This document will show how to perform initial configuration of Azure DevOps and add a self-hosted agent to this service (agent can be understood as a virtual computer that will execute our code.)

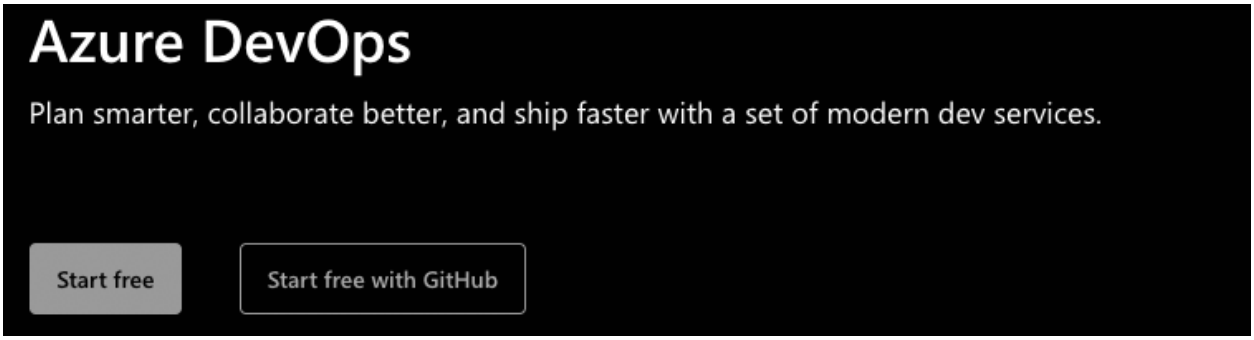
Prerequisite: git needs to be installed on the Linux VM before you start. You might have done it already before you started working on this runbook, or it might have been provided already preinstalled in your VM's image.
check if git is installed using the following command:

git --version

If it returned the current version, git is installed. If not run the following command to install git:

