Activities of Super Long Term Support Kernel Workgroup in Civil Infrastructure Platform Project

Pavel Machek
CIP Kernel Maintainer, Denx

SZ Lin (林上智)
CIP Kernel Working Group Chairperson and TSC Representative, Moxa Inc.

ELCE 2019, France, 29th Oct.
About Us

- Pavel Machek <pavel@denx.de>
  - Cooperating with Denx
  - Kernel hacker
  - 4096R/92DFCE96
    - 4FA7 9EEF FCD4 C44F C585 B8C7 C060 2241 92DF CE96

- SZ Lin (林上智) <sz.lin@moxa.com>
  - Working for Moxa Inc.
  - Debian developer
  - Contribute to Linux and other OSS projects
  - 4096R/9561F3F9
    - 178F 8338 B314 01E3 04FC 44BA A959 B38A 9561 F3F9
The key challenges

- Apply IoT concepts to industrial systems.
- Ensure quality and longevity of products.
- Keep millions of connected systems secure.

**Industrial grade**
- Reliability
- Functional Safety
- Real-time capabilities

**Sustainability**
- Product life-cycles of decades
- Backwards compatibility
- Standards

**Security**
- Security & vulnerability management
- Firmware updates
- Minimize risk of regressions

Embedded Linux Conference Europe 2019
CIP is the Solution

Industrial grade
- Reliability
- Functional Safety
- Real-time capabilities

Sustainability
- Product life-cycles of decades
- Backwards compatibility
- Standards

Security
- Security & vulnerability management
- Firmware updates
- Minimize risk of regressions
CIP is the Solution

Establishes an “Open Source Base Layer (OSBL)”

- CIP Core packages (tens)
- CIP kernel (10+ years maintenance, based on LTS kernels)
- company-specific middleware and applications (hundreds)
The Scope of CIP

**User space**
- App container infrastructure (mid-term)
- App Framework (optionally, mid-term)

**Middleware/Libraries**
- Domain Specific communication (e.g. OPC UA)
- Shared config. & logging
- Multimedia

**Linux Kernel**
- CIP Core Packages
- Safe & Secure Update
- Monitoring
- Security
- Real-time support
- Real-time / safe virtualization

**Tools**
- Build environment (e.g. bitbake, dpkg)
- Test automation
- Tracing & reporting tools
- Configuration management
- Device management (update, download)
- Application life-cycle management

**Concepts**
- Functional safety architecture/strategy, including compliance w/ standards (e.g., NERC CIP, IEC61508)
- Long-term support Strategy: security patch management
- Standardization collaborative effort with others
- License clearing
- Export Control Classification

**On-device software stack**

**Product development and maintenance**

**Super Long Term Supported Kernel (STLS)**

**CIP Core Packages**
- Security

**Tools**
- Build environment (e.g. bitbake, dpkg)
- Test automation
- Tracing & reporting tools
- Configuration management
- Device management (update, download)
- Application life-cycle management

**Concepts**
- Functional safety architecture/strategy, including compliance w/ standards (e.g., NERC CIP, IEC61508)
- Long-term support Strategy: security patch management
- Standardization collaborative effort with others
- License clearing
- Export Control Classification

**On-device software stack**
Super Long Term Support Kernel Workgroup

• The first action taken by the CIP project is to select and maintain Linux kernels for very long time (10+ years).

• Applying the PREEMPT_RT patch to CIP Kernel, then maintaining as CIP-RT.

CIP Projects and its scopes

1. SLTS kernel
2. Real-time
3. Testing
4. CIP Core
5. Security WG(*)
6. Software update WG

(*): Workgroup

<table>
<thead>
<tr>
<th>Industrial grade</th>
<th>Sustainability</th>
<th>Security</th>
</tr>
</thead>
<tbody>
<tr>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
</tbody>
</table>

