Hey! It seems like you’ve run into an issue where your GitHub repository is public, but the visibility in Azure is set to private, and you're trying to change the settings.

Here’s how you can resolve this:

**Check GitHub Repository Visibility:**

* 1. Go to your GitHub repository.
  2. On the repository’s page, click on the "Settings" tab (usually located on the top right of the repo page).
  3. Scroll down to the "Danger Zone" section and check the repository’s visibility.
  4. If it's **public** and you want it **private**, click on "Change repository visibility" and set it to private.
  5. If you want to change it **from private to public**, follow the same steps.

**Check Azure Repo Visibility:**

* 1. In your Azure project or DevOps, make sure you're accessing the correct repository settings.
  2. Go to **Project Settings** in Azure DevOps.
  3. Look for the **Repository settings** or **Security settings** (depending on where the visibility is set).
  4. Adjust the visibility here to match your GitHub settings (either public or private).

**Sync Visibility Between GitHub and Azure:**

* 1. If you are using Azure DevOps pipelines or GitHub Actions, make sure you have linked the repositories correctly in the settings.
  2. If you linked a **private repository in GitHub** to **Azure**, Azure may require additional permissions to access it. You may need to adjust your connection settings or access tokens.

**Error Message:**

* 1. If the error is asking you to change visibility in project settings, make sure you’re in the correct section in **Azure DevOps** or **Azure portal** and adjust the permissions accordingly.

If this doesn’t resolve your issue, could you provide more details about the exact error message you're seeing? That would help pinpoint the problem more accurately.

##### You said:

[error]No hosted parallelism has been purchased or granted. To request a free parallelism grant, please fill out the following form https://aka.ms/azpipelines-parallelism-request

###### ChatGPT said:

The error message you're seeing indicates that your Azure DevOps organization doesn't have parallelism enabled for running multiple builds or pipelines simultaneously. This is a common issue when you're using Azure Pipelines and don't have the required resources or parallelism grants.

Here’s how you can resolve the issue:

### Steps to Request Free Parallelism Grant:

**Fill Out the Request Form:**

* 1. Go to the link provided in the error message: [https://aka.ms/azpipelines-parallelism-request](https://aka.ms/azpipelines-parallelism-request" \t "C:\\Users\\User\\AppData\\Local\\Temp\\_new).
  2. This form allows you to request a free parallelism grant for your Azure DevOps organization.

**Request Parallelism for Your Organization:**

* 1. When filling out the form, make sure you provide accurate information about your organization and its usage of Azure DevOps Pipelines.
  2. The Azure team will review your request, and if eligible, they will grant free parallelism resources.

**Review and Adjust Your Pipeline Settings:**

* 1. After the parallelism grant is approved, you should be able to run multiple builds and pipelines concurrently without hitting the limit.
  2. If the request is denied or you need more parallel jobs, you may need to consider upgrading to a paid plan for additional parallelism resources.

### Additional Options:

* **Use Hosted Agents:** If you're using the hosted agents (default Microsoft-hosted agents), this issue might occur if you exceed the free tier limits. Azure provides **1 free parallel job** for public repositories, and **1 free parallel job** for private repositories with limits.
* **Self-hosted Agents:** If you need more parallelism or have consistent usage, consider setting up **self-hosted agents** on your infrastructure, which can help avoid these limitations.

Let me know if this helps or if you need more details on any of the steps!

##### You said:

what is parallelism .plai in thunglish and how to solve.please guide me step by step

###### ChatGPT said:

Sure! Let's break it down in simple terms.

### What is "Parallelism" in Azure Pipelines?

**Parallelism** in Azure Pipelines means the ability to run multiple jobs or tasks (like builds, tests, or deployments) **at the same time**. Think of it as running multiple workers simultaneously to get your job done faster, instead of waiting for one task to finish before starting the next.

In Azure DevOps, when you run builds or pipelines, you might hit a limit on how many of those tasks can run at the same time, especially if you're using the free version. This limit is called **parallelism**.

