

Windows Subsystem for Linux

Installation

- Open Windows Power Shell as Administrator, then run the following line and reboot:

```
Enable-WindowsOptionalFeature -Online -FeatureName Microsoft-Windows-Subsystem-Linux
```

- Open the Microsoft Store and search for "Linux". Choose and get Debian, as per 2020-03-10 this is Debian 10 Buster.
- Open *Debian* from the start menu and complete installation.
- Upgrade all packages:

```
sudo apt update  
sudo apt upgrade
```

- Install packages:

```
sudo apt install vim  
sudo apt install wget  
sudo apt install zip  
sudo apt install rsync
```

- Create /run/sshd folder:

```
sudo mkdir /run/sshd
```

- If you want to add a shortcut to your desktop, run the following on the command line to find the exe file:

```
which debian
```

- The root file system can be found at a path similar to:

```
C:\Users\bco\AppData\Local\Packages\TheDebianProject.DebianGNUlinux_76v4gfsz19hv4\LocalState\rootfs
```

- Add  and  icons to the shortcuts

SSH Server

- Install packages:

```
sudo apt install openssh-server
```

- Follow the [SSH Client and Server](#) guide. SSH into your host and open port 22 in the Windows Firewall.

Start SSH Server at Windows boot time

- Open a bash command shell and allow <user> to run ssh as root, add the following line after %sudo:

```
sudo visudo
%sudo ALL=(ALL:ALL) ALL
<user> ALL=(root:root) NOPASSWD: /etc/init.d/ssh
```

- Open Windows Task Scheduler and click *Create Basic Task...*
 1. General
 - Name : Start Bash SSH Server
 - Description: Start the WSL SSH Server via a bash command
 - Run only when user is logged on
 - Run with highest privileges
 2. Trigger
 - Begin the task: At log on
 - Specific user: <user>
 - Enabled
 3. Actions
 - Action: Start a program
 - Program/script: C:\Windows\System32\bash.exe
 - Add arguments (optional): -c "/usr/bin/sudo /etc/init.d/ssh start"
 4. Conditions
 - Start the task only if the computer is on AC power
 5. Settings
 - *Uncheck* Stop the task if it runs longer than:
 6. Alternative Action when starting a non elevated script "startup"
 - Action: Start a program
 - Program/script: C:\Windows\System32\bash.exe
 - Add arguments (optional): -c "/home/user/startup"
 - Script "startup":

```
#!/bin/bash
sudo /etc/init.d/ssh start
```

Mounting encrypted drives

- WSL can mount encrypted drives, but they must be mounted at the time wsl or bash runs the first time after boot.
- If you do not execute any tasks when the host boots up, then just make sure your encrypted drives are mounted when you run wsl or bash.
- If you **do** execute WSL tasks when the host boots up, replace the *Task Scheduler* entry for the trigger and make it run after the *Mount* script has executed:

```
<QueryList>
  <Query Id="0" Path="Microsoft-Windows-TaskScheduler/Operational">
    <Select Path="Microsoft-Windows-TaskScheduler/Operational">*[EventData[@Name='TaskSuccessEvent']][Data[@
```

```
Name='TaskName' ]='\Mount' ]]</Select>  
</Query>  
</QueryList>
```

Links

- [Windows Subsystem for Linux Installation Guide for Windows 10](#)
- [How Does the Windows 10 Subsystem for Linux Work and What Is It For?](#)
- [Background Task Support in WSL](#)
- [SSH on Windows Subsystem for Linux \(WSL\)](#)
- [Backup with rsync on Windows \(WSL\)](#)
- [Docker Running Seamlessly in Windows Subsystem Linux](#)

From:

<https://wiki.condrau.com/> - **Bernard's Wiki**

Permanent link:

<https://wiki.condrau.com/win10:wsl?rev=1584280605>

Last update: **2020/03/15 20:56**