Embedded Linux Conference Europe 2019
Policy and Progress
<table>
<thead>
<tr>
<th>Version</th>
<th>Maintainer</th>
<th>Released</th>
<th>Projected EOL</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.4</td>
<td>Greg Kroah-Hartman &amp; Sasha Levin</td>
<td>2019-XX-XX</td>
<td>Dec, 2021</td>
</tr>
<tr>
<td>4.14</td>
<td>Greg Kroah-Hartman &amp; Sasha Levin</td>
<td>2017-11-12</td>
<td>Jan, 2024</td>
</tr>
<tr>
<td>4.9</td>
<td>Greg Kroah-Hartman &amp; Sasha Levin</td>
<td>2016-12-11</td>
<td>Jan, 2023</td>
</tr>
<tr>
<td>4.4</td>
<td>Greg Kroah-Hartman &amp; Sasha Levin</td>
<td>2016-01-10</td>
<td>Feb, 2022</td>
</tr>
<tr>
<td>3.16</td>
<td>Ben Hutchings</td>
<td>2014-08-03</td>
<td>Apr, 2020</td>
</tr>
</tbody>
</table>
CIP SLTS Kernel Development (Upstream First Development)

• **Goal**
  - Providing CIP kernels with more than 10 years maintenance period
    - **Super Long Time Stable kernel**

• **Status**
  - LTS review process participation
  - CIP SLTS kernels release
    - 4.4.196-cip38
    - 4.19.78-cip12
  - CIP kernel CVE tracker
  - CIP kernel failed patches tracker

![Diagram](https://example.com/diagram.png)

1. Upstreaming
2. Merge
3. Integrate and test

Mainline / LTS

LTS kernel

Mentor / Maintainer

Developers

CIP kernel team

Feature mainlining

Review and test results/ Fixes

CIP SLTS kernel
Introduction to CIP Kernel Team Member

• **Chairperson**
  • SZ Lin (林上智)

• **Maintainer**
  • Nobuhiro Iwamatsu
  • Pavel Machek

• **Mentor**
  • Ben Hutchings
<table>
<thead>
<tr>
<th>Version</th>
<th>Maintainer</th>
<th>First Release</th>
<th>Latest Release</th>
<th>Projected EOL</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.19</td>
<td>Nobuhiro Iwamatsu &amp; Pavel Machek</td>
<td>2019-01-11</td>
<td>2019-10-12</td>
<td>2029+</td>
</tr>
<tr>
<td></td>
<td></td>
<td>v4.19.13-cip1</td>
<td>v4.19.78-cip12</td>
<td></td>
</tr>
<tr>
<td>4.19-rt</td>
<td>Pavel Machek</td>
<td>2019-01-11</td>
<td>2019-10-02</td>
<td>2029+</td>
</tr>
<tr>
<td></td>
<td></td>
<td>v4.19.13-cip1-rt1</td>
<td>v4.19.72-cip10-rt3</td>
<td></td>
</tr>
<tr>
<td>4.4</td>
<td>Nobuhiro Iwamatsu &amp; Pavel Machek</td>
<td>2017-01-17</td>
<td>2019-10-12</td>
<td>2027+</td>
</tr>
<tr>
<td></td>
<td></td>
<td>v4.4.42-cip1</td>
<td>v4.4.196-cip38</td>
<td></td>
</tr>
<tr>
<td>4.4-rt</td>
<td>Pavel Machek</td>
<td>2017-11-16</td>
<td>2019-10-02</td>
<td>2027+</td>
</tr>
<tr>
<td></td>
<td></td>
<td>v4.4.75-cip6-rt1</td>
<td>v4.4.190-cip36-rt25</td>
<td></td>
</tr>
</tbody>
</table>
CIP SLTS Kernel Development

Maintenance Policy

• [https://wiki.linuxfoundation.org/civilinfrastructureplatform/cipkernelmaintenance](https://wiki.linuxfoundation.org/civilinfrastructureplatform/cipkernelmaintenance)

• Follow the stable kernel development rule as the basis

• Validation will be done by CIP test infrastructure and/or members

• Feature backports from CIP members are acceptable
  • All features has to be in upstream kernel before backport to CIP kernel
  • CIP has “Upstream first” policy

• The CIP Project uses the Linux Foundation Developer Certificate of Origin (DCO)
Out-of-tree drivers