For example, if you have 3 builds running and only 1 parallel job is available, the next two builds will have to wait for the first one to finish. If you need more builds to run at once, you need more parallel jobs, and that's where the error comes from.

### The Error Explained

The error you're seeing: **"No hosted parallelism has been purchased or granted."**

This means that your Azure DevOps account doesn't have enough parallel jobs allowed to run multiple tasks at once. You either need to request more parallel jobs (if you're eligible for a free grant) or purchase more parallelism.

### How to Solve the Issue Step-by-Step

#### Step 1: Check How Many Parallel Jobs You Have

**Go to Azure DevOps Portal**:

* + Open [Azure DevOps](https://dev.azure.com/" \t "C:\\Users\\User\\AppData\\Local\\Temp\\_new).
  + Log in to your account and go to your organization.

**Check your parallel job limits**:

* + Go to **Project Settings** (bottom left corner).
  + Click on **Pipelines** → **Parallel Jobs**.
  + Here, you’ll see how many parallel jobs you have available and if you’ve hit the limit.

#### Step 2: Request More Parallel Jobs (Free Grant)

If you don’t have enough parallel jobs, you can request more by filling out a form for a **free parallelism grant**.

**Go to the Request Form**:  
Click on this link: [Request Parallelism](https://aka.ms/azpipelines-parallelism-request" \t "C:\\Users\\User\\AppData\\Local\\Temp\\_new).

**Fill in the Form**:

* + **Organization Name**: Put your Azure DevOps organization name.
  + **Number of Parallel Jobs**: Mention how many parallel jobs you need.
  + **Usage Details**: If the form asks for usage details, briefly explain how many pipelines you're running and why you need more parallel jobs.

**Submit the Form**:

* + Once you submit the form, the Azure team will review your request.
  + If approved, they will give you more parallelism resources, and you should be able to run multiple jobs at the same time.

#### Step 3: Wait for Approval

* The approval usually happens quickly (depending on your usage).
* Once your request is granted, you'll be able to run more tasks at the same time in Azure Pipelines.

#### Step 4: Set Up Self-Hosted Agents (Optional)

If you need more parallelism right away or if you have a larger workload, you can **set up self-hosted agents** (your own servers or virtual machines) to handle the jobs.

1. **Go to Azure DevOps** → **Project Settings**.
2. Under **Pipelines**, click on **Agent Pools**.
3. Create a new **Agent Pool** and follow the instructions to install agents on your own machine.

By using self-hosted agents, you can run your jobs faster and don’t rely on the hosted Azure agents, which might have limits.

#### Step 5: Upgrade to a Paid Plan (Optional)

If you're okay with spending money, you can also upgrade to a **paid Azure DevOps plan**. Paid plans give you more parallel jobs and additional resources.

1. **Go to Azure DevOps Pricing Page**.
2. Choose the plan that suits your needs and follow the instructions to upgrade.

### In Summary:

1. **Check your parallel job limit** in Azure DevOps.
2. **Request free parallelism** through the link provided in the error message.
3. **Wait for approval** or set up **self-hosted agents** if you need more jobs immediately.
4. You can also **upgrade to a paid plan** for more parallelism.

I hope that helps! Let me know if you need more details on any step. 😊

##### You said:

what is parallelism .please tell in thunglish and how to solve.please guide me step by step

###### ChatGPT said:

Sure! Let me explain "parallelism" in simple **Thunglish** (a mix of Tamil and English) so it's easier for you to understand:

### What is Parallelism in Azure Pipelines?

**Parallelism** in Azure Pipelines means **running multiple tasks at the same time** (simultaneously) instead of waiting for one task to finish before starting another.

Example:

* Imagine you are making **three cups of tea**. If you make them one by one, it will take time (sequential).
* But if you make all three cups at the same time (parallel), it will be much faster!

Similarly, in Azure DevOps, if you have multiple **builds** or **deployments**, you can run them at the same time to save time, and this is called **parallelism**.

