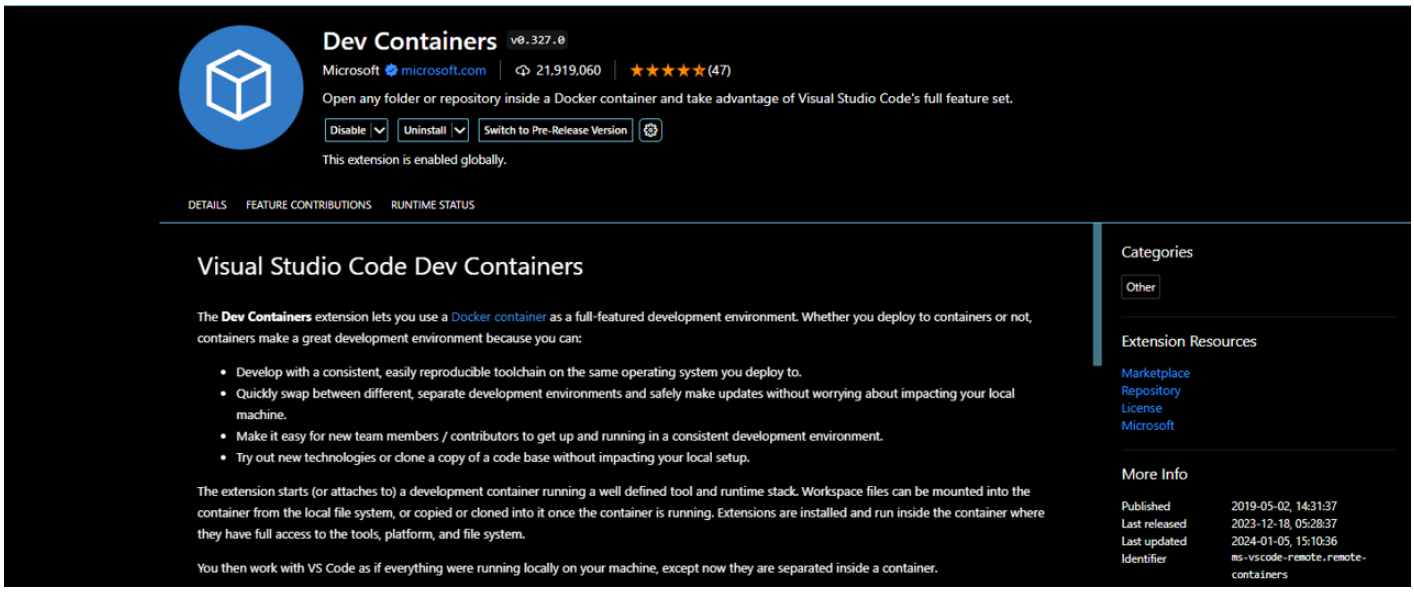
Getting Started with Skylab (DSO Factory)

Updated 10/09/2023



Overview

This document introduces the software and services that are most used in this course. The intent of this lab is to briefly introduce each of these tools and ensure the student can authenticate into each one. Students should have already received a username and password for each service that requires an account.

Estimated time: 20 minutes

Setup Needed:

- Introduction to Skylab/DSO Factory
- Introduction to DSO Factory Architecture
- Student account information provided

Learning Objectives

1. Introduction to each of the tools below
2. How each tool fits within DevSecOps architecture
3. Student can authenticate into each tool

Instructions

Jenkins

Jenkins is an open-source automation server for continuous integration and deployment of applications.