• In general, all out-of-tree drivers are unsupported by CIP

• Users can use CIP kernel with out-of-tree drivers
  • If a bug is found in such a modified kernel, users will first demonstrate that it exists in the CIP kernel source release in order for the CIP maintainers to act on it.
CIP SLTS Kernel Development

Mainline

4.4

4.19

Feature backports

Stable 4.4.y

CIP SLTS (linux-4.4.y-cip)

Maintained by Iwamatsu and Pavel

Backported patches

Stable 4.19.y

CIP SLTS (linux-4.19.y-cip)

EOL

EOL
The Sources of CIP Patches

- **Stable patches**
  - Security issue
  - Bug fixes

- **Backported patches**
  - Security issue
  - Bug fixes
  - Feature backports

- **CIP patches**
Patches Review
Stable Patches Review Participation

Stable patches

Review Stable Patches

# Stable Kernel Patches Review Status
Please list your name and review result below the patch item

* UR: Under Review
* ACK: Acknowledge (if the patch is accepted)
* TBB: To be backported (if other patches should be also backported)
* NAK: Negative acknowledgment (if the patch is rejected, please list the reason)
* IGN: Patch was not reviewed as it is out of scope for CIP project
Stable Patches Review Participation

Kernel 4.4

```c
## v4.4.196
- c61eb668f2c Linux 4.4.196
ACK: iwamatsu
- 2e48675890fabd: fix attr checks in netlink interface
ACK: iwamatsu
- ac375073bb39 smack: use GFP_NOFS while holding inode_smack::smk_lock
```

Kernel 4.19

```c
## v4.19.77
- 6cad9d0 Linux 4.19.77
- 2c60da9 drm/amd/display: Restore backlight brightness after system resume
ACK: pavell
- 4d8bbdf7 mm/compaction.c: clear total(_migrate,free)_scanned before scanning a new zone
ACK: pavell
- 5bead06 fuse: fix deadlock with aio poll and fuse_qnode::waitq.lock
UR: pavell -- bad according to Eric Biggers
- bbe3e20 md/raid0: avoid RAID0 data corruption due to layout confusion.
UR: pavell -- should this be per-array not per-module?
- 4290a9e CIFS: fix oplock handling for SMB 2.1+ protocols
ACK: pavell
- a3a1508 CIFS: fix max ea value size
```
Stable Patches Review Participation

Reviewed by Pavel for 4.19-stable

---

```
a/drivers/infiniband/core/restrack.c
+++ b/drivers/infiniband/core/restrack.c
@@ -209,7 +209,7 @@ void rdma_restrack_del(struct rdma_restr
     struct ib_device *dev;
>
if (!res_valid)
-    return;
+    goto out;
>
dev = res_to_dev(res);
if (!dev)
    return;

This test does return, does it need to go through ‘goto out’, too? (I see it should not happen, but...)
```

```
-222,8 +222,10 @@ void rdma_restrack_del(struct rdma_restr
     down_write(&dev_res.rwsem);
     hash_del(&res_node);
     res_valid = false;
+    up_write(&dev_res.rwsem);
+    +out:
+    if (res_task)
+        put_task_struct(res_task);
-    - up_write(&dev->res.rwsem);
+    }
```

Mainline says res_task = NULL is needed there, see fe9bc1644918aa1d.

