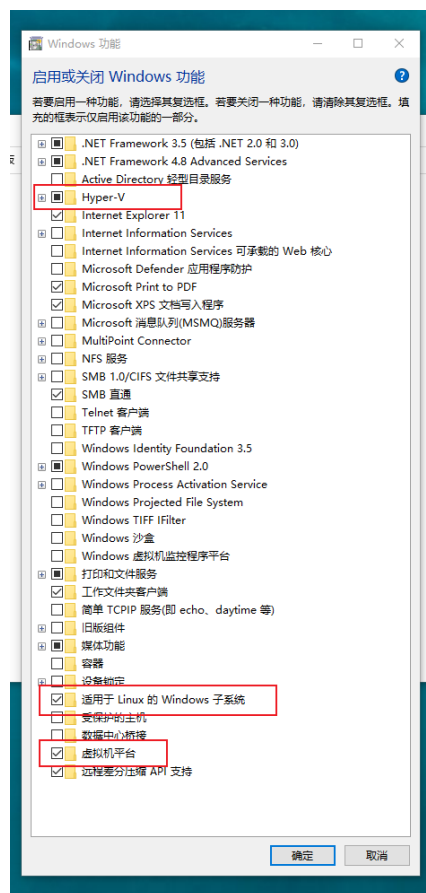


Use WSL2 to compile Yocto

1. Set up the installation environment for WSL2

Step 1: Turn on Virtualization Technology (VT)in BIOS, divided into IntelVT and AMD VTs. It must be turned on so that Ubuntu does not make any mistakes when it is installed.

Step 2: Find the control panel and open the program - turn Windows on or off. Turn on the developer options in the system settings.



Command-line methods (all commands need to be executed in PowerShell with administrator rights):

```
dism.exe /online /enable-feature /featurename:Microsoft-Windows-Subsystem-
```

```
Linux /all /norestart
```

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

Then restart your computer as prompted.

2. Install WSL2

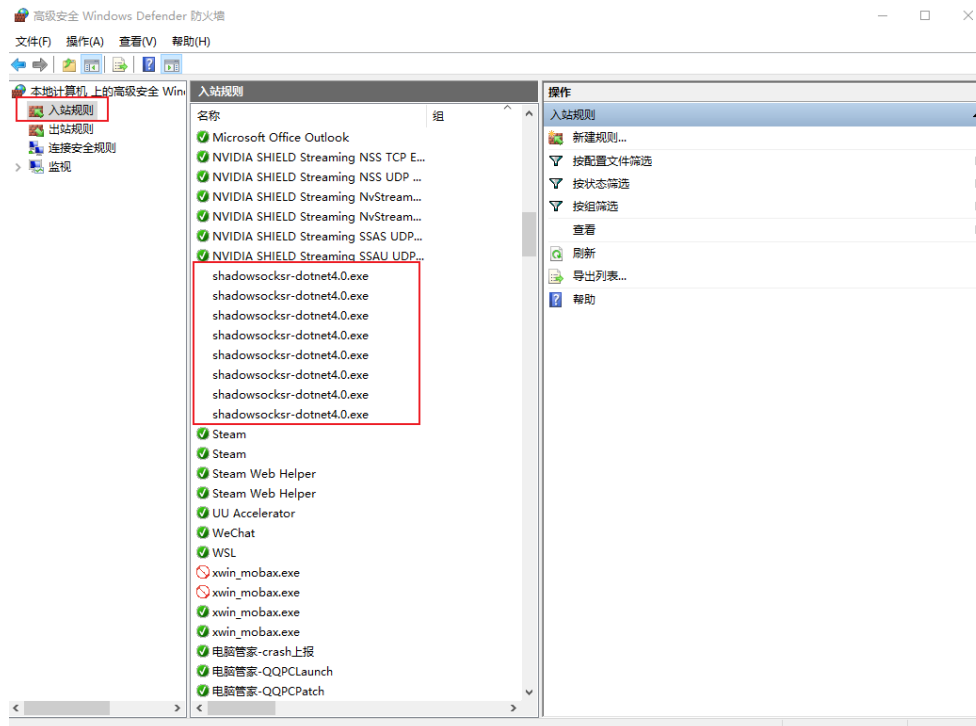
Set the default version to WSL2: **wsl --set-default-version 2**. Only the WSL2 system is more complete and compilation does not go wrong. Search for and install Ubuntu18.04/ Ubuntu20.04 in Windows Store. Once installed, you can use **wsl --list -verbose** to view the current systemstatus.

```
PS C:\Windows\System32> wsl --list --verbose
NAME          STATE      VERSION
* Ubuntu-18.04  Running   2
PS C:\Windows\System32>
```

3. Configure the agent at .bashrc (CN must do it)

```
export hostip=$(cat /etc/resolv.conf |grep -oP '(?<=nameserver\ ).*')
export https_proxy="http://${hostip}:8080"
export http_proxy="http://${hostip}:8080"
export all_proxy="socks5://${hostip}:8080"
```

Listen to the 8080 port of Windows' proxy server. You also need to turn off the review of proxy tools in windows' firewall policies.



