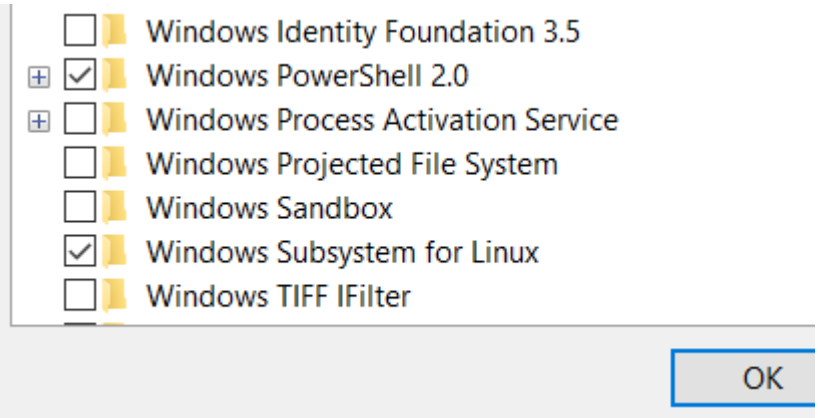
SIFT-WSL Pre-Cooked

Installing SIFT Workstation (Server mode) under Windows Subsystem for Linux (WSL)

The following instructions will guide you through download and installation of a command line version of SIFT workstation that you can invoke (as well as all the tools included) from a Windows shell.

Pre-requisite: Verify that Windows Subsystem for Linux is enabled (optional Windows Components)



Download the SIFT-wsl precooked distribution

Find SIFT-wsl-18.7z and download it to your computer.
Extract the .7z (7zip) to the target directory.
The extracted file SIFT-wsl-18.04 should be 5.91 GB

```
get-filehash .\SIFT-wsl-18.04

Algorithm Hash
--------- ----
SHA256
1D9EB82FD43E70AD8F2B57B6BC63F5ADAEEF6C5867881CCF3F8C3CD935C4EF13
```

[PowerShell]

List installed distributions

```
PS D:\PowerShell> wsl --list
Windows Subsystem for Linux Distributions:
kali-linux (Default)
Ubuntu-18.04
```

Import the SIFT-WSL distro

Syntax: wsl --import {name of distro} {where the distro will live} {path to precooked distro}

In this case I made a WSL directory on my D: drive, and a sub-directory for SIFT. The distro file is in the subdirectory already when running the import command.

```
PS D:\WSL> wsl --import SIFT-WSL D:\WSL\SIFT\ D:\WSL\SIFT\SIFT-wsl-18.04
```

It will take a few+ minutes to process. Once it does and you are back at a prompt, re-run the command to list the installed distros. You should now see **SIFT-WSL** listed.

```
PS D:\PowerShell> wsl --list
Windows Subsystem for Linux Distributions:
kali-linux (Default)
Ubuntu-18.04
SIFT-WSL
```

Launch SIFT-WSL

From any command prompt you should now be able to call **wsl -d SIFT-WSL,** and you will be running a CLI version of SIFT server.

```
PS D:\PowerShell> wsl -d SIFT-WSL
```

My usual test to make sure all is operational is to run vol.py -h (validates Volatility - my #1 SIFT tool, is functional).

```
PS D:\PowerShell> wsl -d SIFT-WSL
root@HOSTNAME:/mnt/d/PowerShell# vol.py -h
Volatility Foundation Volatility Framework 2.6.1
Usage: Volatility - A memory forensics analysis platform.

Options:
-h, --help list all available options and their default values.
Default values may be set in the configuration...
```

Create a User "forensicator"

Can be the name of your choice. I prefer this for Op-Sec in screen captures.

```
root@HOSTNAME:/mnt/d/PowerShell# sudo useradd -m forensicator
root@HOSTNAME:/mnt/d/PowerShell# sudo passwd forensicator
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
root@HOSTNAME:/mnt/d/PowerShell# sudo usermod -aG sudo forensicator
```

Switch to forensicator user

```
root@HOSTNAME:/mnt/d/PowerShell# su - forensicator
```

(re)Verify (Volatility)

```
$ vol.py -h
Volatility Foundation Volatility Framework 2.6.1
Usage: Volatility - A memory forensics analysis platform.

Options:
-h, --help list all available options and their default values.
Default values may be set in the configuration file
(/etc/volatilityrc)
```