# Onboarding to the COM AWS Team

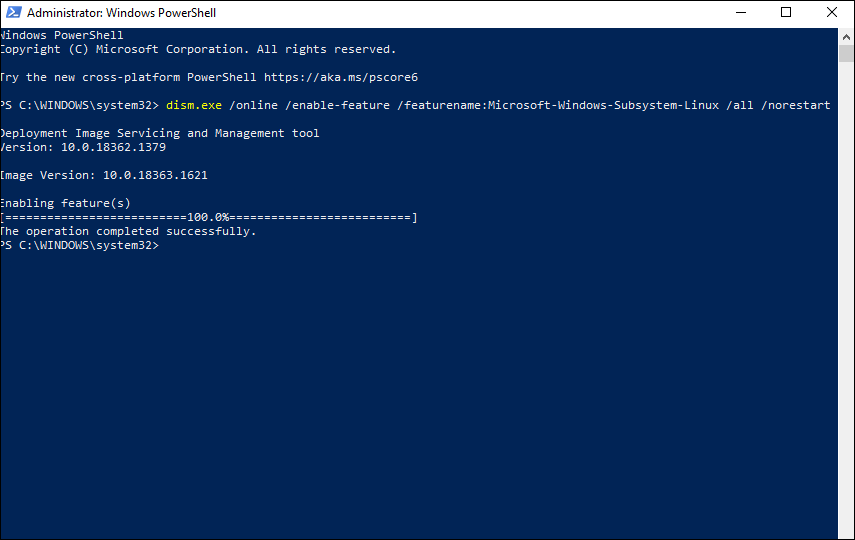
# WSL

**Step 1 - Enable the Windows Subsystem for Linux**

You must first enable the "Windows Subsystem for Linux" optional feature before installing any Linux distributions on Windows.

Open PowerShell as Administrator and run:

dism.exe /online /enable-feature /featurename:Microsoft-Windows-Subsystem-Linux /all /norestart



## Step 2 - Check requirements for running WSL 2

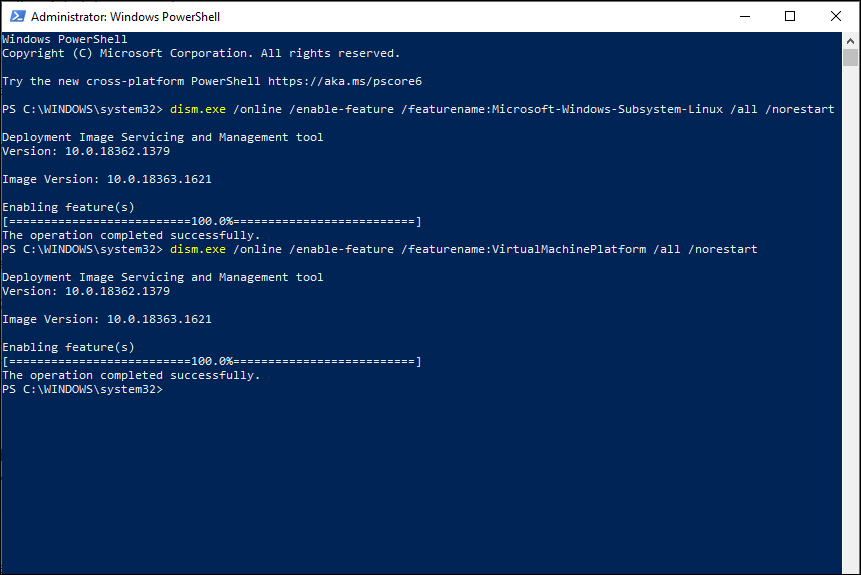


## Step 3 - Enable Virtual Machine feature

Open PowerShell as Administrator and run:

dism.exe /online /enable-feature /featurename:VirtualMachinePlatform /all /norestart

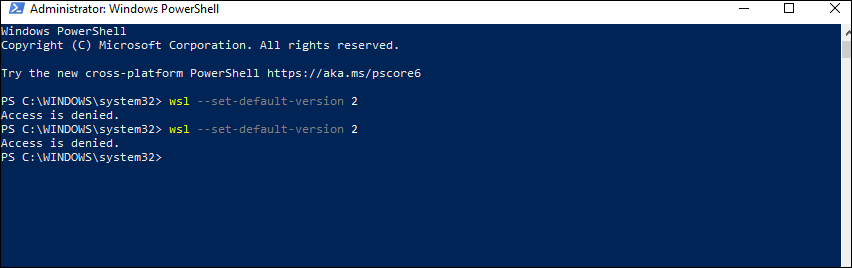
**Restart** your machine to complete the WSL install and update to WSL 2.