### Problem: "No hosted parallelism has been purchased or granted."

This error is saying that **your account doesn’t have enough parallel jobs** to run multiple tasks at once. You might be on the free plan, which only gives you **one parallel job**. If you need more jobs (to run multiple tasks at once), you have to either **request more free parallelism** or **buy a paid plan**.

### Step-by-Step Solution:

#### Step 1: Check Your Parallel Jobs Limit

1. **Go to Azure DevOps** (open [https://dev.azure.com](https://dev.azure.com" \t "C:\\Users\\User\\AppData\\Local\\Temp\\_new)).
2. **Login to your account** and select your organization.
3. **Go to Project Settings** (bottom left corner).
4. Under **Pipelines**, select **Parallel Jobs**.
   * Here, you will see how many parallel jobs you currently have.
   * If you see "1", that means only one task can run at a time.

#### Step 2: Request Free Parallelism

Azure gives free parallel jobs for open-source projects or on special requests. If you don’t have enough, you can ask for more.

**Click on this link to request more parallelism**: [Request Parallelism](https://aka.ms/azpipelines-parallelism-request" \t "C:\\Users\\User\\AppData\\Local\\Temp\\_new).

**Fill the form**:

* + Put your **organization name** and details about why you need more parallel jobs.
  + Submit the form.

**Wait for approval**:

* + After submission, Azure team will check your request. If approved, they will give you more parallel jobs.

#### Step 3: Self-Hosted Agents (Optional)

If you want **more jobs immediately**, you can set up your **own machine** to run tasks. This way, you don’t have to depend on Azure's free jobs.

1. **Go to Project Settings** → **Agent Pools**.
2. **Create a new Agent Pool** and follow the steps to install an agent on your own machine or server.
3. Once done, you can use this machine to run parallel jobs without waiting for Azure's hosted agents.

#### Step 4: Upgrade to Paid Plan (Optional)

If you need even more parallel jobs and faster builds, you can **upgrade to a paid plan**.

1. Go to Azure DevOps pricing page and select a plan that gives you more parallelism.

### Simple Summary:

1. **Parallelism** means running tasks at the same time, like making 3 cups of tea together.
2. If you get the error about parallelism, it means you don't have enough jobs to run tasks at once.
3. **Request more free parallel jobs** via the link.
4. Optionally, set up **self-hosted agents** or **upgrade to a paid plan**.

Hope this helps! Let me know if anything is unclear or if you need more help. 😊

##### You said:

step 3 need to pay

###### ChatGPT said:

Yes, you’re correct! In **Step 3**, if you want **more parallelism** immediately and you don't want to wait for the free grant, you'll need to **pay** for more parallel jobs.

Here's a more detailed breakdown for you:

### Option 1: ****Request More Free Parallelism**** (without paying)

* Azure DevOps gives **one free parallel job** for private repositories and unlimited parallel jobs for **public repositories** (on the free plan).
* If you need more jobs **for free**, you can **request additional free parallelism** (based on your usage).

**How to request free parallelism**:

1. Go to this link: [Request Parallelism](https://aka.ms/azpipelines-parallelism-request" \t "C:\\Users\\User\\AppData\\Local\\Temp\\_new).
2. Fill out the form with your organization’s details.
3. Wait for Azure to approve your request.

If your request gets approved, they will grant you more parallel jobs for free.

### Option 2: ****Pay for More Parallel Jobs**** (Paid)

If you need **more parallel jobs immediately** or if you’re doing a **lot of builds** and need them to run faster, **you’ll need to upgrade to a paid plan**. Azure charges for additional parallel jobs beyond the free limit.

**Steps to Upgrade and Pay for More Parallelism**:

1. Go to the [Azure DevOps Pricing Page](https://azure.microsoft.com/en-us/pricing/details/devops/azure-devops-services/" \t "C:\\Users\\User\\AppData\\Local\\Temp\\_new).
2. Look for the section about **parallel jobs** and choose the plan that suits your needs.
   * Paid plans usually give you **additional parallel jobs** based on the tier you choose.
3. **Sign up** for a paid plan.
4. After upgrading, your Azure DevOps organization will get more parallel jobs to run multiple builds simultaneously.

