LTSI Project update

Long Term Support Initiative

Tsugikazu SHIBATA, NEC
12, October 2016
Embedded Linux Conference Europe
Maritim Hotel, Berlin
Who am I

• Tsugikazu SHIBATA, NEC
• Founder and project lead of Long Term Support Initiative (LTSI) that will help industry people to use Long Term Stable kernel with helpful activities.
• Involved Linux kernel community since 2.4 and also a board member of Linux Foundation
Linux is running everywhere

• Linux is running for *multiple use cases*
  – NYSE, London, Tokyo Stock Exchange
  – Amazon, Google, Facebook, Twitter
  – Android, TV, Camera, Router, Car navigation

• Supports *multiple architectures*
  – x86, arm, s390, ia64, mips, parisc, sparc, sh ...

• All those come from *Single Source code* tree
Developed by the community

• 1500 developer, 200 companies every release
• Yearly 2.2Mlines of code increased, 4900 files
  – Continue to increase developers
• 25 Years of history
• Maintainers have great skill to manage the subsystem and professional knowledge of its area of technologies
Status of Latest Linux Kernel

• Latest released Kernel : 4.8
  – Released: October 2\textsuperscript{nd}, 2016
  – Lines of code : 22,071,048 (+350,053)
  – Files : 55,476 (+1100)

• Current Stable Kernel: 4.8.1

• Current development kernel: \texttt{merge window for 4.9-rc1}
Linux development policy

• Upstream is only the place to accept the patches
  – Reviewed by skilled maintainer
  – Tested with other proposals to confirm no conflicts
  – Follow the development process
Linux Development process

- Just after the release of 4.n, two weeks of merge window will be opened for proposal of new features
- After 2 weeks of merge window, -rc1 will be released and the stabilization will be started
- 4.n+1 will be released when it becomes reasonably stable by some of -rcX released
Kernel release cycle

- Release cycle of Linux kernel is about 65 days

<table>
<thead>
<tr>
<th>Version</th>
<th>Release</th>
<th>Rel. span</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.10</td>
<td>2013-6-30</td>
<td>63</td>
</tr>
<tr>
<td>3.11</td>
<td>2013-9-2</td>
<td>64</td>
</tr>
<tr>
<td>3.12</td>
<td>2013-11-15</td>
<td>74</td>
</tr>
<tr>
<td>3.13</td>
<td>2014-1-21</td>
<td>67</td>
</tr>
<tr>
<td>3.14</td>
<td>2014-3-30</td>
<td>68</td>
</tr>
<tr>
<td>3.15</td>
<td>2014-6-8</td>
<td>70</td>
</tr>
<tr>
<td>3.16</td>
<td>2014-8-3</td>
<td>56</td>
</tr>
<tr>
<td>3.17</td>
<td>2014-10-5</td>
<td>63</td>
</tr>
<tr>
<td>3.18</td>
<td>2014-12-7</td>
<td>63</td>
</tr>
<tr>
<td>3.19</td>
<td>2015-2-9</td>
<td>64</td>
</tr>
<tr>
<td>4.0</td>
<td>2015-4-12</td>
<td>62</td>
</tr>
<tr>
<td>4.1</td>
<td>2015-6-22</td>
<td>71</td>
</tr>
<tr>
<td>4.2</td>
<td>2015-8-30</td>
<td>69</td>
</tr>
<tr>
<td>4.3</td>
<td>2015-11-2</td>
<td>64</td>
</tr>
<tr>
<td>4.4</td>
<td>2016-1-10</td>
<td>68</td>
</tr>
<tr>
<td>4.5</td>
<td>2016-3-14</td>
<td>64</td>
</tr>
<tr>
<td>4.6</td>
<td>2016-5-15</td>
<td>63</td>
</tr>
<tr>
<td>4.7</td>
<td>2016-7-24</td>
<td>70</td>
</tr>
<tr>
<td>4.8</td>
<td>2016-10-2</td>
<td>70</td>
</tr>
</tbody>
</table>

Yearly 5-6 chance to merge
Stable kernel release

- Recommended branch for users who want the most recent stable kernel
- 3 part version like 4.n.m
- Contain small and critical fixes for security problems or significant regressions discovered in a latest development version
- Becomes End Of Life when next stable kernel were released
Status of Latest Linux Kernel Again

• Latest released Kernel : 4.8
• Current Stable Kernel: 4.8.1
• Current development kernel: in the merge window for 4.9-rc1
Stable_kernel_rules.txt

- Strict rule to back port from latest version

- Published since 2006.