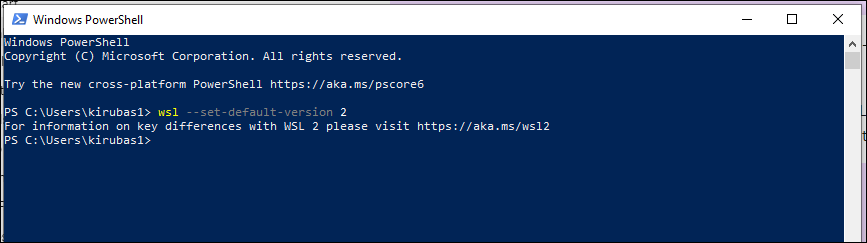
## Step 4 - Download the Linux kernel update package

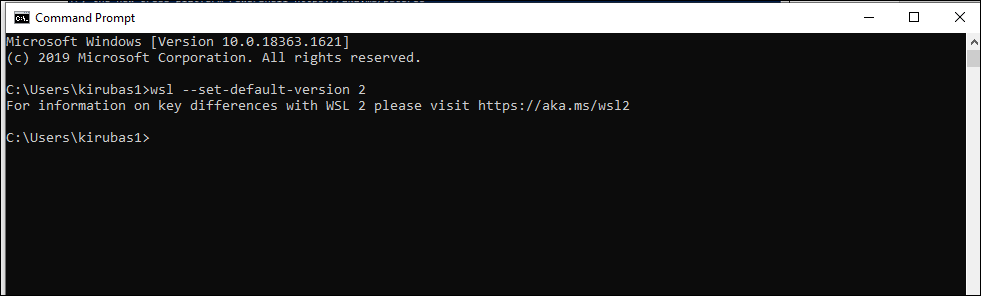
1. Download the latest package:

**\*\*\*The error shows up in elevated powershell.**



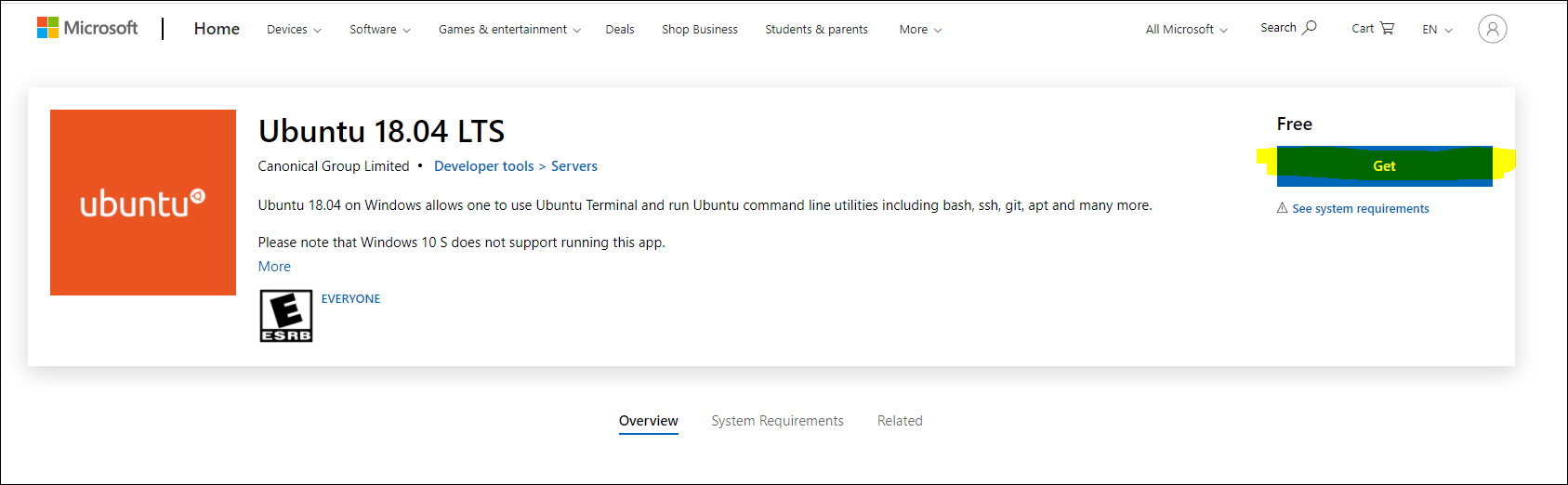
**A normal PowerShell can run all the commands.**