Best regards,

Pavel

Reviewed by Pavel for 4.19-stable

```
Difference Between -stable Kernel Rules and Rules Imposed in Practice

a) It or an equivalent fix must already exist in Linus' tree (upstream).
   Enforced. Strong preference is given to merging exactly the same patch as in upstream.

b) It must be obviously correct and tested.
   Preference is given to a): buggy patch is merged, then fix is merged as followup.

c) It must fix a real bug that bothers people (not a, "This could be a problem..." type thing).
   Anything that looks like a bugfix is merged, includes trivial memory leaks that leak few bytes per boot and missing of_node_put().

d) It must fix a problem that causes a build error (...), an oops, a hang, data corruption, a real security issue, or some "oh, that's not good" issue. In short, something critical.
   Build-time warnings, run-time warnings, confusing printk messages and loglevel changes for printk messages are applied.

e) It cannot contain any "trivial" fixes in it (spelling changes, whitespace cleanups, etc).
   Certainly not enforced. There is preference to taking patches from mainline with no changes.
CIP Kernel Team Contribution to Upstream Statistics (Retrieved 25th, Oct 2019)

Email discussion/ review
Patch commits
CIP Members Patches Review Participation

Kernel 4.4

[cip-dev] [PATCH 4.4-cip] ARM: dts: socfpga: Rename socfpga_cyclone5_de0_{sockit, nano_soc}  
Jan Kiszka

[cip-dev] [PATCH 4.4-cip] ARM: dts: socfpga: Rename socfpga_cyclone5_de0_{sockit, nano_soc}  
Nobuhiro Iwamatsu

Kernel 4.19

[cip-dev] [PATCH 4.19.y-cip 0/4] Add USB support  
Biju Das

[cip-dev] [PATCH 4.19.y-cip 0/4] Add USB support  
Pavel Machek

[cip-dev] [PATCH 4.19.y-cip 0/4] Add USB support  
Pavel Machek
Feature Backported Patches Statistics in CIP Kernel

![Bar chart showing patch commits for CIP kernel 4.4 and 4.19](chart.png)

- CIP kernel 4.4: 617 patch commits
- CIP kernel 4.19: 436 patch commits
CIP SLTS Real-time Support

CIP SLTS+PREEMPT_RT (will be separately maintained by CIP members)
- CIP kernel tree based on linux-stable-rt and patches from CIP SLTS
- Validation will be done by CIP
CIP SLTS Real-time Support

• CIP has become a Gold Member of the Real Time Linux Project

• What’s next
  • Work together with the RTL Project

• More information
  • https://wiki.linuxfoundation.org/realtime/rtl/start
# CIP SLTS Kernel Release Policy

<table>
<thead>
<tr>
<th>Release regularly</th>
<th>Release on demand</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release <strong>4.19 twice a month and 4.4 once a month</strong> (Effective June, 2019)</td>
<td>It depends on critical bug/ security fix</td>
</tr>
<tr>
<td>- Kernel 4.19</td>
<td></td>
</tr>
<tr>
<td>- second and fourth Fridays of the month</td>
<td></td>
</tr>
<tr>
<td>- Kernel 4.4</td>
<td></td>
</tr>
<tr>
<td>- second Friday of the month</td>
<td></td>
</tr>
<tr>
<td>Release <strong>4.19-rt once a month and 4.4-rt once every two months</strong> (Effective Nov, 2019)</td>
<td>Ditto</td>
</tr>
</tbody>
</table>

**Note**: Difficult to estimate actual release date because of number of patches depends on each stable release
CIP Kernel and Real-time Kernel Release Statistics

<table>
<thead>
<tr>
<th>Year</th>
<th>v4.4-cip</th>
<th>v4.4-cip-rt</th>
<th>v4.19-cip</th>
<th>v4.19-cip-rt</th>
</tr>
</thead>
<tbody>
<tr>
<td>2017</td>
<td>15</td>
<td>3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2018</td>
<td>14</td>
<td>17</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2019 (Retrieved 27th, Oct)</td>
<td>9</td>
<td>5</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>Total (Retrieved 27th, Oct)</td>