1. Use the following link to navigate to the login page: https://jenkins.dev.afsmtddso.com/
2. You should see the following screen where you can enter your login credentials

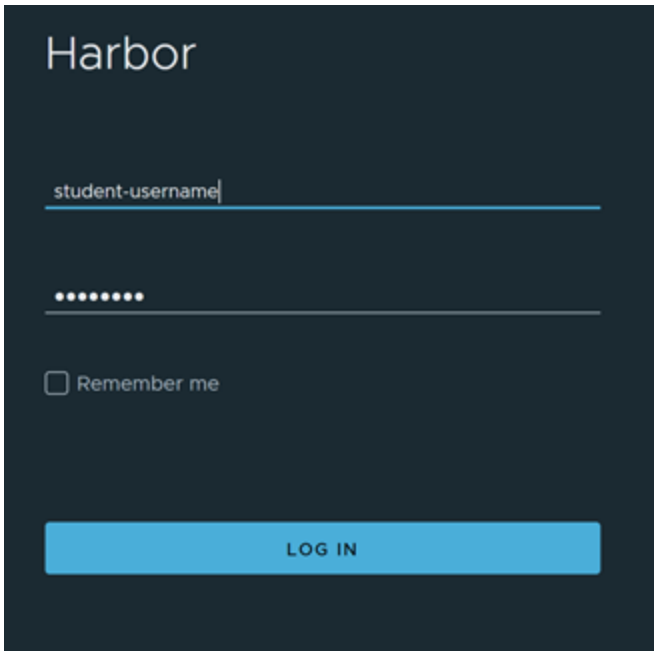
> **Note:** Use the Skylab/DSO Factory credentials emailed to you to login.

Harbor

Harbor is an open-source container image registry that secures the images with role-based access, scans them for vulnerabilities, and are signed as trusted images.

1. Use the following link to navigate to the login page: https://registry.dev.afsmtddso.com/
2. You should see the following screen where you can enter your login credentials

> **Note:** Use the Skylab/DSO Factory credentials emailed to you to login.

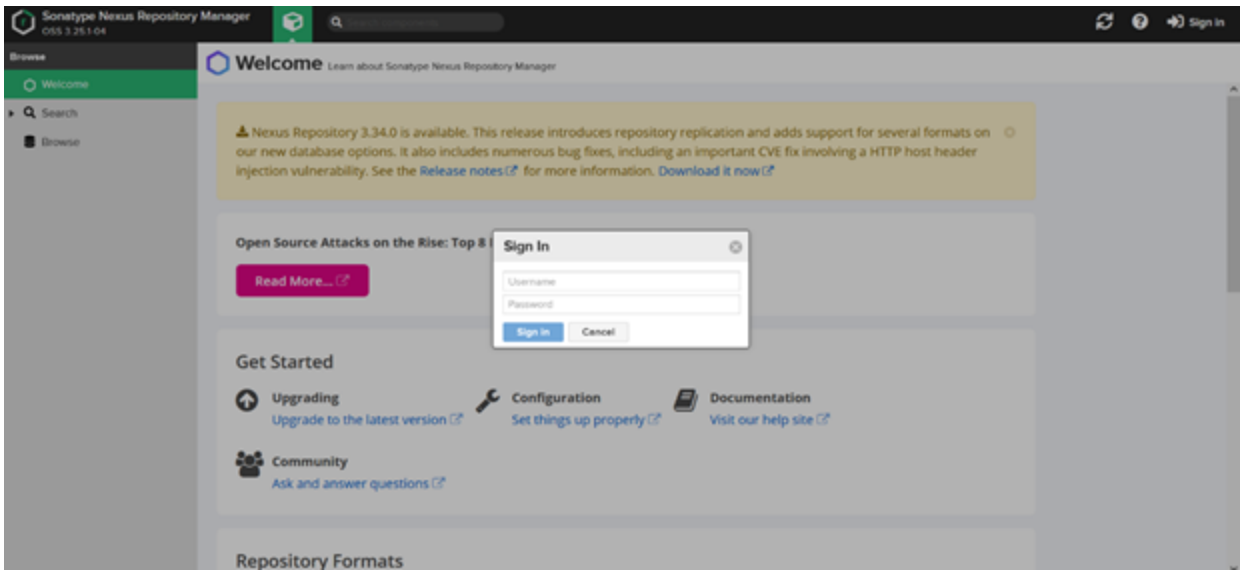


Nexus

Nexus is a repository manager used to store different formats of artifacts like; packaged code and binaries needed to support code deployments.