### Option 3: ****Self-hosted Agents (Free or Paid)****

Another way to avoid paying extra for hosted parallel jobs is to set up **self-hosted agents** on your own machine or server.

1. **Go to Azure DevOps → Project Settings → Agent Pools.**
2. Follow the instructions to set up your own machine as an agent.
3. You can run builds on your own machines, and this will give you more flexibility without relying on Azure’s hosted agents.

Setting up **self-hosted agents** is free but will require you to manage your own infrastructure (i.e., your own machines or virtual machines).

### In Summary:

* If you want **more parallel jobs without paying**, request a **free parallelism grant**.
* If you need **instant access to more parallel jobs**, you'll need to **pay for a higher plan**.
* **Self-hosted agents** are another **free option**, but you need your own machines.

Let me know if you need help with any of these steps! 😊

##### You said:

So i used this link to request free parallelism https://aka.ms/azpipelines-parallelism-request.However i searched in googled there is another option creating agent pools . Actually i am not awareon that/pharaphase

###### ChatGPT said:

Ah, I see! You’re saying that you used the link to request free parallelism, but you also came across the option of creating **agent pools** and aren't sure about that option. Let me explain both options to help you understand the difference:

### 1. ****Requesting Free Parallelism (via the link)****:

* **What it is**: This is where you fill out a form to request **more parallel jobs** from Azure DevOps. This can be a free grant based on your usage or the type of project you're working on.
* **Why use it**: You request this if you need **more parallel jobs** to run multiple builds or tasks at the same time, and you don't want to wait for a long time or set up your own infrastructure.
* **How it works**: You simply fill out the form and, if approved, Azure will grant you additional parallelism (free of charge).

### 2. ****Creating Agent Pools****:

* **What it is**: Agent pools are **groups of machines or agents** that you can set up to run your tasks (like builds, deployments, etc.). These can be **self-hosted agents**, meaning you manage them yourself (e.g., on your own server or machine). Alternatively, you can use **Azure’s hosted agents**, but self-hosted ones give you more control.
* **Why use it**: Setting up **agent pools** allows you to run tasks (builds, etc.) on your **own machines** instead of relying solely on Azure’s hosted agents. This can be particularly useful if you want more flexibility or want to avoid paying for hosted parallel jobs.
* **How it works**: You create an agent pool in Azure DevOps, and then you can link your own machines (or even virtual machines) to it. You can run multiple tasks at once using these self-hosted agents, without being limited by Azure’s hosted agents.

### In Summary:

**Requesting Free Parallelism** is for getting **more parallel jobs** from Azure directly, without needing to set up your own machines. This is easier and quicker, especially for those who need just a few more parallel jobs.

**Creating Agent Pools** is for setting up your **own infrastructure** (either your physical machines or VMs) to handle your builds and tasks. This option gives you more control, but you will need to manage the machines or virtual machines yourself.

Both options help you run tasks in parallel, but **requesting parallelism** is simpler, while **agent pools** require more effort and setup but provide more flexibility.

Let me know if this clears things up!

##### You said:

So i used this link to request free parallelism https://aka.ms/azpipelines-parallelism-request.However i searched in googled there is another option creating agent pools . Actually i am not awareon that//pharaphase this sentences

###### ChatGPT said:

Sure! Here’s a paraphrased version of your sentence:

"I used the link to request free parallelism ([https://aka.ms/azpipelines-parallelism-request](https://aka.ms/azpipelines-parallelism-request" \t "C:\\Users\\User\\AppData\\Local\\Temp\\_new)), but while searching on Google, I came across another option called creating agent pools. I'm not familiar with that option."

##### You said:

Creating Agent Pools you need own machine

###### ChatGPT said:

Yes, that's correct! When you create **Agent Pools** in Azure DevOps, you typically need your **own machine** or **virtual machine (VM)** to act as an agent. These self-hosted agents allow you to run your builds and tasks on your own infrastructure instead of relying on Azure's hosted agents.