<td>38</td>
<td>25</td>
<td>12</td>
<td>3</td>
</tr>
<tr>
<td>Total estimated in 2019</td>
<td>42</td>
<td>26</td>
<td>16</td>
<td>5</td>
</tr>
</tbody>
</table>

(Retrieved 27th, Oct)
Introduction to "cip-kernel-sec"

- This project tracks the status of security issues, identified by CVE ID, in mainline, stable, and other configured branches.
The Maintenance Scope of "cip-kernel-sec" – “cip-kernel-config”

• The security issues are determined to be fixed base on kernel configurations provided by CIP members
CVE Issue Format - Supported by CIP

Issue Format - YAML
CVE Issue Format – Unsupported by CIP

description: 'IB/mlx5: Fix leaking stack memory to userspace'
references:
- https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2018-20855
- https://git.kernel.org/cgit/linux/kernel/git/torvalds/linux.git/commit/?id=0625b4ba1a5d4703c7fb01c497bd6c156908af00
- https://github.com/torvalds/linux/commit/0625b4ba1a5d4703c7fb01c497bd6c156908af00
comments:
  Debian-bwh: |
  Introduced in Linux 4.17 by commit 41d902cb7c32 "RDMA/mlx5: Fix
definition of mlx5_ib_create_qp Resp".
introduced-by:
  mainline: [41d902cb7c326d711674977763c4b30df87611bc]
fixed-by:
  mainline: [0625b4ba1a5d4703c7fb01c497bd6c156908af00]
ignore:
  linux-4.19.y-cip: No member enables the mlx5_ib driver
  linux-4.19.y-cip-rt: No member enables the mlx5_ib driver
  linux-4.4.y-cip: No member enables the mlx5_ib driver
  linux-4.4.y-cip-rt: No member enables the mlx5_ib driver
Introduction to "classify-failed-patches"

- This project tracks the status of failed patches, and classifies patches into “applied” and “ToApply” types.
Introduction to "classify-failed-patches"

<table>
<thead>
<tr>
<th>Applied patches</th>
</tr>
</thead>
<tbody>
<tr>
<td>[APPLIED] arm64: Disable unhandled signal log messages by default 56b579d20f9b9a5d503f8e5f7b74c1ee18f8fed9863 arm64: Disable unhandled signal log messages by default</td>
</tr>
<tr>
<td>[APPLIED] ARC: hide unused function umx_hdr_alloc 4d28512b7fca8445eb6f65e6800c2033cf0619640000 ARC: hide unused function umx_hdr_alloc</td>
</tr>
<tr>
<td>[APPLIED] btrfs: Ensure replaced device doesn't have pending chunk allocation 986543cf50c8a3681e6c44cac2d4c588e25ab34 btrfs: Ensure replaced device doesn't have pending chunk allocation</td>
</tr>
<tr>
<td>[APPLIED] btrfs: Ensure replaced device doesn't have pending chunk allocation 986543cf50c8a3681e6c44cac2d4c588e25ab34 btrfs: Ensure replaced device doesn't have pending chunk allocation</td>
</tr>
<tr>
<td>[APPLIED] btrfs: Ensure replaced device doesn't have pending chunk allocation 986543cf50c8a3681e6c44cac2d4c588e25ab34 btrfs: Ensure replaced device doesn't have pending chunk allocation</td>
</tr>
<tr>
<td>[APPLIED] btrfs: Ensure replaced device doesn't have pending chunk allocation 986543cf50c8a3681e6c44cac2d4c588e25ab34 btrfs: Ensure replaced device doesn't have pending chunk allocation</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>To be Applied Patches</th>
</tr>
</thead>
<tbody>
<tr>
<td>[TOAPPLY] inet: update the IP ID generation algorithm to patch 355b98853789b646ed7ad801a619ff898471b92 standards.</td>
</tr>
<tr>
<td>[TOAPPLY] scsi: ufs: Fix RX_TERMINATION_FORCE_ENABLE define value</td>
</tr>
<tr>
<td>[TOAPPLY] inet: update the IP ID generation algorithm to patch 355b98853789b646ed7ad801a619ff898471b92 standards.</td>
</tr>
<tr>
<td>[TOAPPLY] IB/hfi: Failed to drain send queue when QP is put into error state</td>
</tr>
<tr>
<td>[TOAPPLY] arm64: mm: Ensure tail of unaligned initrd is reserved</td>
</tr>
<tr>
<td>[TOAPPLY] fs/proc/task_mmu.c: fix uninitialized variable warning</td>
</tr>
<tr>