1. Use the following link to navigate to the login page: https://nexus.dev.afsmtddso.com/
2. Click “Sign In” in the upper right corner and you will be prompted to enter your login credentials

> **Note:** Use the Skylab/DSO Factory credentials emailed to you to login.



GitHub

You will need a GitHub account to use throughout this course. If you do not have one, you can create an account at https://github.com.

You will need a Personal Access Token (PAT) so you can log into your GitHub account from the command line. Please use the following article as a guide for making and using a PAT: Personal Access Token Tutorial Here

> **Note:** When you have to select permissions to give your PAT, select everything in the "repo" category (you can select more if you wish).

> **Tip:** Your PAT will only be *shown once*, so **paste it into the cheat sheet we provided for you**.

VSCode

You will use Visual Studio Code (VSCode) as your text editor while working in your remote EC2. In addition to installing VSCode, you'll be installing two VSCode extensions: Remote SSH and Dev Containers. Remote SSH will allow VSCode to login to your EC2 via SSH and edit files. Dev Containers is similar to Remote SSH but it is used on Docker containers you'll be creating in the Containerization lab.

1. Download and install VSCode from Software Center on your system: a. Press the Windows key and type "Software Center"

   b. Select Software Center app to launch it

   c. Search for "Code" to look for "Microsoft Visual Studio Code" application.

> **Note:** If you don't see it under "Applications", look under the "Installation Status" option in Software Center to see if it is already installed.

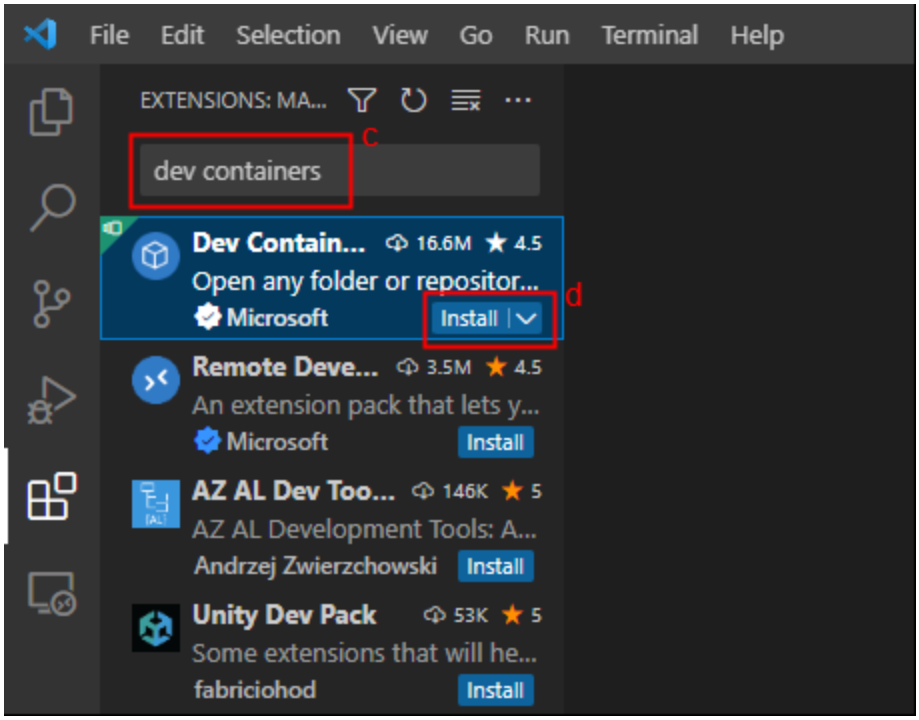
2. Install VSCode Dev Container Extension