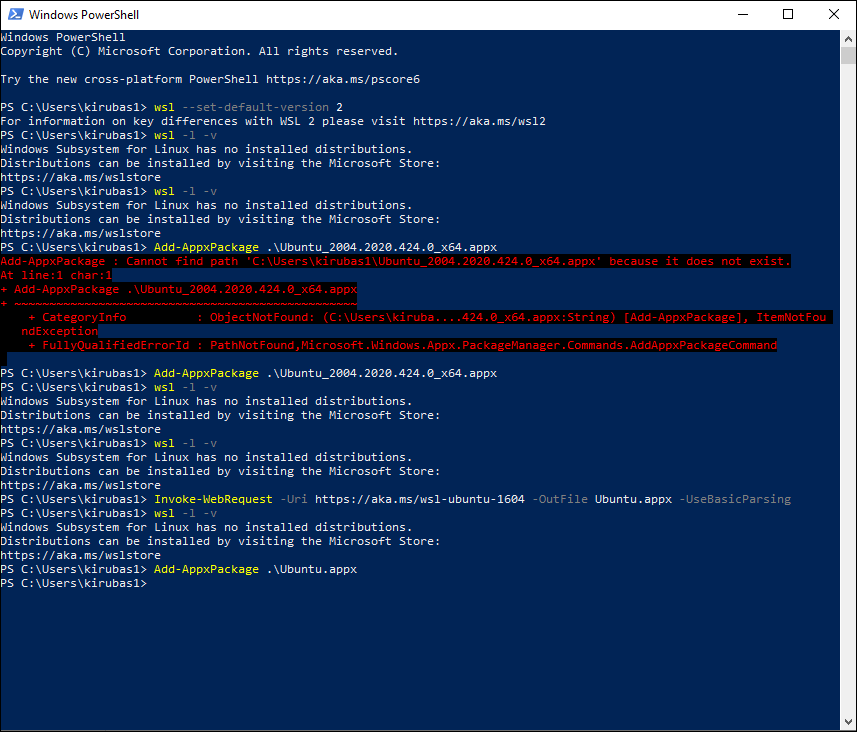
## Step 5- Install your Linux distribution of choice

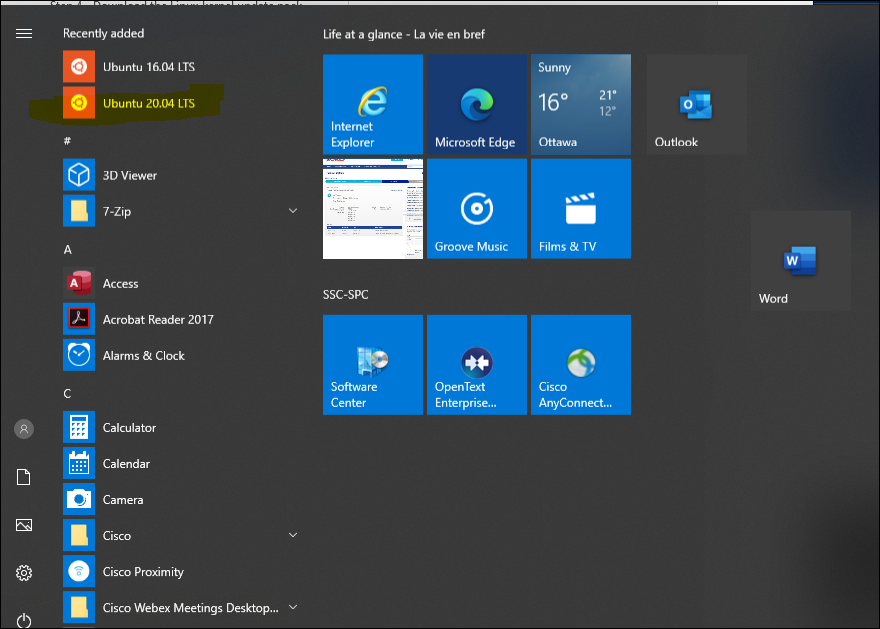
https://docs.microsoft.com/en-us/windows/wsl/install-manual