<td>[TOAPPLY] tlp: Fix the type of the return value in calc_tpm2_event_size()</td>
</tr>
<tr>
<td>[TOAPPLY] block: bio_map_user_iov should not be limited to BIO_MAX_PAGES</td>
</tr>
<tr>
<td>[TOAPPLY] clk: ingenic/jz4725b: Fix parent of pixel clock</td>
</tr>
<tr>
<td>[TOAPPLY] i2c-pixi4: Add Hygon Elyana SMBus support</td>
</tr>
<tr>
<td>[TOAPPLY] tty: serial core, add -&gt;install</td>
</tr>
</tbody>
</table>
Testing
CIP Testing Talk at CIP Mini Summit

- Thursday in CIP Mini Summit
  - 8:00 – 13:00 @ Lyon Convention Centre

Testing Architecture Overview
Summary

• Routine tasks
  • The frequency of CIP kernel release
    • The kernel of
      • 4.4 will be released at least once and 4.19 will be released at least twice every month
      • 4.4-rt will be released at least once every two months and 4.19-rt will be released at least twice a month
    • Failed patch tracker
    • Linux kernel CVE tracker

• Occasional tasks
  • Build up kernel and rt-kernel testing
  • Define and update the wiki for kernel maintenance scope
Weekly Regular Online Meeting

• CIP IRC weekly meeting – Every Thursday UTC (GMT) 09:00

<table>
<thead>
<tr>
<th>US-West</th>
<th>US-East</th>
<th>UK</th>
<th>DE</th>
<th>TW</th>
<th>JP</th>
</tr>
</thead>
<tbody>
<tr>
<td>02:00</td>
<td>05:00</td>
<td>09:00</td>
<td>10:00</td>
<td>17:00</td>
<td>18:00</td>
</tr>
</tbody>
</table>

• Channel:
  * irc:chat.freenode.net:6667/cip

• The meeting will take 30 min although it can be extended to an hour if it makes sense and those involved in the topics can stay. Otherwise, the topic will be taken offline or in the next meeting.
CIP Kernel Workgroup Repository

• CIP Linux kernel & real-time kernel
  • https://git.kernel.org/pub/scm/linux/kernel/git/cip/linux-cip.git

• CIP Linux kernel CVE tracker
  • https://gitlab.com/cip-project/cip-kernel/cip-kernel-sec

• CIP Linux kernel failed patches tracker
  • https://gitlab.com/cip-project/cip-kernel/classify-failed-patches
Contact Information and Resources

To get the latest information, please contact:

- CIP Mailing List: cip-dev@lists.cip-project.org

Other resources

- Twitter: @cip_project
- CIP Web Site: https://www.cip-project.org
- CIP News: https://www.cip-project.org/news/in-the-news
- CIP Wiki: https://wiki.linuxfoundation.org/civilinfrastructureplatform/
- CIP Source Code
  - CIP repositories hosted at kernel.org: https://git.kernel.org/pub/scm/linux/kernel/git/cip/
  - CIP GitLab: https://gitlab.com/cip-project
CIP Talks at ELCE, CIP Mini Summit, and ATS

• Today (ELCE)
  • 15:15 @ Tête d'Or 2 (CIP)
    Open Source Projects to Live long and Prosper: Linux for Smart Infrastructure and Industry - Yoshitake Kobayashi, Toshiba Corporation & Urs Gleim, Siemens AG

• Thursday in CIP Mini Summit
  • 8:00 – 13:00 @ Lyon Convention Centre
    • https://www.cvent.com/Events/Register/RegNumConfirmation.aspx?e=66d78d44-9d3c-4c92-85fa-a87ef8e8b62b&_ga=2.224040339.985516515.1571730868-505872952.1553045385

• Thursday in Automated Testing Summit (ATS)
  • 15:10 @ Rhone 3AB (CIP Testing)
    A Guide to CIP Testing - Chris Paterson, Renesas Electronics Europe & Michael Adler, Siemens AG
Please Visit CIP Booth!

Place: FORUM 4/5 Sponsor Showcase

“CIP mini-summit” will be held on Oct. 31th (Thu), but sold out already, thank you!
Join us

CIP for sustainable Smart Cities with Open Source Software
Question?
Thank you
References

• *How to make Smart Cities stay smart with Open Source Projects*, OSS-J 2019, *Yoshitake Kobayashi*


• *Debian and Yocto Project based Long-term Maintenance Approaches for Embedded Products*, ELCE 2019, *Jan Kiszka and Kazuhiro Hayashi*