AWS CLI

Contents

[AWS CLI – Command Line Interface 3](#_Toc94864702)

[AWS CLI Commands 3](#_Toc94864703)

[Docker for CLI 3](#_Toc94864704)

[Configuration 3](#_Toc94864705)

[Docker Troubleshooting 4](#_Toc94864706)

[Before a Reset of Docker. 4](#_Toc94864707)

[After the Reset: 4](#_Toc94864708)

[https://stackoverflow.com/questions/57460393/the-docker-client-must-be-run-elevated-to-connect 4](#_Toc94864709)

[– Command Line Interface 5](#_Toc94864710)

[AWS CLI Commands 5](#_Toc94864711)

[Docker for CLI 5](#_Toc94864712)

[Configuration 6](#_Toc94864713)

[Docker Troubleshooting 6](#_Toc94864714)

[Before a Reset of Docker. 7](#_Toc94864715)

[After the Reset: 7](#_Toc94864716)

[https://stackoverflow.com/questions/57460393/the-docker-client-must-be-run-elevated-to-connect 7](#_Toc94864717)

# AWS CLI – Command Line Interface

**AWS CLI uses Windows command line – On Windows, run commands at the Windows command prompt or in PowerShell.**

To install on Windows:

<https://awscli.amazonaws.com/AWSCLIV2.msi>

# AWS CLI Commands

**>>> aws configure**

AWS Access Key ID [None]: AKIA6QAFBHRWS7AV3PU5

AWS Secret Access Key [None]: 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

Default region name [None]: us-west-2

Default output format [None]: table

**>> aws lambda list-functions**

**>> aws lambda list-functions --output text**

**>> aws lambda list-functions --output table**

**>> aws iam list-users --output json**

# Docker for CLI

>> **docker –version**

**>> docker run --rm -it amazon/aws-cli**

>> docker run --rm -it amazon/aws -cli –version

* docker – invocate the docker environment
* run – create a container
* --rm - causes Docker to automatically remove the container when it exits.
* -it – run docker interactively for amazon/aws
* –version – tbd

# Configuration

Before using Boto3, you need to set up authentication credentials for your AWS account using either the IAM Console or the AWS CLI. You can either choose an existing user or create a new one.

# Docker Troubleshooting

<https://github.com/docker/for-win/issues/12523>

<https://forums.docker.com/t/solved-docker-failed-to-start-docker-desktop-for-windows/106976>

<https://stackoverflow.com/questions/57460393/the-docker-client-must-be-run-elevated-to-connect>

|  |
| --- |
| Before a Reset of Docker. >> [System.IO.Directory]::GetAccessControl(“./pipe/docker\_engine”) | Format-Table -Warp  [image](https://user-images.githubusercontent.com/1080406/46826841-e9969000-cd4b-11e8-8026-5c457120c3ab.png) After the Reset: >> [System.IO.Directory]::GetAccessControl(“./pipe/docker\_engine”) | Format-Table -Warp  [image](https://user-images.githubusercontent.com/1080406/46826899-121e8a00-cd4c-11e8-8a77-39cbbbbc8e55.png) |

Top of Form

## <https://stackoverflow.com/questions/57460393/the-docker-client-must-be-run-elevated-to-connect>

"Hardware assisted virtualization and data execution protection must be enabled in the BIOS" Typically, this problem is due to a disabled Hyper-V or because the hypervisor agent is not running. Open a Windows shell as administrator and enable Hyper-V with:

**dism /Online /Enable-Feature:Microsoft-Hyper-V /All**

Reboot your machine and see if Docker for Windows starts. If the above doesn't work, open a Windows shell as administrator and enable the hypervisor with:

**bcdedit /set hypervisorlaunchtype auto**

If Docker for Windows is still not working properly, try the following additional solutions, which are based on instructions extracted from this Docker for Windows issue thread.