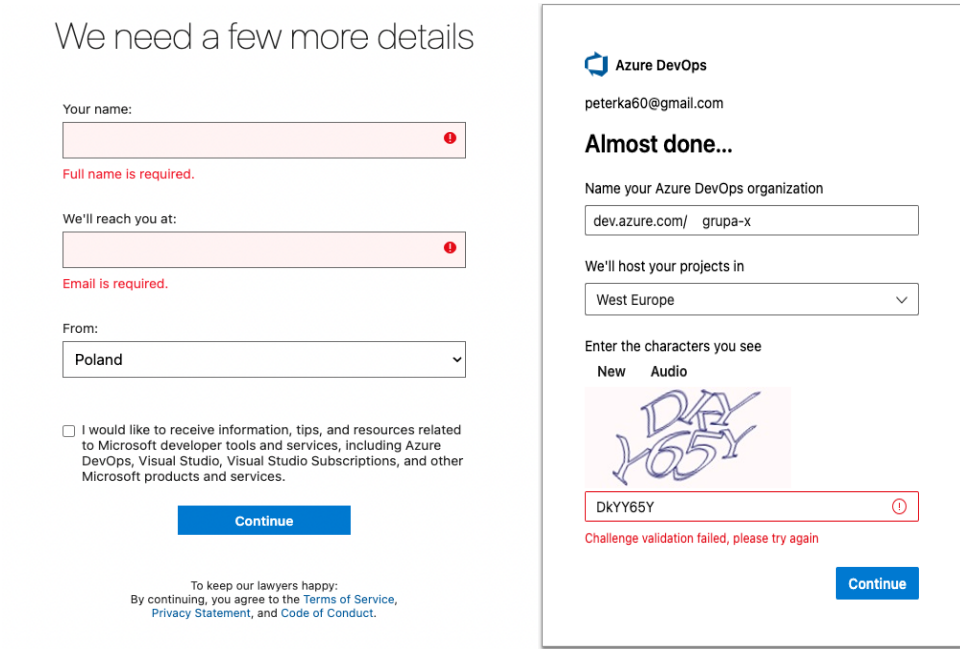
sudo apt install git

1. Go to https://azure.microsoft.com/en-us/products/devops/ and click "Start free with GitHub"

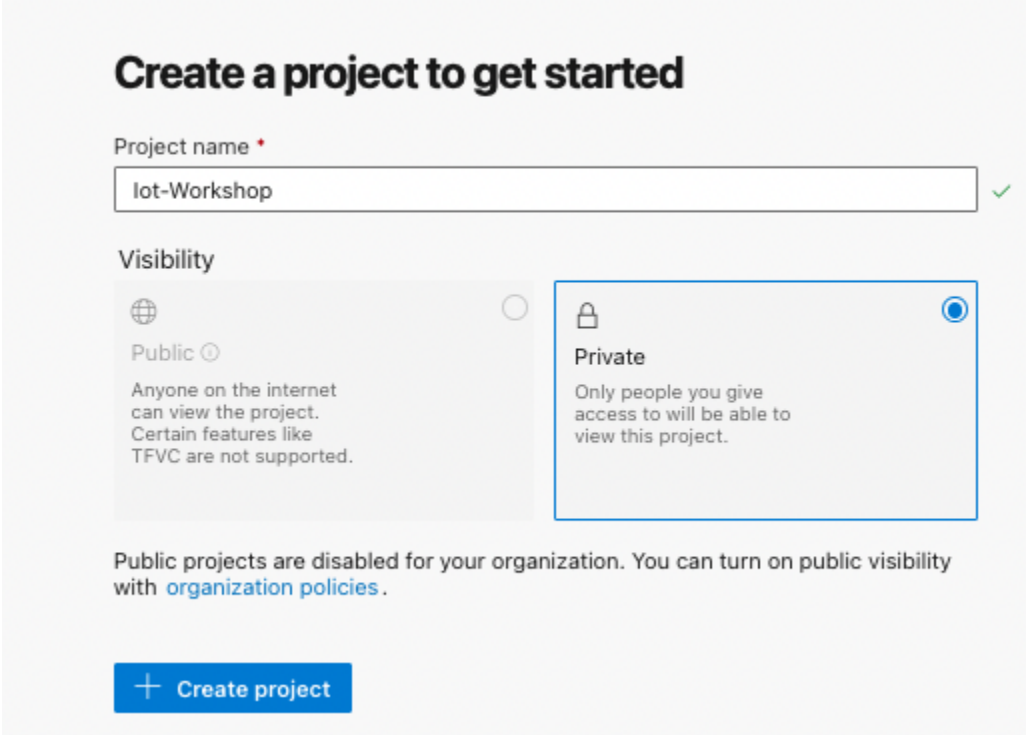


If you have any issues with logging in with a GitHub account:
- use this URL to sing out of all MS services- https://login.microsoftonline.com/logout.srf
- close all the browser windows/tabs
- reopen the browser and try again

2. Using the Github account created for this workshop log into Azure DevOps
There will be a few pages of details to fill out e.g.:
- If possible, Use your AWS username as the Name of your Azure DevOps organization.

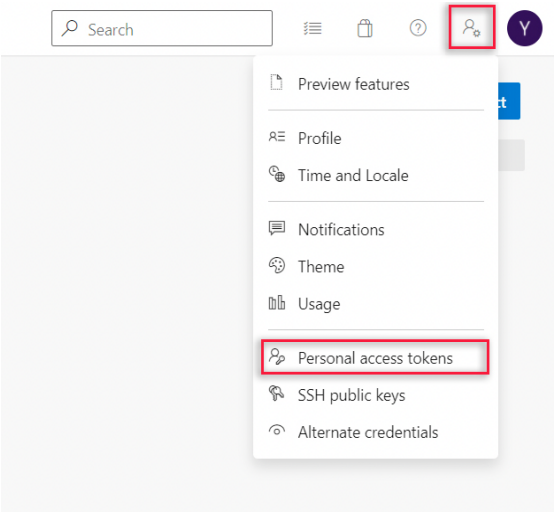


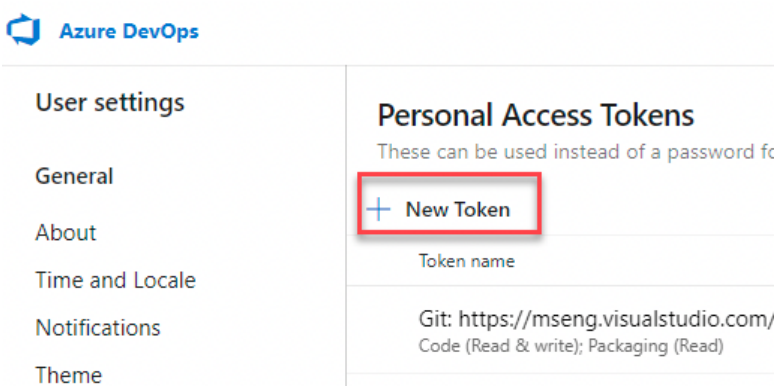
3. After you fill out all the details you will be taken to "Create a Project" page, do it.:
-Use "IoT-Workshop" for the Project Name

