Dandified way to package management in Yocto Project

Jun 18th, 2019
Wang Mingyu, Fujitsu
wangmy@cn.fujitsu.com
whoami

- Wang Mingyu, Fujitsu Ltd.
- Embedded Linux Developer
- In-House Embedded Linux Distro of Fujitsu

- Our Distribution includes LTSI Kernel and is built with Yocto Project
- Our Distribution is used for
  - IVI, Server System Controller, Storage System, Network Equipment, Printer, etc.
Fujitsu’s contributions to Yocto community

- Data comes from yocto (2018-07-01 ~ 2019-07-01)

<table>
<thead>
<tr>
<th>Layers</th>
<th>Changesets</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 poky</td>
<td>46</td>
</tr>
<tr>
<td>2 oe-core</td>
<td>48</td>
</tr>
<tr>
<td>3 meta-oe</td>
<td>109</td>
</tr>
<tr>
<td>4 meta-cloud-services</td>
<td>50</td>
</tr>
<tr>
<td>5 bitbake</td>
<td>0</td>
</tr>
</tbody>
</table>

- Maintain meta-spdxscanner
Fujitsu’s contributions to Yocto community

- Data comes from yocto (2018-07-01 ~ 2019-07-01)

Developers with the most changesets

<table>
<thead>
<tr>
<th>No.</th>
<th>Our Developer</th>
<th>Changesets</th>
</tr>
</thead>
<tbody>
<tr>
<td>poky</td>
<td></td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>Zang Ruochen</td>
<td>25 (0.6%)</td>
</tr>
<tr>
<td>40</td>
<td>Hong Liu</td>
<td>11 (0.3%)</td>
</tr>
<tr>
<td>71</td>
<td>Lei Maohui</td>
<td>7 (0.2%)</td>
</tr>
<tr>
<td>oe-core</td>
<td></td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>Zang Ruochen</td>
<td>25 (0.4%)</td>
</tr>
<tr>
<td>50</td>
<td>Hong Liu</td>
<td>12 (0.2%)</td>
</tr>
<tr>
<td>63</td>
<td>Lei Maohui</td>
<td>7 (0.1%)</td>
</tr>
<tr>
<td>Meta-oe</td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>Zang Ruochen</td>
<td>72 (3.7%)</td>
</tr>
<tr>
<td>17</td>
<td>Hong Liu</td>
<td>23 (1.2%)</td>
</tr>
<tr>
<td>29</td>
<td>Lei Maohui</td>
<td>11 (0.6%)</td>
</tr>
<tr>
<td>meta-cloud-services</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>Zang Ruochen</td>
<td>26 (22.6%)</td>
</tr>
<tr>
<td>2</td>
<td>Hong Liu</td>
<td>20 (17.4%)</td>
</tr>
</tbody>
</table>
Agenda

What is dnf-plugin-tui
- Background
- Overview
- Features

How to use dnf-plugin-tui
- Getting started
- Demo

Contribution
- Oe-core
- Meta-oe

Future
- Support more package types
- Make TUI more user friendly
- Contribution plan
- ClearlyDefined and Cve-check-tool
What is dnf-plugin-tui

- Background
- Overview
- Features
Background

**DNF in Yocto**

- In Yocto, DNF is the default Package management to generate rootfs and SDK.

Image Generation

SDK Generation

Rootfs

Application Development SDK

Output Packages

Here is where DNF works

After packages were generated, The Yocto Build System will use DNF to generate the rootfs and SDK by installing Packages.
Issues we met

Typical case you meet when you try to modify your rootfs.

Not the proper rootfs you want?

Reconfigure the image bb file

Rebuild the image again

User Configuration

Yocto Build System

Rootfs

Application Development SDK

New Rootfs

Background
We redeveloped the DNF and Yocto to solve the issues mentioned above. We call it **dnf-plugin-tui** which means it can work on the Cross-Development environment in your host for package management as a plugin of DNF.

Git-Repo: [https://github.com/ubinux/dnf-plugin-tui.git](https://github.com/ubinux/dnf-plugin-tui.git)
Status: Support Yocto 2.6
Overview

Yocto Build server

Build
Deploy
Build
Deploy
Build
Deploy
Build
Deploy
......

