Yocto-SPDX Walkthrough

Daniel Wright
Timothy Strevey
Thomas T. Gurney
Kevin Lumbard

Goal

- Cover installation instructions
- Starting a build

Clone DoSOCS

▸ First step is to clone the DoSOCS repo

```
tstrevey@spdxdev:~/DOSOCS$ git clone https://github.com/socs-dev-env/DoSOCS
Cloning into 'DoSOCS'...
remote: Counting objects: 351, done.
remote: Total 351 (delta 0), reused 0 (delta 0), pack-reused 350
Receiving objects: 100% (351/351), 276.54 KiB, done.
Resolving deltas: 100% (238/238), done.
tstrevey@spdxdev:~/DOSOCS$
```

Edit the DoSOCS install script

- Add the correct username and password for the DoSOCS database to the install.sh script.
- On spdxdev the username and password are "spdx"

```
#Setup User Name and Password for MySql
u=spdx
p=spdx
```

Run the install script

```
tstrevey@spdxdev:~/DOSOCS/DoSOCS$ ./install.sh
Cloning into 'SOCSDatabase'...
remote: Counting objects: 890, done.
remote: Compressing objects: 100% (612/612), done.
remote: Total 890 (delta 235), reused 890 (delta 235), pack-reused 0
Receiving objects: 100% (890/890), 5.93 MiB | 8.96 MiB/s, done.
Resolving deltas: 100% (235/235), done.
Cloning into 'DoSOCS'...
remote: Counting objects: 351, done.
remote: Total 351 (delta 0), reused 0 (delta 0), pack-reused 350
Receiving objects: 100% (351/351), 276.54 KiB, done.
Resolving deltas: 100% (238/238), done.
Cloning into 'SOCSDashboard'...
remote: Counting objects: 340, done.
remote: Total 340 (delta 0), reused 0 (delta 0), pack-reused 340
Receiving objects: 100% (340/340), 524.17 KiB, done.
Resolving deltas: 100% (212/212), done.
Install SPDX Database...
Creating Upload Directory...
Reconfigure Repo Folder Structure
Removing old Repo Directories
Remove Database Repo
Install Complete
Don't forget to update the setting files ('DoSOCS/settings.py' AND 'SOCSDashboard/function/Data_Source.php
') with the database
connection information, and with the paths to Ninka and FOSSology.
tstrevey@spdxdev:~/DOSOCS/DoSOCS$ 
```

Edit DoSOCS/settings.py

- The install script will create a file named settings.py.
- Copy the username and password from before as well as set the path for FOSSology.

```
'''
Paths to scanners
'''
NINKA_PATH = "/usr/share/ninka/ninka.pl"
FOSSOLOGY_PATH = "/usr/share/fossology/nomos/agent/nomos"

'''
Database connection settings
'''
database_user = "spdx"
database_pass = "spdx"
database_host = "localhost"
database_port = "3306"
database_name = "SPDX"
~
~
~
~
```

Clone more Repos

▸ Move back to the home directory and clone our Yocto-SPDX repo as well as the yocto project repo

```
tstrevey@spdxdev:~$ git clone https://github.com/ttgurney/yocto-spdx
Cloning into 'yocto-spdx'...
remote: Counting objects: 541, done.
remote: Compressing objects: 100% (167/167), done.
remote: Total 541 (delta 95), reused 0 (delta 0), pack-reused 374
Receiving objects: 100% (541/541), 434.70 KiB, done.
Resolving deltas: 100% (318/318), done.
tstrevey@spdxdev:~$
```

```
tstrevey@spdxdev:~$ git clone http://git.yoctoproject.org/git/poky
Cloning into 'poky'...
remote: Counting objects: 262581, done.
remote: Compressing objects: 100% (65167/65167), done.
Receiving objects:  14% (37791/262581), 12.41 MiB | 1.43 MiB/s
```

Use the "dizzy" branch for poky

▸ Track the dizzy branch of the Yocto release to ensure we are using the current release.

```
tstrevey@spdxdev:~$ cd poky
tstrevey@spdxdev:~/poky$ git checkout -b dizzy origin/dizzy
Branch dizzy set up to track remote branch dizzy from origin.
Switched to a new branch 'dizzy'
tstrevey@spdxdev:~/poky$ 
```

Adding our project

- Copy the bitbake class and license configuration file from our repo to the poky directory.
- Within the poky/meta/conf directory change licenses.conf
- Edit SPDX_MANIFEST_DIR to point to the output directory.
- Edit DOSOCS_PATH to point to the main DoSOCS script.

```
tstrevey@spdxdev:~/poky$ cp ~/yocto-spdx/src/spdx.bbclass meta/classes
tstrevey@spdxdev:~/poky$ cp ~/yocto-spdx/src/licenses.conf meta/conf
```

```
# Output directory for completed SPDX documents
SPDX_MANIFEST_DIR = "/home/tstrevey/fossology_scans"

# Command line options to DoSOCS
DOSOCS_FLAGS = "--scanOption fossology --print json"

# Location of DoSOCS executable
DOSOCS_PATH = "/home/tstrevey/DOSOCS/DoSOCS/DoSOCS/DoSPDX.py"
-- INSERT --
```

Almost ready

- Run the Yocto project set up environment script

```
tstrevey@spdxdev:~/poky$ source oe-init-build-env
```

- Creates local.conf within the poky/build/conf directory
- Add spdx to the USER_CLASSES string

```
#
# Additional image features
#
# The following is a list of additional classes to use when building images which
# enable extra features. Some available options which can be included in this variable
# are:
#    - 'buildstats' collect build statistics
#    - 'image-mklibs' to reduce shared library files size for an image
#    - 'image-prelink' in order to prelink the filesystem image
#    - 'image-swab' to perform host system intrusion detection
# NOTE: if listing mklibs & prelink both, then make sure mklibs is before prelink
# NOTE: mklibs also needs to be explicitly enabled for a given image, see local.conf.extended
USER_CLASSES ?= "buildstats image-mklibs image-prelink spdx"
```

Starting a build

- Move back to the poky/build directory and start a build using bitbake.

```
tstrevey@spdxdev:~/poky/build$ bitbake -k core-image-minimal
Parsing recipes:   5% |###                                          | ETA:  00:00:32
```

- The Yocto build will take a while to complete.
- SPDX documents will be saved to the directory specified earlier.

Fossology_scans

```
tstrevey@spdxdev:~$ cd fossology_scans
tstrevey@spdxdev:~/fossology_scans$ ls
attr-native.spdx              gmp-native.spdx         mpfr-native.spdx      texinfo-dummy-native.spdx
autoconf-native.spdx          gnu-config-native.spdx  ncurses-native.spdx   unifdef-native.spdx
automake-native.spdx          gperf-native.spdx       pkgconfig-native.spdx xz-native.spdx
bison-native.spdx             libmpc-native.spdx      pseudo-native.spdx    zlib-native.spdx
flex-native.spdx              libtool-native.spdx     quilt-native.spdx
gettext-minimal-native.spdx   m4-native.spdx          sqlite3-native.spdx
tstrevey@spdxdev:~/fossology_scans$ vim attr-native.spdx
```