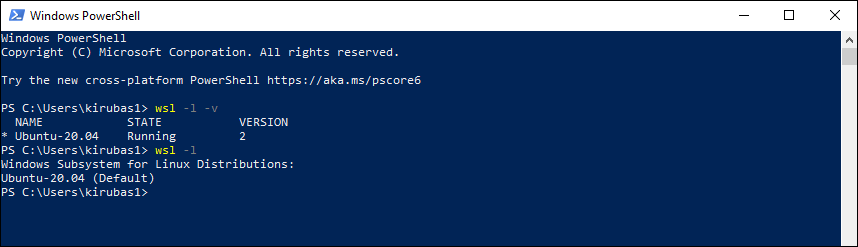


**Issues with GET button in Microsoft Store. Manually download Windows Subsystem for Linux distro packages.**

<https://docs.microsoft.com/en-us/windows/wsl/install-manual>

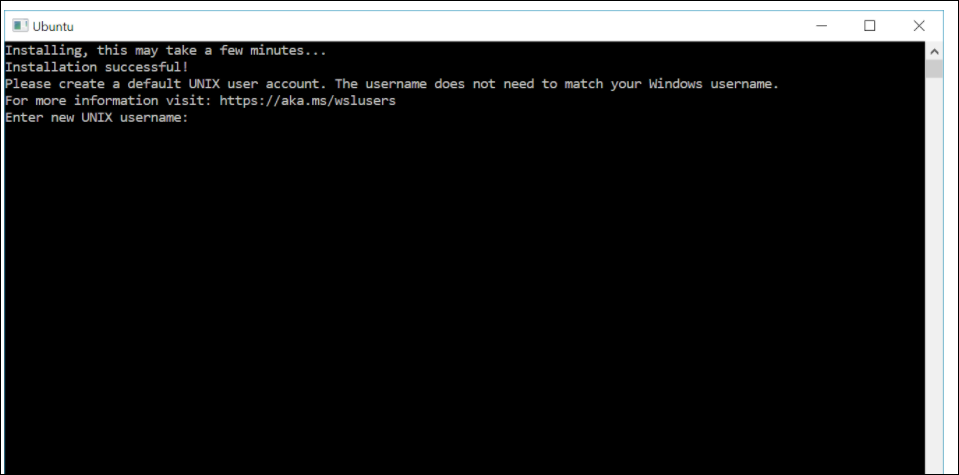


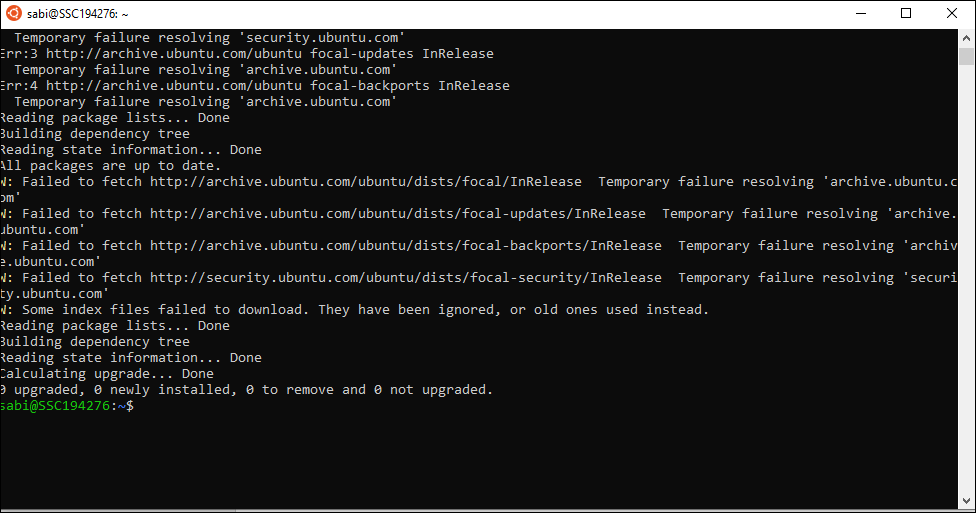




The first time you launch a newly installed Linux distribution, a console window will open and you'll be asked to wait for a minute or two for files to de-compress and be stored on your PC. All future launches should take less than a second.

**You will then need to create a user account and password for your new Linux distribution.**





<https://askubuntu.com/questions/91543/apt-get-update-fails-to-fetch-files-temporary-failure-resolving-error>

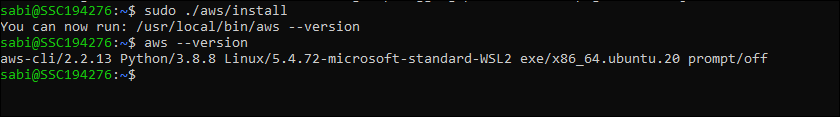


WSL changes IP address. So I had to run the DNS command in link above to make sure connectivity exists.