Build and Deploy for Each Targets

Host

Packages

SDK
dnf-host

......

Build Once, Deploy Anywhere
Features

- Work as a plugin of dnf
- Support TUI
- Manage Bom
- Support config file
- Support multiple formats of image
- Support command line
Features(1/6)

Work as a plugin of DNF

- You can use the dnf-plugin-tui as a plugin of DNF

The code is independent of DNF
Support TUI

- You can use the Text-based UI to manage packages

```
Package Installer
| Select package |

[] acl
[] attr
[] base-files
[] base-passwd
[] base-passwd-update
[] bash
[] bash-bashbug
[] bash-completion
[] bash-completion-extra
[] bc
[] bsdcpio
[] bsdifd
[] busybox
[] busybox-hwclock
[] busybox-syslog
[] busybox-udhcpd
[] busybox-udhcpd
[] bzip2
[] ca-certificates
[] catchsegv
[] coreutils
[] cryptodev-linux
[] curl

All Packages [2364]  Installed Packages [0]  Selected Packages [0]

F1:select/unselect All  F2:Search  F3:Next  F4:Back  F5:Info  F9:Exit
```
Features (3/6)

Manage Bom

- Packages
- SRPM packages
- SPDX files
- Recipe files
- Build
- dnf-plugin-tui
  - Deploy
  - Upgrade
- Create archives
- Create archives
- Create archives

Copyright 2019 FUJITSU COMPUTER TECHNOLOGIES LIMITED
About Openchain

The Companies behind OpenChain

Adobe, arm, BOSCH, CISCO, COMCAST, facebook, FUJITSU, Google, HITACHI, Microsoft, Panasonic, Qualcomm, Siemens, Sony, TOSHIBA, TOYOTA, Uber, Western Digital.
Features (4/6)

Support config file

- Sample for config file

<table>
<thead>
<tr>
<th>Select install type</th>
</tr>
</thead>
<tbody>
<tr>
<td>New ---</td>
</tr>
</tbody>
</table>

- Like kernel config, selected package list can be saved and reused.

```
[ ] acl
[ ] attr
[ ] base-files
[*] base-passwd
[*] base-passwd-update
[ ] bash
[ ] bash-bashbug
[ ] bash-completion
[ ] bash-completion-extra
[*] bc
[ ] busybox
```

```
$ cat .config
base-passwd
bc
base-passwd-update
```

Load package list (Optional)

- host1
- host2
- host3
Features (5/6)

Support multiple formats of image

- Support more image format, e.g. EXT4, Squashfs, UBIFS and so on.
Support command line

- After init, you can use dnf tui command line to manage packages.

```bash
# dnf tui --command list
# dnf tui --command search <spec>
# dnf tui --command info <spec>
# dnf tui --command repolist
# dnf tui --command install <spec>
# dnf tui --command remove <spec>
# dnf tui --command upgrade <spec>
```

- New options
  - **--nosave**

```bash
# dnf tui --command install bash --nosave
# dnf tui --command remove bash --nosave
```

- **--pkg_list**

```bash
# dnf tui --command install --pkg_list pkg.list //Install packages that list in pkg.list
# dnf tui --command remove --pkg_list pkg.list //Remove packages that list in pkg.list
```
How to use dnf-plugin-tui

- Getting Started
- Demo
Add **dnf-plugin-tui** into toolchain

- First you need to add dnf-plugin-tui into your Yocto Project.

  ```bash
  $ git clone https://github.com/ubinux/dnf-plugin-tui.git
  $ ls dnf-plugin-tui/poky-patches/
  0001-meta-oe-patch-dnf-plugin-tui-new-recipe.patch
  0001-poky-2.6-Added-some-nativesdk-oss-for-nativesdk-dnf.patch
  0002-poky-2.6-Dnf-tui-plugin-patch-for-upstream-dnf.patch
  
  $ cd poky
  $ patch -p1 < XXX.patch
  ```