Reboot your machine and see if Docker for Windows starts. Copy this path: Control Panel\Programs\Programs and Features into a File Explorer address bar.

Click Turn Windows feature on or off on the left side of the Control Panel. Clear the Hyper-V check box in the list of Windows features.

Reboot your machine and start Docker for Windows (or reinstall if necessary). If the above doesn't work, then Virtualization Based Security (VBS) might be interfering and needs to be disabled. Try the following to disable VBS: Open the Local Group Policy Editor (gpedit.msc)

Go to Computer Configuration\Administrative Templates\System\Device Guard Disable VBS.

# – Command Line Interface

**AWS CLI uses Windows command line – On Windows, run commands at the Windows command prompt or in PowerShell.**

To install on Windows:

<https://awscli.amazonaws.com/AWSCLIV2.msi>

# AWS CLI Commands

**>>> aws configure**

AWS Access Key ID [None]: AKIA6QAFBHRWS7AV3PU5

AWS Secret Access Key [None]: 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

Default region name [None]: us-west-2

Default output format [None]: table

**>> aws lambda list-functions**

**>> aws lambda list-functions --output text**

**>> aws lambda list-functions --output table**

**>> aws iam list-users --output json**

# Docker for CLI

>> **docker –version**

**>> docker run --rm -it amazon/aws-cli**

>> docker run --rm -it amazon/aws -cli –version

* docker – invocate the docker environment
* run – create a container
* --rm - causes Docker to automatically remove the container when it exits.
* -it – run docker interactively for amazon/aws
* –version – tbd

# Configuration

Before using Boto3, you need to set up authentication credentials for your AWS account using either the IAM Console or the AWS CLI. You can either choose an existing user or create a new one.

# Docker Troubleshooting

<https://github.com/docker/for-win/issues/12523>

<https://forums.docker.com/t/solved-docker-failed-to-start-docker-desktop-for-windows/106976>

<https://stackoverflow.com/questions/57460393/the-docker-client-must-be-run-elevated-to-connect>

|  |
| --- |
| Before a Reset of Docker. >> [System.IO.Directory]::GetAccessControl(“./pipe/docker\_engine”) | Format-Table -Warp  [image](https://user-images.githubusercontent.com/1080406/46826841-e9969000-cd4b-11e8-8026-5c457120c3ab.png) After the Reset: >> [System.IO.Directory]::GetAccessControl(“./pipe/docker\_engine”) | Format-Table -Warp  [image](https://user-images.githubusercontent.com/1080406/46826899-121e8a00-cd4c-11e8-8a77-39cbbbbc8e55.png) |

Top of Form

## <https://stackoverflow.com/questions/57460393/the-docker-client-must-be-run-elevated-to-connect>

"Hardware assisted virtualization and data execution protection must be enabled in the BIOS" Typically, this problem is due to a disabled Hyper-V or because the hypervisor agent is not running. Open a Windows shell as administrator and enable Hyper-V with:

**dism /Online /Enable-Feature:Microsoft-Hyper-V /All**

Reboot your machine and see if Docker for Windows starts. If the above doesn't work, open a Windows shell as administrator and enable the hypervisor with:

**bcdedit /set hypervisorlaunchtype auto**

If Docker for Windows is still not working properly, try the following additional solutions, which are based on instructions extracted from this Docker for Windows issue thread.

Reboot your machine and see if Docker for Windows starts. Copy this path: Control Panel\Programs\Programs and Features into a File Explorer address bar.

Click Turn Windows feature on or off on the left side of the Control Panel. Clear the Hyper-V check box in the list of Windows features.

Reboot your machine and start Docker for Windows (or reinstall if necessary). If the above doesn't work, then Virtualization Based Security (VBS) might be interfering and needs to be disabled. Try the following to disable VBS: Open the Local Group Policy Editor (gpedit.msc)

Go to Computer Configuration\Administrative Templates\System\Device Guard Disable VBS.