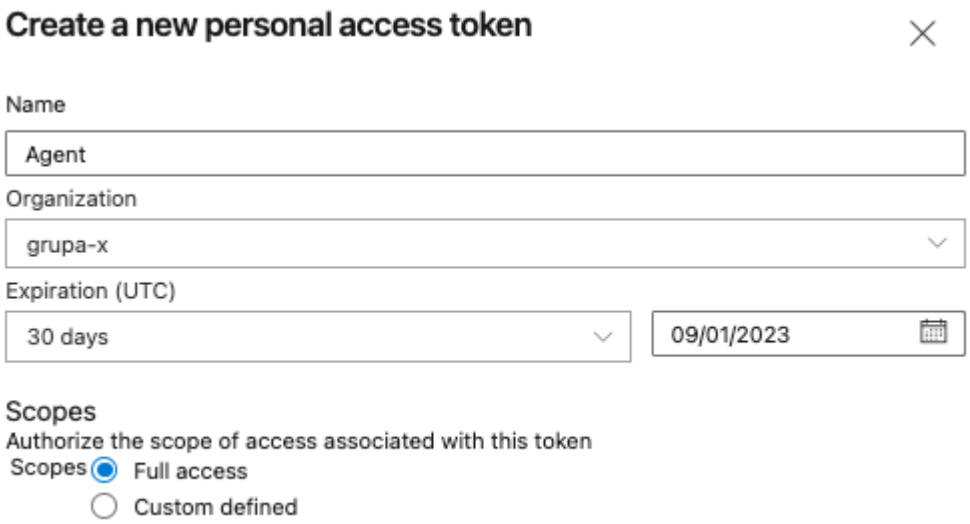


The basis of our CICD service is ready.

4. Now we have to add a self-hosted agent that will execute our code. First, we create a Personal Access Token - we will use it later to authenticate the agent in Azure DevOps
Create a token as indicated in the screenshots - remember to save it somewhere, if you lose it create a new one.