- Build meta-toolchain to get sdk that includes dnf-host

  ```bash
  $ bitbake meta-toolchain
  ```
Getting Started

How to use dnf-plugin-tui

- Then you can use dnf-plugin-tui on the Cross-Development environment.

```
$ sh poky-glibc-x86_64-meta-toolchain-i586-qemu-x86
```

```
$ . /opt/poky/2.6.1/environment-setup-i586-poky-linux
```

```
Prepare your rpm repo, then:
$ dnf tui --init
```

Use TUI
```
$ dnf tui
```

Install the toolchain
Source the toolchain
Init the environment
dnf-host tui
Complete!
Prepare rootfs
progress: [###]100%
Put rootfs to destination
progress: [###]100%
Do you like to keep the tarball of rootfs. [Y/n]?

The tarball is /home/teset/rootfs-201907101143.tar.bz2
Target dir is /opt/ubq/devkit/arm64/rootfs-201907101143
Contribution

- Oe-core
- Meta-oe
Contribution

- **oe-core**
  commit content: Make dnf-nativesdk work
  commit status: dnf-nativesdk is accepted

- **meta-oe**
  commit content: Add recipe of dnf-plugin-tui
  commit status: in discussion

Get the Source Code from github

Add meta-oe to build system

Bitbake the SDK
Future

- Make TUI more user friendly
- Contribution plan
- ClearlyDefined and Cve-check-tool
Future

Make TUI more user friendly

[ ] Bash
[*] Base-files
[ ] Base-passwd
......
[ ] Coreutils
[*] Util-linux
......
[ ] Httpd
......

[*] Base → [ ] Bash
  [*] Base-files
  [ ] Base-passwd
  ......

[*] Utils → [ ] Coreutils
  [*] Util-linux
  ......

[ ] Web Server → [ ] Httpd
  ......
Future

Contribution plan

- Continuous development.
- Release dnf-plugin-tui follow the step of Yocto.
About ClearlyDefined

ClearlyDefined and our parent organization, the Open Source Initiative, are on a mission to help FOSS projects thrive by being, well, clearly defined. Lack of clarity around licenses and security vulnerabilities reduces engagement — that means fewer users, fewer contributors and a smaller community.

**ClearlyDescribed**
Knowing simple things like the source location for the open source component you are using enables contribution of docs, bug fixes, or new features. It also inspires confidence by enabling IP and security code scans and source code archiving and disclosure. Round that out with project and issue tracking site info, and you have a sound basis for engagement. Learn more...

**ClearlyLicensed**
Defining and knowing the license for an open source component is essential to a successful partnership. Communities choose a license with terms they like. ClearlyDefined helps clarify that choice and enables consumers to follow the terms by identifying key data such as license set, attribution parties, and code location. Learn more...

**ClearlySecure**
Teams working hard to create quality, secure free and open source components need a simple way of recording security issues they find and fix. Bug report and pull requests are great. CVEs and global notifications are even better. It can still be hard to relate that data to the components you use. ClearlyDefined gives communities a security forum that builds confidence and makes for even more collaboration. Learn more...

**Package information**

**Manage spdx file**

**Manage result of cve-check-tool**
Cve-check-tool

- Manage the cve result file of packages installed by dnf-tui-plugin.
  Cve result file is the result of checking packages by cve-check-tool.

```
PACKAGE NAME: db
PACKAGE VERSION: 5.3.28
CVE: CVE-2016-0682
CVE STATUS: Unpatched
CVE SUMMARY: Unspecified vulnerability in the DataStore component in Oracle Berkeley DB
11.2.5.0.32, 11.2.5.1.29, 11.2.5.2.42, 11.2.5.3.28, 12.1.6.0.35, and 12.1.6.1.26 allows local users to affect confidentiality, integrity, and availability via unknown vectors, a different vulnerability than CVE-2016-0689, CVE-2016-0692, CVE-2016-0694, and CVE-2016-3418.
CVSS v2 BASE SCORE: 6.9
VECTOR: LOCAL
MORE INFORMATION: https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2016-0682
```
Any Questions?

wangmy@cn.fujitsu.com
shaping tomorrow with you