Beginners Guide to Writing a Recipe for OpenEmbedded and Yocto Projects

Author: Leah Thompson
Date: 7/11/2014
Version 1.0
www.wolfssl.com

This tutorial comprises simple instructions for writing a recipe for the Yocto Project/OpenEmbedded and then installing it on a linux image. The Yocto Project has a quickstart guide that is helpful in getting a basic introduction in how to build an image and run it on a machine. For the Yocto Project, it is a good idea to download the latest version of Poky (which contains packages for building a certain image). The quickstart guide for Yocto recommends going into a specific branch, however, for our build, we stayed in master for all layers.

After downloading Poky, you may need a board support package(bsp) for your given device, in our case, the Minnowboard, which utilizes the meta-minnow bsp and meta-intel layers (described below). You can find these layers in the Yocto Project Source Repositories.

WRITING THE RECIPE

1. **Layer Directory**
   Create layer directory and name it meta-<name> i.e., meta-wolfssl.

2. **Conf File**
   Inside of your layer directory, create a conf directory with a layer.conf file stored in it:
   conf/layer.conf.
   This file is required by all bitbake images. It should contain standard information that can be seen here.

3. **Recipe Directory**
   Create a recipe directory in the layer i.e., meta-wolfssl/cyassl.
   This is the name used when adding the recipe package to a Yocto Project bitbake image.

4. **.bb File**
   Create the bitbake recipe file, i.e., meta-wolfssl/cyassl/cyassl_3.0.0.bb. File format: <package name>_<version>.bb file

5. **Formatting the .bb Recipe**

For formatting your recipe, check out the Yocto Project Development manual page - make note of whether you are on the Development Manual or the Reference manual, as they are 2 separate items - and the OpenEmbedded user manual page.

6. **Terminology**

Some common terms used in recipes:

${WORKDIR} - this is the location inside of your build (after you run source oe-init-build-env) and is referring to:

```
build/tmp/work/<machine_architecture>/<recipe_name>/<recipe_version>
```

${PN} - Package name, i.e., name of the recipe. Same as the name you gave your .bb file.
${PV} - Package version, i.e., version of the recipe.

SRC_URI = “the file your recipe is built off of.” This can be a .zip, .c, or whichever file you need to build your recipe with.

md5, sha256 - these can be found by typing “openssl md5 <file_name>” into the command line.

For understanding more of the terminology used in your recipe, check out the Yocto Project Reference manual here.

ADDING THE RECIPE TO YOUR IMAGE

1. **Add Layer Path**

After running source oe-init-build-env, the file location of your layer should be added to your bblayers.conf file. This file is located in build/conf/bblayers.conf. Enter the location of the layer in:

```
BBLAYERS ?=
```

This is what the conf/bblayers.conf for meta-wolfssl on the Minnowboard looks like:

```
BBLAYERS ?= " \
/home/name/poky/meta \
/home/name/poky/meta-yocto \
/home/name/poky/meta-yocto-bsp \
/home/name/poky/meta-intel \
/home/name/poky/meta-minnow \
/home/name/wolfSSL/meta-wolfssl \
"
```

2. **Append the Package**

   Within the build/conf/local.conf file, add this line anywhere:

   ```
   IMAGE_INSTALL_append = " cyassl benchmark ctaocrypttest"
   ```

   We included all three recipe packages from meta-wolfssl here. You can add multiple recipes from different layers (as long as these layers have a path located in step one) here or just one, or none.

3. **Add Package to Image Recipe**

   If you installed Poky from the Yocto Project, the recipe for your image will be located at meta/recipes-<name>/images/<name-of-recipe>.bb. You can create your own custom image recipe, or add to an existing one. Since we used core-image-sato, we edited core-image-sato.bb in this location. If you do this, make sure you know exactly which changes you made so you can revert back to the original image if needed. To add a new recipe to your image, add this line (or add to the IMAGE_INSTALL line if it already exists):

   ```
   IMAGE_INSTALL += "package-name package2-name package3-name"
   ```

4. **Run bitbake Command**

   After you have created and inserted everything into your image, you can bitbake the image and run it on your board. You can also bitbake each recipe individually to check for errors before trying to build the entire image. This can be done simply by typing:

   ```
   bitbake <recipe-name>
   ```

   where <recipe-name> is the directory where your .bb file is stored. Then you will run bitbake image-name

5. **Types of Pre-Built Images**

   A list of the various Yocto Project images can be found here. The image we used for building with meta-wolfssl on the Minnowboard was core-image-sato. This image takes quite a while to build - expect over a day if you are building inside a virtual machine. It is also very large - not the image itself, but the build components take up almost 50GB, so make sure you have enough space on your computer. The actual image that will be booted onto your embedded device will be very small.