echo "nameserver 8.8.8.8" | sudo tee /etc/resolv.conf > /dev/null

# AWS CLI

<https://docs.aws.amazon.com/cli/latest/userguide/install-cliv2-linux.html>



|  |
| --- |
| **curl "https://awscli.amazonaws.com/awscli-exe-linux-x86\_64.zip" -o "awscliv2.zip"**  **unzip awscliv2.zip**  **sudo ./aws/install** |

Might be prompted to download zip first.

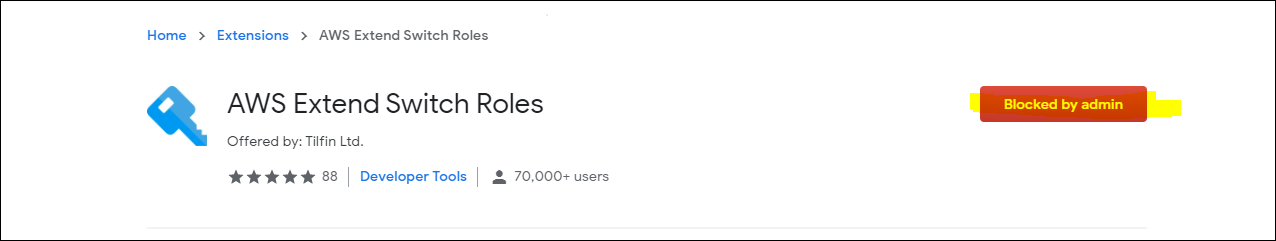
# Extended Role Switch

This chrome extension is useful when managing the large number of accounts that the AWS SEA deploys.

Below is a configuration used for the SSC Development environment.

Unfortunately the account numbers are removed so as not to store them in github, but the common names of roles and account names remain and just need to substituted with account IDs. Alternatively request a member of the team share their configuration over an approved channel.

SSC owns control of Google Chrome browser. Cannot install the extension.