Here's how it works:

1. **Self-hosted Agents**: You set up your own physical machine or virtual machine (VM) to run your tasks. This gives you more control over your environment.
2. **How to Use**: Once the machine is set up, you link it to your **Azure DevOps Agent Pool**. The tasks (like builds or deployments) will run on your own machine.

So, with **Agent Pools**, you manage your own infrastructure and handle the setup, but it gives you flexibility and can help you avoid running into limits on Azure's hosted agents.

Let me know if you'd like more details on how to set up your own agents! 😊

##### You said:

pwn machine is laptop?

###### ChatGPT said:

Yes, when I say "own machine," it can refer to your **laptop**, **desktop**, or any **physical server** or **virtual machine (VM)** that you own or control.

So, for **self-hosted agents** in Azure DevOps, if you have a laptop or desktop that you can dedicate to running your builds and tasks, you can set it up as an **agent**. Here's how it works:

1. **Own Machine**: This can be any physical device like your **laptop** or **PC** that is always on and accessible.
2. **Self-Hosted Agent**: You install the **Azure Pipelines agent** on that machine (laptop or desktop), and it will act as an agent that runs your builds or tasks.
3. **Configuration**: You'll link that machine to your **Azure DevOps Agent Pool**.

This way, your laptop can work as an Azure DevOps agent that helps with the processing of your builds.

Let me know if you need help setting it up!

##### You said:

guide me step by step now only i download the agen for windows in zip file

###### ChatGPT said:

Sure! Since you’ve downloaded the Azure DevOps agent as a zip file for Windows, here’s a step-by-step guide to setting up a **self-hosted agent** on your machine (laptop/PC):

### Step-by-Step Guide to Set Up the Azure DevOps Self-Hosted Agent:

#### Step 1: ****Extract the Zip File****

1. **Locate the ZIP file** you downloaded (usually in your **Downloads** folder).
2. **Right-click** on the ZIP file and select **Extract All**.
3. Choose a location (for example, C:\agent), and click **Extract**. This will unzip the agent files to that folder.

#### Step 2: ****Open Command Prompt as Administrator****

1. Press **Windows + X** and select **Command Prompt (Admin)** or **Windows PowerShell (Admin)** to open it as an administrator.
2. **Navigate to the folder** where you extracted the agent files. For example:

bash

Copy

cd C:\agent

#### Step 3: ****Configure the Agent****

In the Command Prompt, **run the configuration command** to set up the agent:

bash

Copy

.\config.cmd

You’ll be prompted for the following information:

**Azure DevOps Organization URL**:  
Enter the URL of your Azure DevOps organization. It will look like this:

plaintext

Copy

https://dev.azure.com/{your\_organization\_name}

For example: https://dev.azure.com/mycompany

**Personal Access Token (PAT)**:  
You need to provide a **Personal Access Token** (PAT) for authentication.

* + - To create a PAT, follow these steps:
      1. Go to your Azure DevOps Profile.
      2. Click **New Token** and provide the necessary details (like name, expiration, and scopes).
      3. Copy the PAT once created.
    - Paste the **PAT** in the command prompt when prompted.

**Agent Pool**:  
You'll be asked which **Agent Pool** to use. If you're setting up a new agent pool, just press **Enter** to use the default one, or specify a custom pool.

**Agent Name**:  
You can give your agent a name (or just press **Enter** to use the default name).

#### Step 4: ****Run the Agent****

After configuring the agent, you’ll be prompted to **run the agent**. In the command prompt, type:

bash

Copy

.\run.cmd

This will start the agent and make it ready to receive jobs from your Azure DevOps organization.