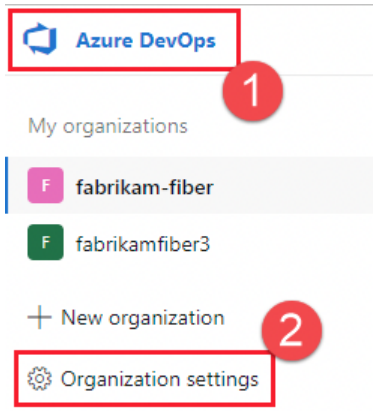


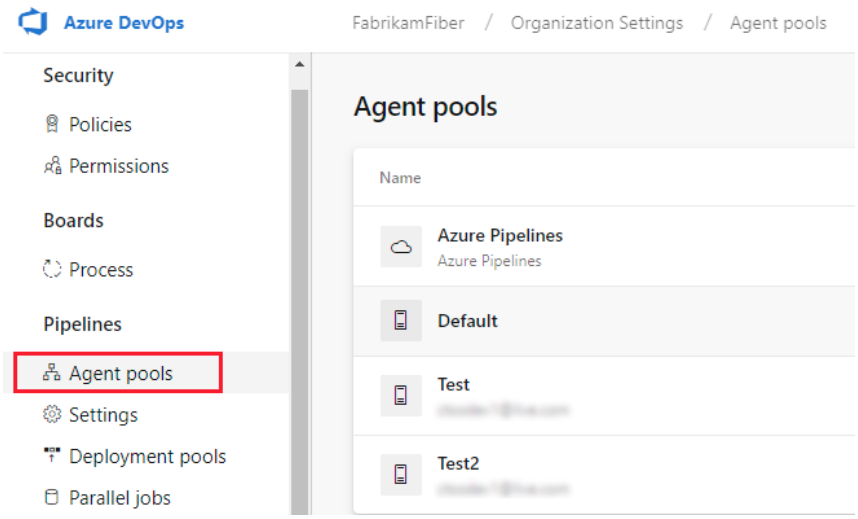




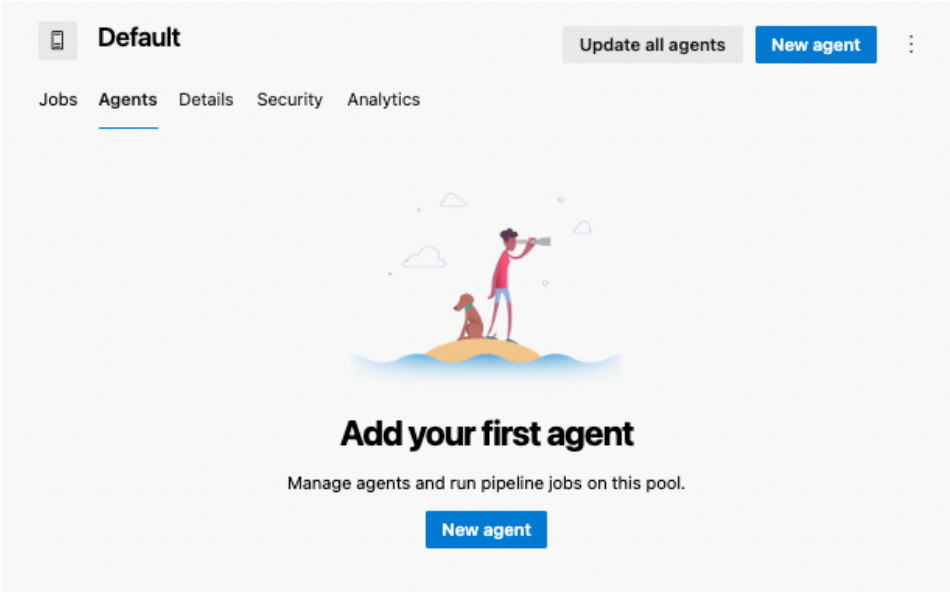
5. Now we add a new agent to the agent pool in Azure DevOps.
(We do it with an inbound connection to Azure DevOps from our virtual machine.)

- Using screenshots as a guide, go to the Agent pools in and choose the Default pool

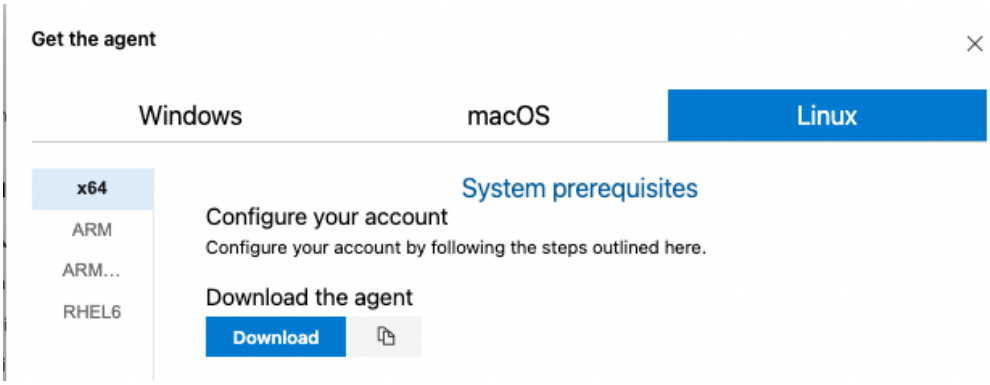




- Click New Agent



- Choose the “Linux” tab and click “Download”



- Open the terminal on the VM and run the following commands one by one, to prepare the agent for configuration:

mkdir myagent && cd myagent

tar zxvf ~/Downloads/vsts-agent-linux-x64-2.213.2.tar.gz

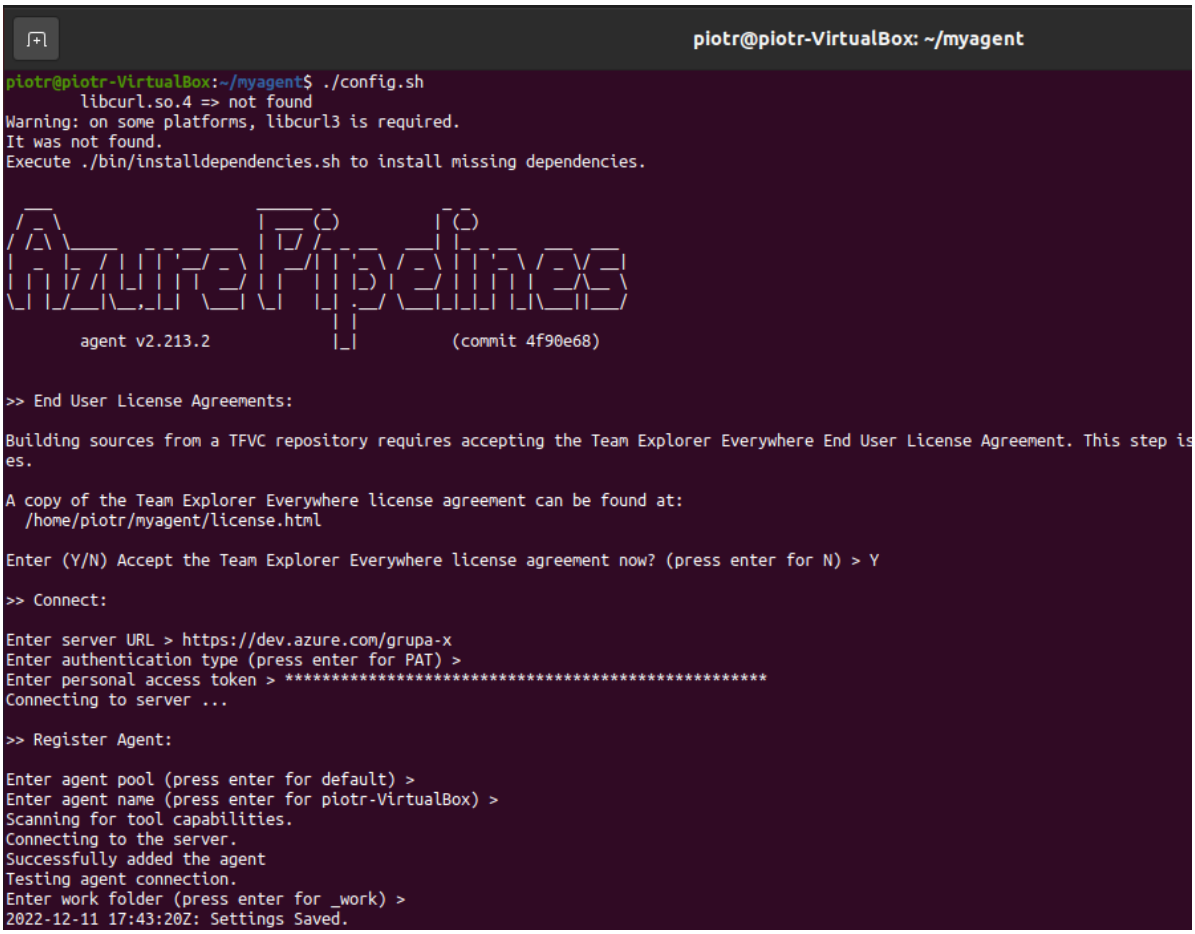
sudo ./bin/installdependencies.sh

- Run the following command in the terminal to begin the configuration:

./config.sh

There will invoke a script with an interactive prompt.
Refer to the block below for exact input and to the screenshot for the expected feedback.

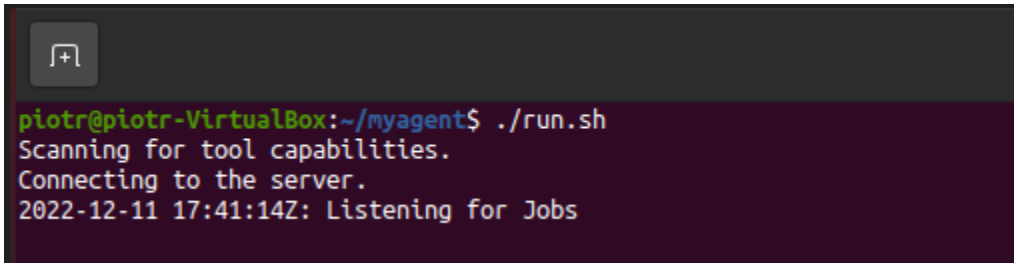
Consecutive input for the config script:
- *Y*
- *https://dev.azure.com/{name of the organization}* (for me it was https://dev.azure.com/grupa-x; look at your Azure DevOps URL if you have any doubts)
- *(enter)*
- *(the PAT that we created in step 3.)*
- *(enter)*
- *(enter)*
- *(enter)*



6. Start the agent by running the following command in the terminal

-Run the following command in the terminal:

./run.sh



This agent runs interactively (as long as the terminal window is not closed).

If you need to use the terminal open a new tab using the button in the top left corner.