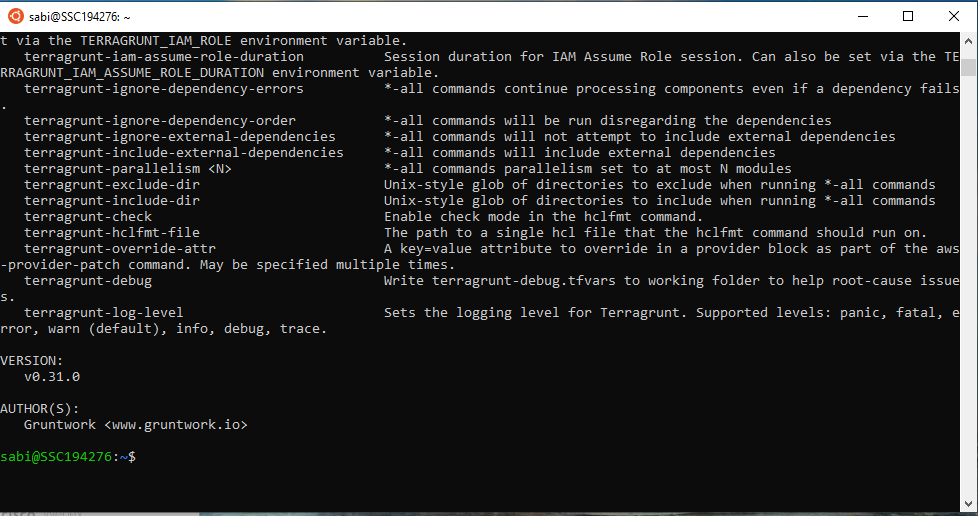
# Terraform

Ensure that your system is up to date, and you have the gnupg, software-properties-common, and curl packages installed. You will use these packages to verify HashiCorp's GPG signature, and install HashiCorp's Debian package repository.

|  |
| --- |
| $ sudo apt-get update && sudo apt-get install -y gnupg software-properties-common curl  Add the HashiCorp [GPG key](https://apt.releases.hashicorp.com/gpg).  $ curl -fsSL https://apt.releases.hashicorp.com/gpg | sudo apt-key add -  Add the official HashiCorp Linux repository.  $ sudo apt-add-repository "deb [arch=amd64] https://apt.releases.hashicorp.com $(lsb\_release -cs) main"  Update to add the repository, and install the Terraform CLI.  $ sudo apt-get update && sudo apt-get install terraform  Verify that the installation worked by opening a new terminal session and listing Terraform's available subcommands.  $ terraform -help  Usage: terraform [-version] [-help] <command> [args] |

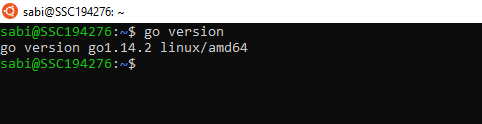
# Terragrunt

|  |
| --- |
| $wget https://github.com/gruntwork-io/terragrunt/releases/download/v0.31.0/terragrunt\_linux\_amd64  $ mv terragrunt\_linux\_amd64 terragrunt  $ chmod +x terragrunt  $ sudo mv terragrunt /usr/local/bin |