* + **If you want the agent to always run** (even after restarting the computer), you can install it as a **service** by running:

bash

Copy

.\svc.sh install

#### Step 5: ****Verify the Agent****

1. Once the agent is running, go back to **Azure DevOps**.
2. Navigate to **Project Settings** → **Agent Pools**.
3. You should now see the **self-hosted agent** listed in the **Agent Pool** you selected earlier.
4. You can test if the agent is working by triggering a build in your pipeline, and the job should run on your self-hosted agent.

### Optional: ****Set the Agent to Automatically Start****

If you want the agent to start automatically whenever your computer reboots, you can set it up as a service using the following command:

bash

Copy

.\svc.sh install

After this, it will start automatically whenever your machine is restarted.

### In Summary:

1. **Extract the agent zip file**.
2. **Open Command Prompt** and navigate to the folder.
3. **Run the** config.cmd to configure the agent with your Azure DevOps details.
4. **Run the** run.cmd to start the agent.
5. **Verify the agent** in the Azure DevOps **Agent Pools** section.

Let me know if you run into any issues along the way or need further assistance! 😊

##### You said:

asking authentication Pat

###### ChatGPT said:

Got it! When setting up the self-hosted agent, you need to authenticate with Azure DevOps using a **Personal Access Token (PAT)**. Here's how to create one and use it for authentication:

### Step 1: ****Create a Personal Access Token (PAT)****

**Go to Azure DevOps**:

* + Open your browser and go to your Azure DevOps portal: [https://dev.azure.com](https://dev.azure.com" \t "C:\\Users\\User\\AppData\\Local\\Temp\\_new).
  + Sign in with your Azure DevOps account.

**Create a PAT**:

* + In the top-right corner, click on your **profile picture** and select **Personal access tokens** from the dropdown.
  + Click on the **New Token** button.

**Fill in the PAT Details**:

* + **Name**: Give your token a name (e.g., "Self-hosted Agent Token").
  + **Organization**: Select the organization you want to connect the agent to.
  + **Scopes**: Choose the required scopes. For the agent, you'll need the following permissions:
    - **Agent Pools (Read & Manage)**.
    - Optionally, you can give it **Build (Read & Execute)** if you're using it for build pipelines.
  + **Expiration**: You can set an expiration date (e.g., 30 days, 90 days).
  + **Create**: After filling out the details, click **Create**.

**Copy the PAT**:

* + Once the PAT is created, **copy it** immediately. This is the only time you’ll be able to view it.

### Step 2: ****Use the PAT to Authenticate the Agent****

**Go Back to the Command Prompt**:

* + After you've extracted and configured the agent, when you run the command .\config.cmd, you will be prompted to enter the **Personal Access Token (PAT)**.

**Enter the PAT**:

* + When asked for the PAT in the command prompt, **paste the token** you copied earlier.

Example:

plaintext

Copy

Enter your Personal Access Token: <Paste your PAT here>

**Complete the Setup**:

* + After entering the PAT, the agent should be successfully authenticated and ready to use.

### Step 3: ****Verify the Agent****

1. Once the agent is configured and running, go back to **Azure DevOps**.
2. Go to **Project Settings** → **Agent Pools**.
3. Your self-hosted agent should appear in the **Agent Pool** you selected.

### Quick Notes:

* **PAT Permissions**: Ensure that the PAT has **Agent Pools (Read & Manage)** permission so it can connect to the agent pool and run tasks.
* **Security**: Keep your PAT **secure** and avoid sharing it. If you think it’s compromised, regenerate the token.

Let me know if you need further help with this process! 😊

##### You said:

Enter authentication type (press enter for PAT) >

###### ChatGPT said:

When you see the prompt:

arduino

Copy

Enter authentication type (press enter for PAT) >

This is asking you to choose how you want to authenticate the agent.

Since you want to use a **Personal Access Token (PAT)**, **just press Enter**. The system will then ask you to input the **PAT** you created.

### Here's what to do:

**Press Enter** (without typing anything) when you see the prompt.

You will then be asked to **Enter your Personal Access Token**.

* + **Paste** the PAT you copied earlier from Azure DevOps.