- It must be obviously correct and tested.
- It cannot be bigger than 100 lines, with context.
- It must fix only one thing.
- It must fix a real bug that bothers people.
- It must fix a problem that causes a build error, an oops, a hang, data corruption, a real security issue, in short, something critical.
- Serious issues as reported by a user may also be considered.
- New device IDs and quirks are also accepted.
- No "theoretical race condition" issues, unless an explanation of how the race can be exploited is also provided.
- It cannot contain any "trivial" fixes in it.
- It or an equivalent fix must already exist in Linus' tree (upstream).
LTS: LongTerm Stable Kernel

- Kernel tree continue to back port bug and Security fixes
- Extended maintenance period for stable kernel
- Pick one version per year and maintain 2 years
Why LTS?

• Only the tree to get fixes from the community
• In the real use case, tested/confirmed kernel is important, less important for new features
• Fixes will be released # of times and should be applied frequently, Security/Bug fixes are being more important
• Bugs found in LTS should be reported and fixed in upstream
## Current LTS versions

<table>
<thead>
<tr>
<th>Version</th>
<th>Maintainer</th>
<th>Released</th>
<th>Projected EOL</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.9</td>
<td>Greg Kroah-Hartman</td>
<td>Not yet</td>
<td></td>
</tr>
<tr>
<td>4.4</td>
<td>Greg Kroah-Hartman</td>
<td>2016-01-10</td>
<td>Feb, 2018</td>
</tr>
<tr>
<td>4.1</td>
<td>Sasha Levin</td>
<td>2015-06-21</td>
<td>Sep, 2017</td>
</tr>
<tr>
<td>3.18</td>
<td>Sasha Levin</td>
<td>2014-12-07</td>
<td>Jan, 2017</td>
</tr>
<tr>
<td>3.16</td>
<td>Ben Hutchings</td>
<td>2014-08-03</td>
<td>Apr, 2020</td>
</tr>
<tr>
<td>3.12</td>
<td>Jiri Slaby</td>
<td>2013-11-03</td>
<td>Jan, 2017</td>
</tr>
<tr>
<td>3.10</td>
<td>Willy Tarreau</td>
<td>2013-06-30</td>
<td>Oct, 2017</td>
</tr>
<tr>
<td>3.4</td>
<td>Li Zefan</td>
<td>2012-05-20</td>
<td>Sep, 2016</td>
</tr>
<tr>
<td>3.2</td>
<td>Ben Hutchings</td>
<td>2012-01-04</td>
<td>May, 2018</td>
</tr>
</tbody>
</table>
LTS includes large number of fixes

- 600 – 700 fixes included in a Stable release
- LTS include several thousands of fixes

<table>
<thead>
<tr>
<th>Version</th>
<th>From</th>
<th>To</th>
<th># of commits</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.0</td>
<td>3.0.101</td>
<td>3953 (EOL)</td>
<td></td>
</tr>
<tr>
<td>3.1</td>
<td>3.1.10</td>
<td>695 (EOL)</td>
<td></td>
</tr>
<tr>
<td>3.2</td>
<td>3.2.82</td>
<td>7165</td>
<td></td>
</tr>
<tr>
<td>3.3</td>
<td>3.3.8</td>
<td>698 (EOL)</td>
<td></td>
</tr>
<tr>
<td>3.4</td>
<td>3.4.112</td>
<td>5797</td>
<td></td>
</tr>
<tr>
<td>3.5</td>
<td>3.5.7</td>
<td>816 (EOL)</td>
<td></td>
</tr>
<tr>
<td>3.6</td>
<td>3.6.11</td>
<td>757 (EOL)</td>
<td></td>
</tr>
<tr>
<td>3.7</td>
<td>3.7.10</td>
<td>718 (EOL)</td>
<td></td>
</tr>
<tr>
<td>3.8</td>
<td>3.8.13</td>
<td>996 (EOL)</td>
<td></td>
</tr>
<tr>
<td>3.9</td>
<td>3.9.11</td>
<td>746 (EOL)</td>
<td></td>
</tr>
</tbody>
</table>