4. Move the Ubuntu to a non-system disk

Do not install ubuntu to disk C, and then download yocto on another disk, there will be a lot of file system access errors, because WSL2 file system permissions do not fully control the Windows disk. Therefore, the installed WSL needs to be moved to a non-system disk.

The export distribution is a tar file to a d disk

```
wsl --export Ubuntu-20.04 d:\wsl-ubuntu20.04.tar
```

Sign out of the current distribution

```
wsl --unregister Ubuntu-20.04
```

Re-import and install WSL on the D disk

```
wsl --import Ubuntu-20.04 d:\wsl-ubuntu20.04 d:\wsl-ubuntu20.04.tar --version 2
```

Set the default landing user to be the user name at the time of installation

```
ubuntu2004 config --default-user USERNAME
```

Delete the wsl-ubuntu20.04 .tar

```
del d:\wsl-ubuntu20.04.tar
```

Finally, you can compile yocto according to the manual!

One last point: WSL2 enables file interoperability with Windows. You can enter \\wsl\$\Ubuntu-20.04 by entering the windows resource management bar, or copy it directly to the Windows file system under CTRL-C.

Note:

Errors are reported when compiling nxp-wlan:

```
make[1]: *** /lib/modules/4.19.128-microsoft-standard/build: No such file or directory. Stop.
```

The WSL2 kernel does not have a build tool and needs to download the wsl kernel to compile the build tool.

The solution is as follows:

```
$ wget https://github.com/microsoft/WSL2-Linux-Kernel/archive/4.19.128-microsoft-standard.tar.gz
$ tar xf 4.19.128-microsoft-standard.tar.gz
$ cd WSL2-Linux-Kernel-4.19.128-microsoft-standard/
$ make defconfig
$ sudo make modules_prepare
$ sudo mkdir -p /lib/modules/4.19.128-microsoft-standard
$ sudo ln -s /home/****/WSL2-Linux-Kernel-4.19.128-microsoft-standard/ /lib/modules/4.19.128-microsoft-standard/build
```

Compiled successfully:

```
lzm@WIN-JRIFK9G98EH: /imx-yocto-bsp/build_imx3mm$ bitbake imx-image-core
Loading cache: 100% |#####| Time: 0:00:00
Loaded 4623 entries from dependency cache.
Parsing recipes: 100% |#####| Time: 0:00:11
Parsing of 3209 .bb files complete (3208 cached, 1 parsed). 4624 targets, 469 skipped, 1 masked, 0 errors.
NOTE: Resolving any missing task queue dependencies

Build Configuration:
BB_VERSION      = "1.44.0"
BUILD_SYS      = "x86_64-linux"
NATIVELSBSTRNG = "ubuntu-18.04"
TARGET_SYS     = "aarch64-poky-linux"
MACHINE        = "imx3mmevk"
DISTRO         = "fsl-imx-wayland"
DISTRO_VERSION = "5.4-zeus"
TUNE_FEATURES  = "aarch64 cortexa53 crc crypto"
TARGET_FPU     = ""
meta
meta-poky      = "HEAD:5408e48d374fc741fa46b57ddfd3a537843bda6f"
meta-oe
meta-multimedia
meta-python    = "HEAD:2b5dd1eb81cd08bc065bc76125f2856e9383e98b"
meta-freescale = "HEAD:14f1a630a47375432f93c556927b879b51d84c4e"
meta-freescale-3rdparty = "HEAD:dbcc686f52c3c84db8cb86aa8973a4e373651b98"
meta-freescale-distro = "HEAD:ca27d12e4964d1336e662bcc60184bbf526c857"
meta-bsp
meta-sdk
meta-ml        = "HEAD:780b023a4c77ac9c0011d858ad377f4a9593bfea"
meta-nxp-demo-experience = "HEAD:c7263d9f3cc7bbf44e7164ffeda494cf283d3dec"
meta-browser   = "HEAD:ee3be3b5986a4aa0e73df2204a625ae1fe5df37e"
meta-rust      = "HEAD:7f235b6f8973cc5269448375f2a3f9867bb2a369"
meta-clang     = "HEAD:711e593d5984aad3bf35c51b7ac4482982bc16c7"
meta-gnome
meta-networking
meta-filestems = "HEAD:2b5dd1eb81cd08bc065bc76125f2856e9383e98b"
meta-qt5       = "HEAD:5144a99a696a8a72c0765ea0e8be8ceafaa78f85"
meta-python2   = "HEAD:4400f9155ec193d028208cf0c66aeed2ba2b00ab"

Initialising tasks: 100% |#####| Time: 0:00:02
Sstate summary: Wanted 0 Found 0 Missed 0 Current 1721 (0% match, 100% complete)
NOTE: Executing Tasks
NOTE: Setscene tasks completed
NOTE: Tasks Summary: Attempted 4703 tasks of which 4703 didn't need to be rerun and all succeeded.
lzm@WIN-JRIFK9G98EH: /imx-yocto-bsp/build_imx3mm$
```

<https://www.catalog.update.microsoft.com/Search.aspx?q=wsl>

Download the install kernel 5.4 update, and then in shell:wsl-shutdown, the restart is the new kernel.