Example:

plaintext

Copy

Enter your Personal Access Token: <Paste your PAT here>

After entering the PAT, it will authenticate your agent and continue the setup process.

Once you've done this, your agent will be connected to Azure DevOps and ready to run builds or tasks.

Let me know if you need any more help! 😊

##### You said:

Enter personal access token > //i pasted the token but it shows failed

###### ChatGPT said:

If you're pasting the **Personal Access Token (PAT)** and it's showing as **failed**, there are a few potential reasons and solutions for this. Let's go through them step by step to troubleshoot:

### 1. ****Check PAT Permissions****

Make sure that the PAT you created has the correct **permissions**. For setting up a self-hosted agent, the **PAT** should have the following permissions:

* **Agent Pools**: Read & Manage
* **Build**: (Optional but recommended) Read & Execute

If your PAT doesn't have these permissions, the authentication will fail. To fix this, you can either **create a new PAT with the correct permissions** or **edit** the existing one.

### How to Create a New PAT with Correct Permissions:

1. Go to [Azure DevOps](https://dev.azure.com" \t "C:\\Users\\User\\AppData\\Local\\Temp\\_new) and click on your profile picture in the top-right corner.
2. Select **Personal access tokens** from the dropdown menu.
3. Click on **New Token**.
4. Set the following permissions:
   * **Agent Pools (Read & Manage)**: Ensure this is selected.
   * **Build (Read & Execute)**: If you want the agent to run builds, select this as well.
5. Set the expiration (e.g., 30 days) and then click **Create**.
6. **Copy the PAT** and try again.

### 2. ****Ensure Correct Copying of the Token****

When you copy the PAT, make sure there are **no extra spaces or hidden characters**. Sometimes copying a token can accidentally add a space at the beginning or end, which could cause authentication to fail.

* **Re-copy** the PAT from Azure DevOps and paste it into the command prompt exactly as it appears.

### 3. ****Check the Organization URL****

Make sure you're entering the correct **Azure DevOps organization URL** when prompted during the setup. It should be in the format:

arduino

Copy

https://dev.azure.com/{your\_organization\_name}

For example: https://dev.azure.com/mycompany

If the organization URL is wrong, the PAT won't work.

### 4. ****Double-check Agent Configuration****

If the agent was previously configured and failed, sometimes an incorrect configuration can cause issues. Try re-running the agent configuration process:

1. **Delete the existing agent folder** (the one where you extracted the zip file).
2. **Re-download and extract the agent** again from Azure DevOps.
3. Follow the configuration process from the start, ensuring you're using the correct organization URL and PAT.

### 5. ****Make Sure the Token Hasn't Expired****

If the **PAT** has expired, the agent will fail to authenticate. Check the **expiration date** of the PAT you created.

### 6. ****Try Using a New Command Prompt****

Sometimes, the terminal can have issues with recognizing pasted content. **Close the current command prompt**, open a new **Administrator Command Prompt**, and go through the configuration process again.

### If the Issue Still Persists:

* **Error message**: If the failure message provides more details (e.g., "Invalid Token" or "Unauthorized"), please share that message so I can assist further.

Try these troubleshooting steps, and let me know if the issue persists or if you encounter any error messages. I'm happy to help further! 😊

##### You said:

> Register Agent: Enter agent pool (press enter for default) > Enter agent name (press enter for DNEZ) > Scanning for tool capabilities. Connecting to the server. Successfully added the agent Testing agent connection. The SSL connection could not be established, see inner exception. PS C:\Users\User\Desktop\Netsynergy\agent> .\run.cmd >> Error reported in diagnostic logs. Please examine the log for more details. - C:\Users\User\Desktop\Netsynergy\agent\\_diag\Agent\_20250206-090557-utc.log An error occurred: Not configured PS C:\Users\User\Desktop\Netsynergy\agent>

###### ChatGPT said:

It looks like you're encountering two main issues:

**SSL Connection Issue**:

* + The error **"The SSL connection could not be established"** indicates that there is an issue with your connection to Azure DevOps. This could be caused by a number of factors, such as network issues or missing certificates.