As of 2016/10/9
LTSI Status
What is LTSI

• Open Source community to create and maintain Linux kernel for long term
  – Based on LTS
  – Add another chance to include further patches on top of LTS
  – Same lifetime with LTS (yearly release and 2 years life time)

• Industry party to share practice and experience among the companies
LTSI includes LTS

**LTSI**
- Add vendor required features
- Share status, info, problem among industry people
- Huge testing by contributors
- Auto test frame work
- Provide help to developer for upstream

**LTS**
- Release 1 version / year, Maintain 2 years
- Frequently and large number of bug /security fixes
History of LTSI

• Established 2011 – 5 yeas now
  – Started for stable Kernel for Android
• Integrated by Yocto (2012, May)
• Have had a workshop/session to share information and discuss issue among industry people
• Released yearly basis; 3.0, 3.4, 3.10, 3.14, 4.1
Shape of LTSI Project

- Small staff to coordinate workshop, session at LF conference
- Maintainer: Greg Kroah-Hartman, Fellow of Linux Foundation
- Working with upstream Linux Community
- Keeping neutral position to be able to use for variety of use case
• Preparation: 4-5 month (~2 of upstream release)
• Merge window: 2 month to propose additional patches. Self contained or later upstream features
• Validation: a month or more. All the contributor must validate and report back
LTSI as a part of Distro

• Yocto project has a normal option to use LTSI
  – You can create your own distribution with LTSI
  – Yocto is doing their own tests for the distribution with LTSI. Both build distro and run it.

• Many of report/blog in the internet using Yocto and LTSI
Fuego: Auto test package

- Jenkins based auto test packages (using Docker)
  - Distributed under 3-clause BSD license
  - Over 50 test packages included with Web based monitor
- Got nice new name “Fuego” by the community
- Already used by number of users
- This is a referenced implementation but works good
Use Case: Automotive Grade Linux

• Developing reference implementation called UCB (Unified Code Base) distribution v1.0 using LTSI 3.14 kernel
  – AGL/UCB is using Yocto and LTSI

• Have had a demonstration at CES 2016 Las Vegas with announcement

• V2.0 Brilliant Blowfish was released July 12

• V3.0 is planned to be released Jan. 2017
Use Case: Civil Infrastructure Platform

• Established Feb 2016, by Toshiba, Hitachi, Siemens and others to create reference distribution for Civil infrastructure

• One of major requirement includes Super Long Term Support: 10-15 years

• Planning to take over maintenance of LTSI after EOL
Q: Plan of this year’s LTSI?

• Kernel Summit 2015 had discussion about LTS and decided to be 4.4
• Greg stated that 4.9 will be this year’s LTS
• There were number of discussion at K-summit ML and maybe move to 4.8 or 4.10 if people send unready patches to upstream
• Also, Discussion about upstream first happened
LTSI Schedule expectation

• Kernel Summit 2016 will be the time to decide LTS version

LTSI release schedule          7M, 2017                9E, 2017
Expectation
Conclusion

• LTSI was started to fill the gap between community and industry but still there is the gap
  – We will continue our activity to discuss both side to better align each other

• Well Eco system support : Yocto and Fuego

• Strong use case is on going

• Why don’t you join LTSI?
  – By joining the community, you will be able to share best practice
  – To get useful information for stable kernel
THANK YOU

LTSI Workshop:
Wednesday October 12, 10AM- @Salon 17, Riga
You can participate LTSI

- Follow on Twitter account:
  @LinuxLTSI
- Web:
  http://ltsi.linuxfoundation.org
- Mailing list:
  https://lists.linuxfoundation.org/mailman/listinfo/ltsi-dev
- Git tree:
  http://git.linuxfoundation.org/?p=ltsi-kernel.git;a=summary