOUR-SPN",
   ```

Overriding the default SPN in JetBrains IDEs

1. In your JetBrains IDE, click the **File** menu (Windows) or the name of the application in the menu bar (macOS), then click **Settings**.
2. In the left sidebar, click **Languages & Frameworks**, then click **GitHub Copilot**.
3. In the "Advanced" section, in the "Override Kerberos Proxy Service Principal Name" field, type the SPN for your proxy service.

Allowing GitHub Copilot to use custom certificates

If your organization uses Copilot for Business, Copilot can read custom SSL certificates installed on a user's machine.

Copilot reads certificates from the operating system's trust store. It also reads extra certificates from the file specified by the standard Node.js environment variable `NODE_EXTRA_CA_CERTS`. For more information, see the Node.js documentation.

Copilot can read certificates regardless of whether a proxy is configured directly on a

user's machine. This allows Copilot to support setups such as transparent proxies or Zscaler.

Installing custom certificates

Generally, if you're using company equipment, your company's IT department should have already installed any required certificates on your machine. If you need to install a certificate, see the following instructions.

> **Warning:** Installing a custom certificate is an instruction for your computer to trust the creator of the certificate, potentially allowing the creator to intercept all Internet traffic from your machine. You should be very careful to verify that you are installing the correct certificate.

- For Windows, see Installing the trusted root certificate in the Microsoft documentation.
- For macOS, see Add certificates to a keychain using Keychain Access on Mac in the Keychain Access User Guide.
- For Linux, see Installing a root CA certificate in the trust store in the Ubuntu documentation. Similar instructions should apply to most Linux distributions.

If you have installed a certificate but Copilot isn't detecting it, see "Troubleshooting network errors for GitHub Copilot."