**Not Configured Error**:

* + The error **"Not configured"** suggests that the agent may not be properly configured, even though it was added successfully. The issue likely lies in the configuration or connection settings.

Let's go step by step to troubleshoot both issues.

### 1. ****SSL Connection Issue****

The SSL connection failure can be due to several reasons:

#### Check Network Connectivity:

* Make sure your machine has internet access and can reach the Azure DevOps server. You can test this by trying to access the Azure DevOps URL directly in a browser:

plaintext

Copy

https://dev.azure.com/{your\_organization\_name}

* If you can't access the URL, there might be network/firewall issues blocking the connection.

#### Ensure Correct Time & Date on the Machine:

* Sometimes SSL errors occur if the system time is not synchronized. Check that your **system date and time** are correct. If the system time is incorrect, it can cause SSL certificate validation to fail.

#### Check Proxy or Firewall Settings:

* If you are behind a **proxy** or **firewall**, ensure that it is not blocking the connection to Azure DevOps. Sometimes, corporate networks use proxies, which can cause SSL issues.
  + You might need to configure the agent to use the proxy settings. To configure the agent to use a proxy, follow [this documentation](https://learn.microsoft.com/en-us/azure/devops/pipelines/agents/proxies" \t "C:\\Users\\User\\AppData\\Local\\Temp\\_new).

#### Check for Missing SSL Certificates:

* In some cases, your machine may be missing the required SSL certificates. Make sure your system is up to date with the latest **Windows updates**. You can also manually update your root certificates if needed.

### 2. ****"Not Configured" Error****

The "Not Configured" error indicates that the agent might not have been fully set up.

#### Check Configuration Files:

* Go to the agent directory (C:\Users\User\Desktop\Netsynergy\agent) and ensure that the configuration files are in place. Specifically, look for the following files:
  + config.json: Contains configuration details.
  + \_diag folder: Check the logs for more detailed error messages.

#### Reconfigure the Agent:

You might need to **reconfigure** the agent to ensure that it is fully set up. Here’s how you can do that:

**Delete Existing Agent Configuration**:

* + Delete the config.json file in the agent directory. This will remove any incomplete configurations.

**Reconfigure the Agent**:

* + Go back to your Command Prompt as Administrator and navigate to the agent directory:

powershell

Copy

cd C:\Users\User\Desktop\Netsynergy\agent

* + Run the config.cmd script again:

powershell

Copy

.\config.cmd

* + Follow the prompts to re-enter your **Azure DevOps URL**, **Personal Access Token (PAT)**, and **Agent Pool**.

**Verify**:

* + After reconfiguring, try running the agent again:

powershell

Copy

.\run.cmd

#### Check Diagnostic Logs:

* If the error persists, check the **diagnostic logs** to get more details:

plaintext

Copy

C:\Users\User\Desktop\Netsynergy\agent\\_diag\Agent\_20250206-090557-utc.log

* Open the log file to look for more specific error messages related to the SSL connection or configuration.

### 3. ****Additional Solutions****

#### Disable SSL Verification (as a temporary workaround):

If you're unable to fix the SSL issue, as a temporary measure, you can disable SSL verification for the agent (although this is **not recommended for production environments** due to security risks).

To disable SSL verification:

1. Open the config.cmd file in a text editor.
2. Add the following line:

bash

Copy

AZP\_DISABLE\_SSL\_VALIDATION=true

1. Save and re-run the agent configuration and start the agent.

This will bypass the SSL validation, but please keep in mind this is just a workaround and **shouldn't be used in production** as it may expose you to security risks.

### Conclusion:

* Check your **network connectivity** and ensure you're not behind a **proxy** or **firewall**.
* Make sure your **system time** is correct and there are no **missing SSL certificates**.
* Try **reconfiguring the agent** by deleting the old configuration and running the setup again.

Let me know if this resolves the issue or if you encounter any new error messages! I'm here to help. 😊