   a. Open the VSCode editor

   b. On the left side bar, click on the "Extensions" icon

c. Type "dev containers" on the search bar

d. Install the "Dev Containers" extension for VSCode



3. Install VSCode Remote SSH

   a. Type "remote ssh" on the search bar

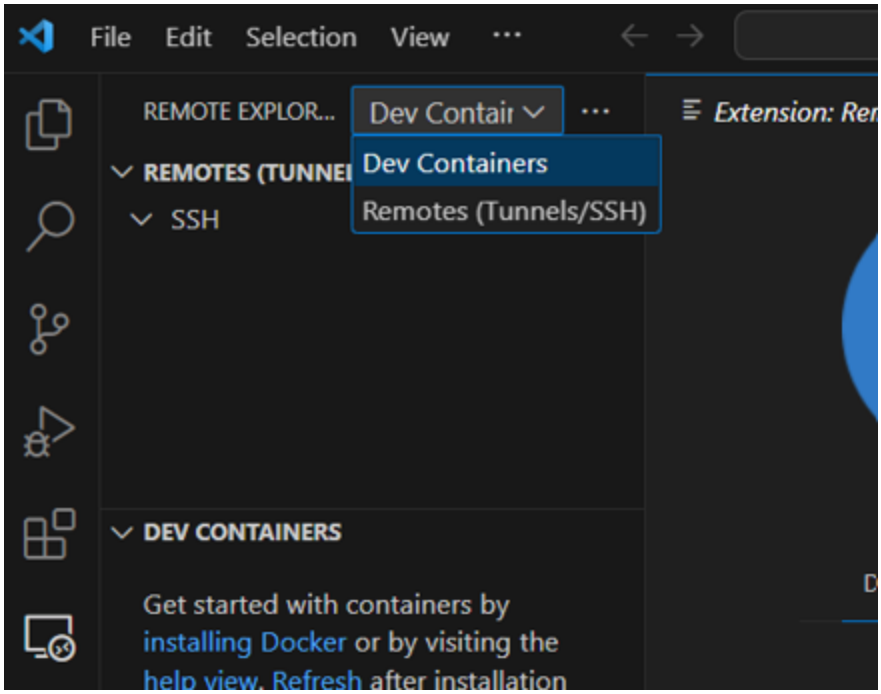
   b. Install the "Remote - SSH" extension for VSCode

> **Note:** The "Remote Explorer" icon will appear on the side bar after installing the remote extensions.

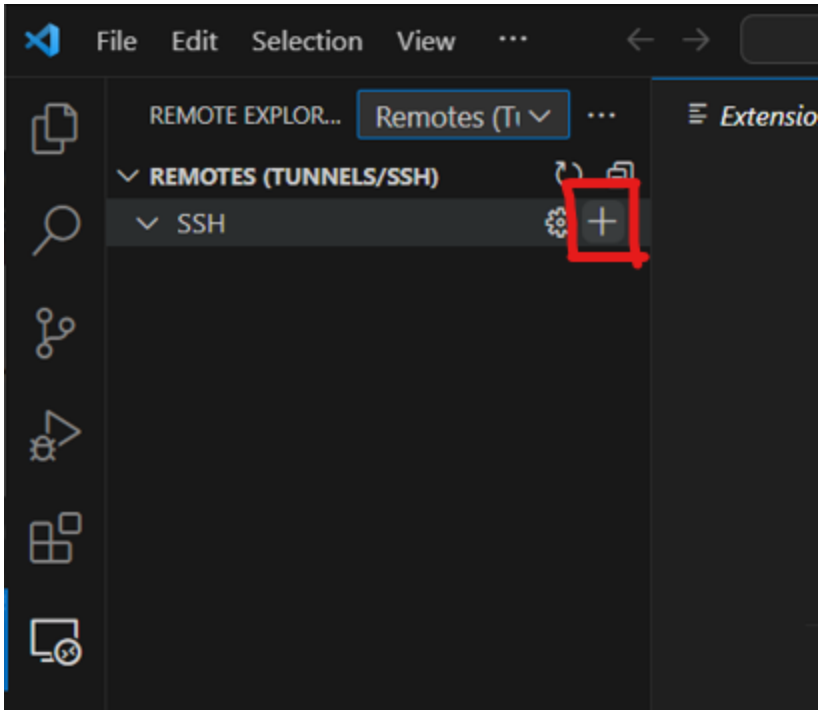
EC2

In this section you will connect to your EC2 with VSCode Remote SSH. You should've been given an IP address, as well as a username and password for it. You'll use these to access your EC2 for the duration of this course.