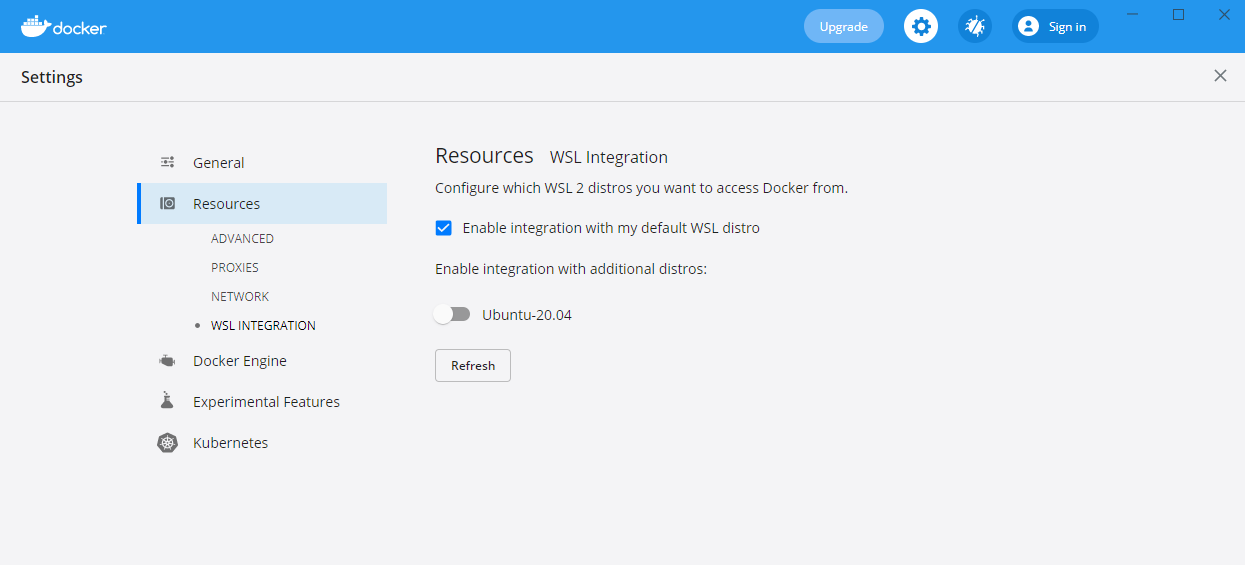
# Terratest / Go

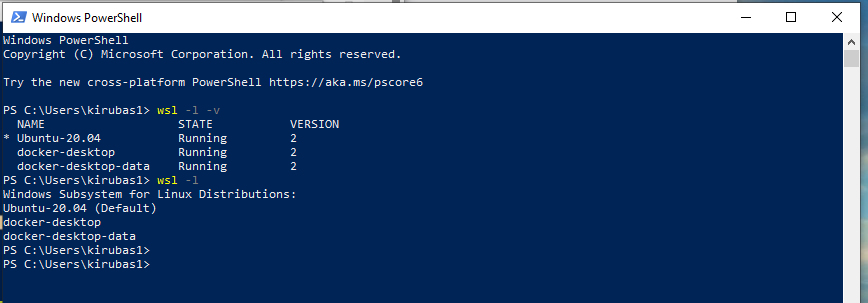
|  |
| --- |
| wget https://dl.google.com/go/go1.14.2.linux-amd64.tar.gz  sudo tar -xvf go1.14.2.linux-amd64.tar.gz  sudo mv go /usr/local  vi ~/.bashrc  -----press "I" for insert add the following at the bottom of the file    export GOROOT=/usr/local/go  export GOPATH=$HOME/go  export PATH=$GOPATH/bin:$GOROOT/bin:$PATH  --- "Esc" to exit insert mode and :wq! to save it    source ~/.bashrc    go version |

