The Yocto Project

Yocto is a Embedded LInux Build System. Yocto provides a flexible set of tools and configurations to help design tailored Linux images for a variety of embedded and IOT devices.

Layers

The software stack is seperated into a series of Layers. Like a cookbook, each layer contains a series of recipes. The recipes tells the compiler how to build applications. Because the architecture of each Embedded Device is different, the applications must be built differently for each device. Yocto handles the device specific configuration for building applications.

Poky

Poky is a reference distribution designed for the Yocto Project. Linux is a family of open-source Unix-like Operating Systems. Each member of the Linux family is called a distribution. Each distribution contains a Linux kernel and package manager. The package manager handles system and app updates. Each distro handles updates differently. The Poky distribution contains sane defaults that can be used by themselves or used to create another distribution.

OpenEmbedded

OpenEmbedded is a robust build system designed for comiling applications for Embedded Systems. The Yocto Project has incorporated OpenEmbedded into their own software stack.

Raspberry Pi

This layer is a Board Support Package or BSP. The layer contains information about supported features on the Raspberry Pi. The layer also contains additional changes to other recipes to help build proper support for the Raspberry Pi.

Qt6

Qt is a modern UX design tool with cross-platform support for Linux, Mac-OS, Winodws, and Android. The Qt layer provides support for the Qt framework in yocto.

Geek Tech

Geektech is a layer designed to build support for the Cybersecurity STEM Learning Course. The course teaches common cyber attacks and the importance of online security.

Build

Dependencies

- Git 1.8.3.1+
- Tar 1.28+
- Python 3.6.0+
- Gcc 5.0+

Install the following development host packages. The required packages will vary depending on the host distribution.

- Ubuntu / Debian / openSUSE / CentOS / Linux Mint
  - `sudo apt install gawk wget git diffstat unzip texinfo gcc build-essential chrpath socat cpio python3 python3-pip python3-pexpect xz-utils debianutils iputils-ping python3-git python3-jinja2 libegl1-mesa libsdl1.2-dev pylint3 xterm python3-subunit mesa-common-dev zstd liblz4-tool bzip2`
- Archlinux / Manjaro
  - `sudo pacman -S --needed git diffstat unzip texinfo python chrpath wget xterm sdl rpcsvc-proto socat cpio inetutils bzip2`
  - Enable the multilib repository and install multilib-devel
- Windows
  - For setting up a operating system agnostic yocto environment, refer to the following
    - https://docs.yoctoproject.org/4.0.5/dev-manual/start.html#setting-up-to-use-cross-platforms-crops
  - For setting up a yocto environment using WSL v2, refer to the following
    - https://docs.yoctoproject.org/4.0.5/dev-manual/start.html#setting-up-to-use-windows-subsystem-for-linux-wslv2

Setup Yocto Environment

```
git clone --branch kirkstone git://git.yoctoproject.org/poky.git

cd poky
```

```
# add layers
git clone -b kirkstone git://git.openembedded.org/meta-openembedded
git clone -b kirkstone git://git.yoctoproject.org/meta-raspberrypi
git clone -b 6.4 git://code.qt.io/yocto/meta-qt6.git
git clone -b kirkstone https://github.com/GeekTech-mirror/meta-geektech.git
```

Building Raspberry Pi 4 image

This image Launches the Cybersecurity Lesson on boot.

```
# setup up build environment
source oe-init-build-env

# copy geektech configs
cp -v ../meta-geektech/conf/local.conf.pi4 conf/local.conf
cp -v ../meta-geektech/conf/bblayers.conf.sample conf/bblayers.conf

# build yocto image
bitbake core-image-base
```

Building Raspberry Pi Zero WiFi image

This image sets up a bare bones access point for use with the Cybersecurity Lesson. Avoid connecting to the access point for extended periods of time, as the network is not very secure.

```
# setup up build environment
source oe-init-build-env

# copy geektech configs
cp -v ../meta-geektech/conf/local.conf.pizero conf/local.conf
cp -v ../meta-geektech/conf/bblayers.conf.sample conf/bblayers.conf

# build yocto image
bitbake core-image-base
```

Flashing Raspberry Pi 4 image

To flash the image, plug in a removable device, such as a micro sd-card. Replace /dev/sdX with the path of the device you want to flash the image to. Use something like `lsblk -o'NAME,SIZE,MODEL,MOUNTPOINT'` to search for your specific device.

```
cd tmp/deploy/images/raspberrypi4-64
bzip2 -dkf core-image-base-raspberrypi4-64.wic.bz2
sudo dd if=core-image-base-raspberrypi4-64.wic of=/dev/sdX
```

Flashing Raspberry Pi Zero WiFi image

To flash the image plug in a removable device, such as a micro sd-card. Replace /dev/sdX with the path of the device you want to flash the image to. Use something like `lsblk -o'NAME,SIZE,MODEL,MOUNTPOINT'` to search for your specific device.

```
cd tmp/deploy/images/raspberrypi0-wifi
bzip2 -dkf core-image-base-raspberrypi0-wifi.wic.bz2
sudo dd if=core-image-base-raspberrypi0-wifi.wic of=/dev/sdX
```