1. Open VSCode
2. Click on the "Remote Explorer" tab on the left column.
3. Select "Remotes (Tunnels/SSH)" from the dropdown menu



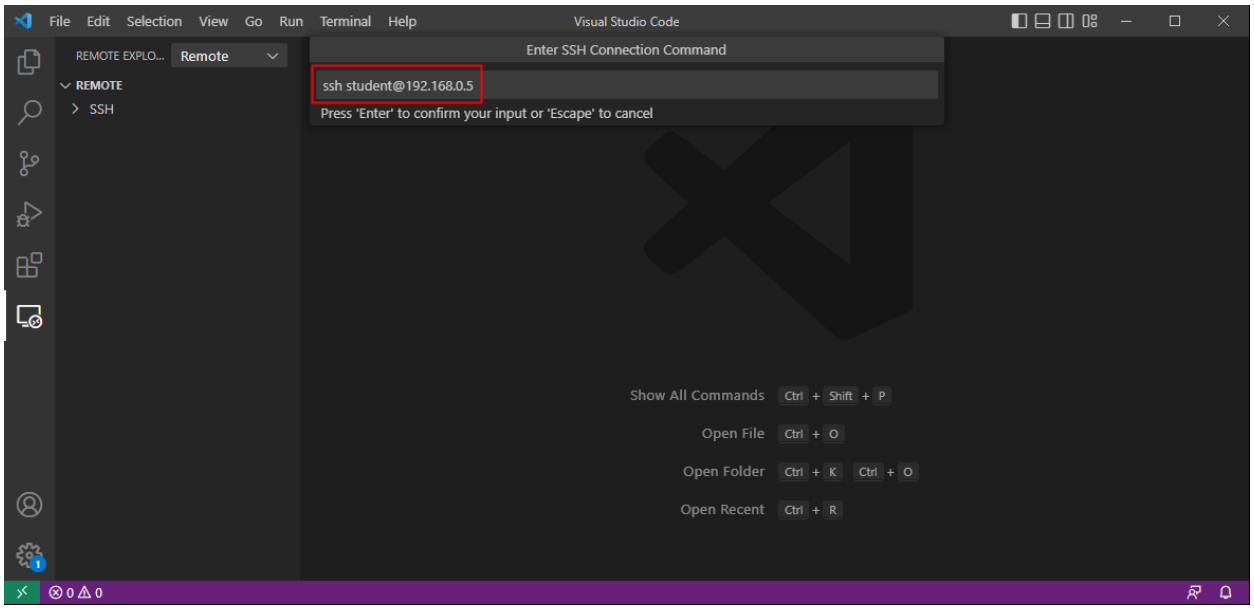
4. Add a new connection by clicking on the plus sign (you'll see this once you hover your cursor over the "SSH" line):



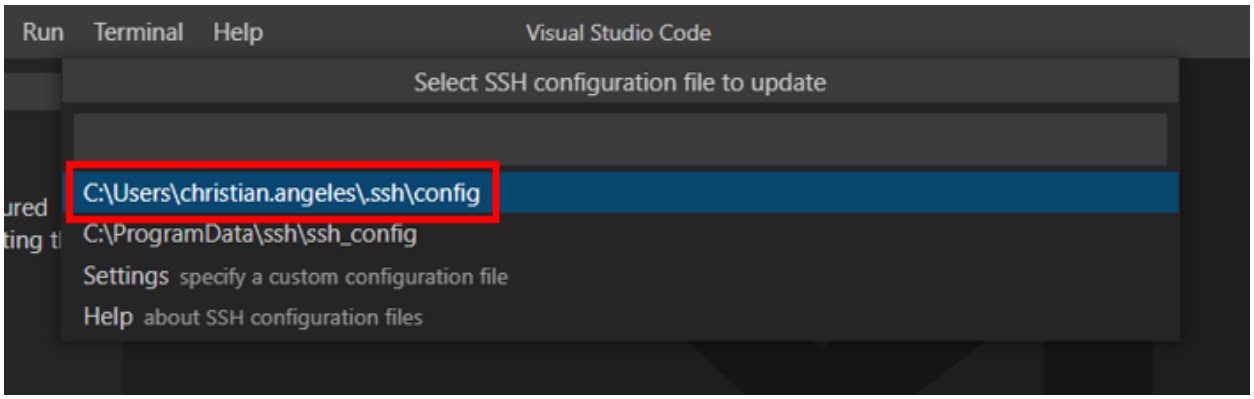
5. Login to your EC2 by entering your SSH command:

```
ssh <USERNAME>@<IP_ADDRESS_OF_EC2>
```