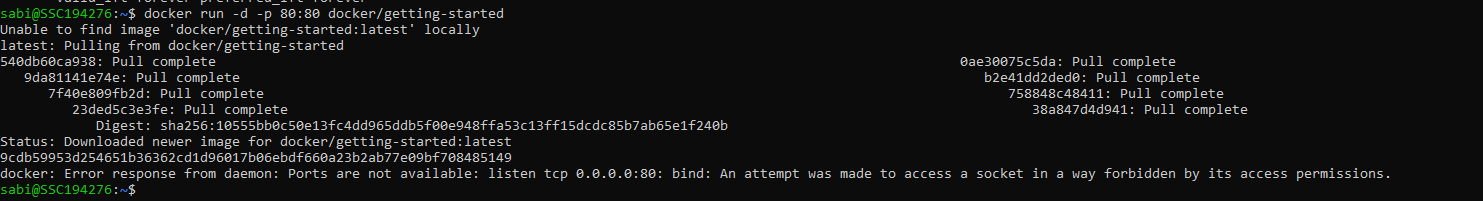


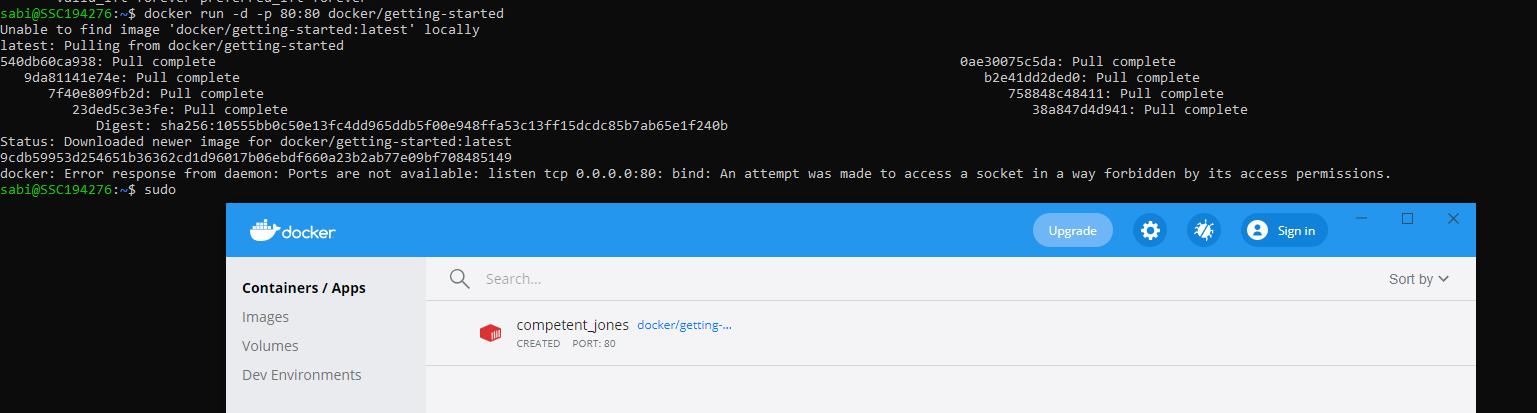
# Docker

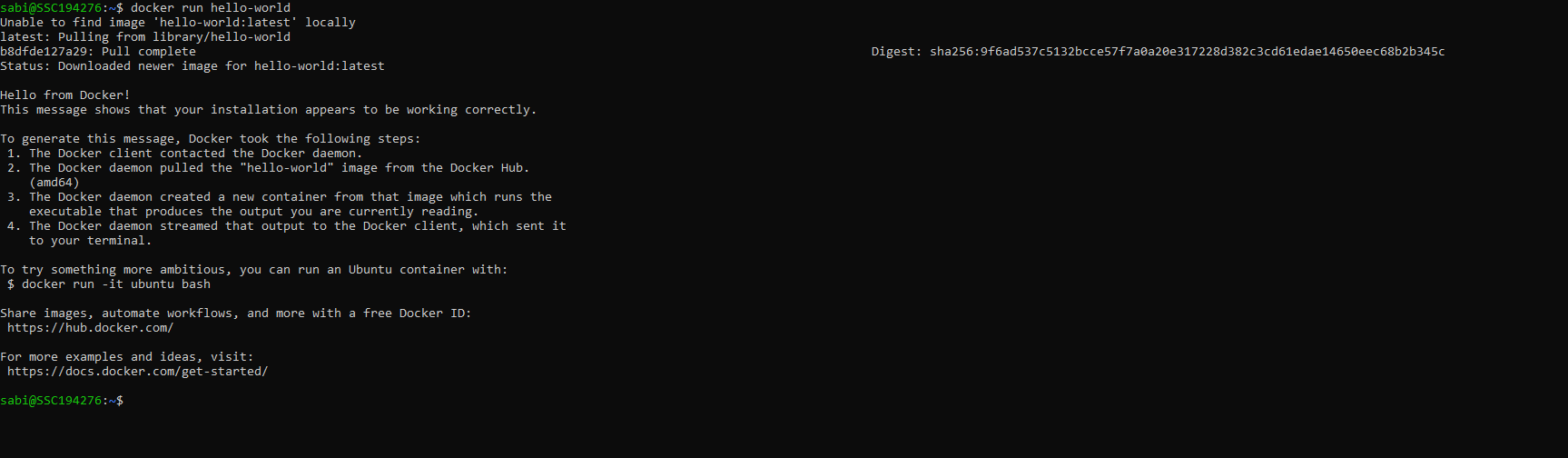
Install Docker Desktop











# Github and AWS Codecommit

# Repositories