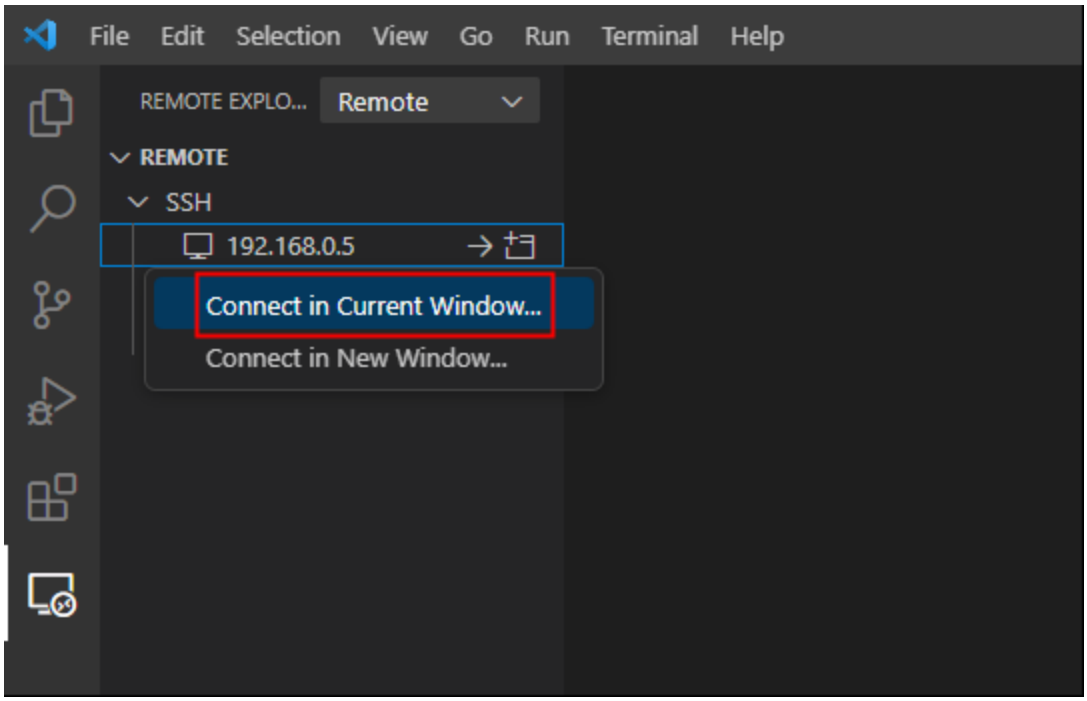
> **Note:** The ssh (Secure Shell) command has 3 parts: the ssh command, username, and host (IP/Domain name).



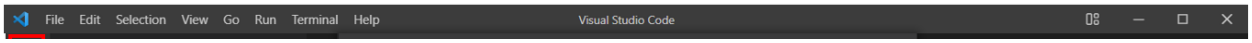
6. When prompted to select which configuration file to update, choose the first option.



7. Right-click on the IP address and connect to your EC2



8. Select "Linux" as the platform for the remote host when prompted
9. Select "Continue" and enter your password when prompted
10. Open your home folder once logged in to your EC2
    a. On the side bar, click on the "Explorer" icon
    b. Click on "Open Folder"
    c. Enter the path `home/student/` to open



> **Note:** If you get a pop-up window that asks "Do you trust the authors of the files in this folder?," click "Trust folder and continue."

d. Enter your password when prompted

> **Tip:** If your terminal disappears, re-open it by right-clicking on the left pane of the "Explorer" tab and selecting "Open in Integrated Terminal"