



———— CIVIL ————
INFRASTRUCTURE
———— PLATFORM ————

SLTS Kernel Maintenance and Patches Management

Nobuhiro Iwamatsu
CIP Kernel Maintainer, Toshiba

Pavel Machek
CIP Kernel Maintainer, Denx

SZ Lin (林上智)
CIP Kernel Working Group Chairperson and
TSC Representative, Moxa Inc.
CIP semi-summit 2019, France, 31st Oct.

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.

1	2	3	4	5	6	(*): Workgroup
SLTS kernel	Real-time	Testing	CIP Core	Security WG(*)	Software update WG	
✓	✓	✓	✓	✓	✓	Industrial grade
✓		✓	✓		✓	Sustainability
✓		✓	✓	✓	✓	Security

CIP Projects and its scopes

Policy and Progress

Current LTS Versions (Retrieved 22nd, Oct 2019)



Version	Maintainer	Released	Projected EOL
5.4	Greg Kroah-Hartman & Sasha Levin	2019-XX-XX	Dec, 2021
4.19	Greg Kroah-Hartman & Sasha Levin	2018-10-22	Dec, 2020
4.14	Greg Kroah-Hartman & Sasha Levin	2017-11-12	Jan, 2024
4.9	Greg Kroah-Hartman & Sasha Levin	2016-12-11	Jan, 2023
4.4	Greg Kroah-Hartman & Sasha Levin	2016-01-10	Feb, 2022
3.16	Ben Hutchings	2014-08-03	Apr, 2020

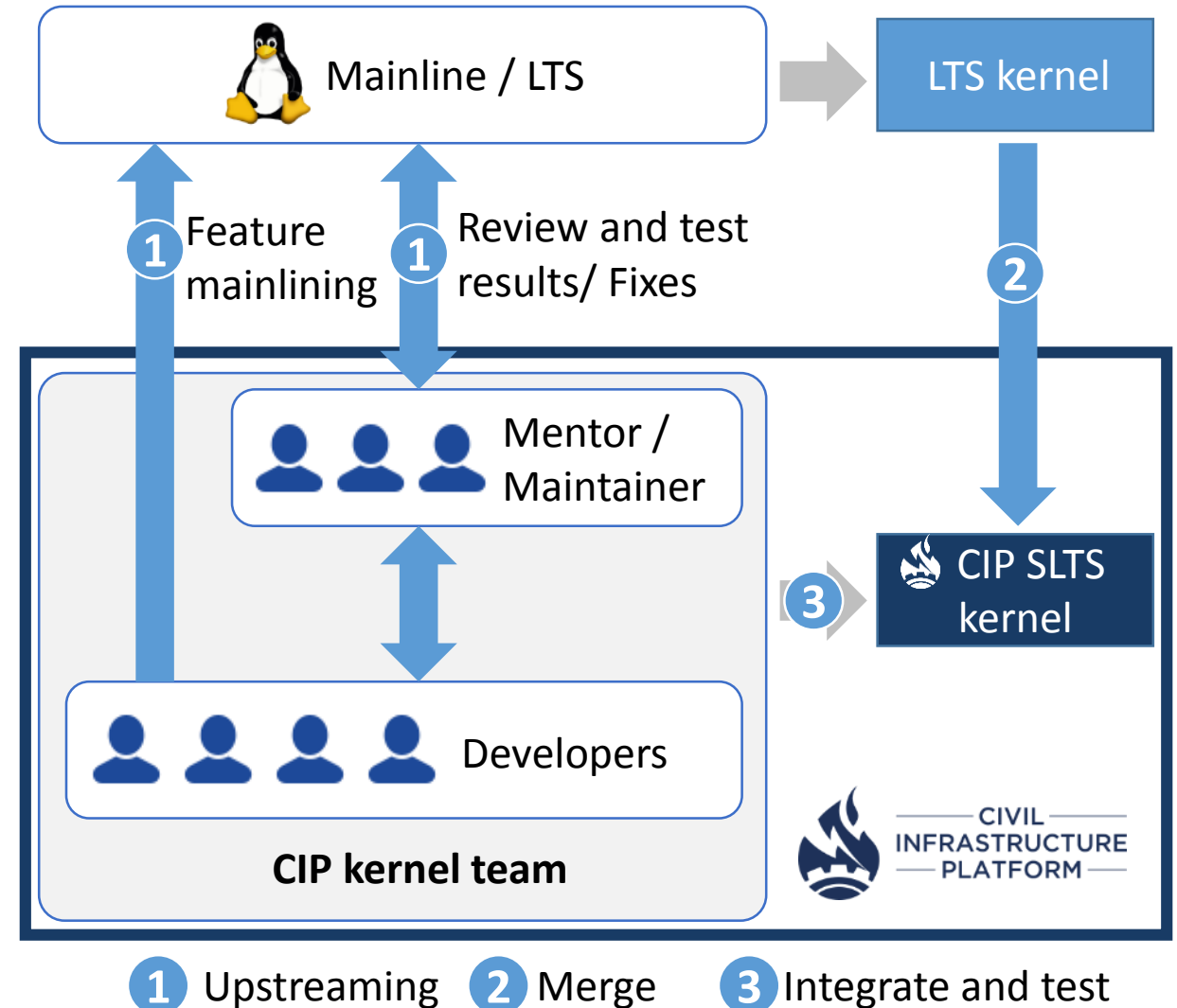
CIP SLTS Kernel Development (Upstream First Development)

- **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



Introduction to CIP Kernel Team Member



- **Chairperson**

- SZ Lin (林上智)

- **Maintainer**

- Nobuhiro Iwamatsu
- Pavel Machek

- **Mentor**

- Ben Hutchings

Current SLTS Versions (Retrieved 22nd, Oct 2019)



Version	Maintainer	First Release	Latest Release	Projected EOL
4.19	Nobuhiro Iwamatsu & Pavel Machek	2019-01-11 • v4.19.13-cip1	2019-10-12 • v4.19.78-cip12	2029+
4.19-rt	Pavel Machek	2019-01-11 • v4.19.13-cip1-rt1	2019-10-02 • v4.19.72-cip10-rt3	2029+
4.4	Nobuhiro Iwamatsu & Pavel Machek	2017-01-17 • v4.4.42-cip1	2019-10-12 • v4.4.196-cip38	2027+
4.4-rt	Pavel Machek	2017-11-16 • v4.4.75-cip6-rt1	2019-10-02 • v4.4.190-cip36-rt25	2027+



Maintenance Policy

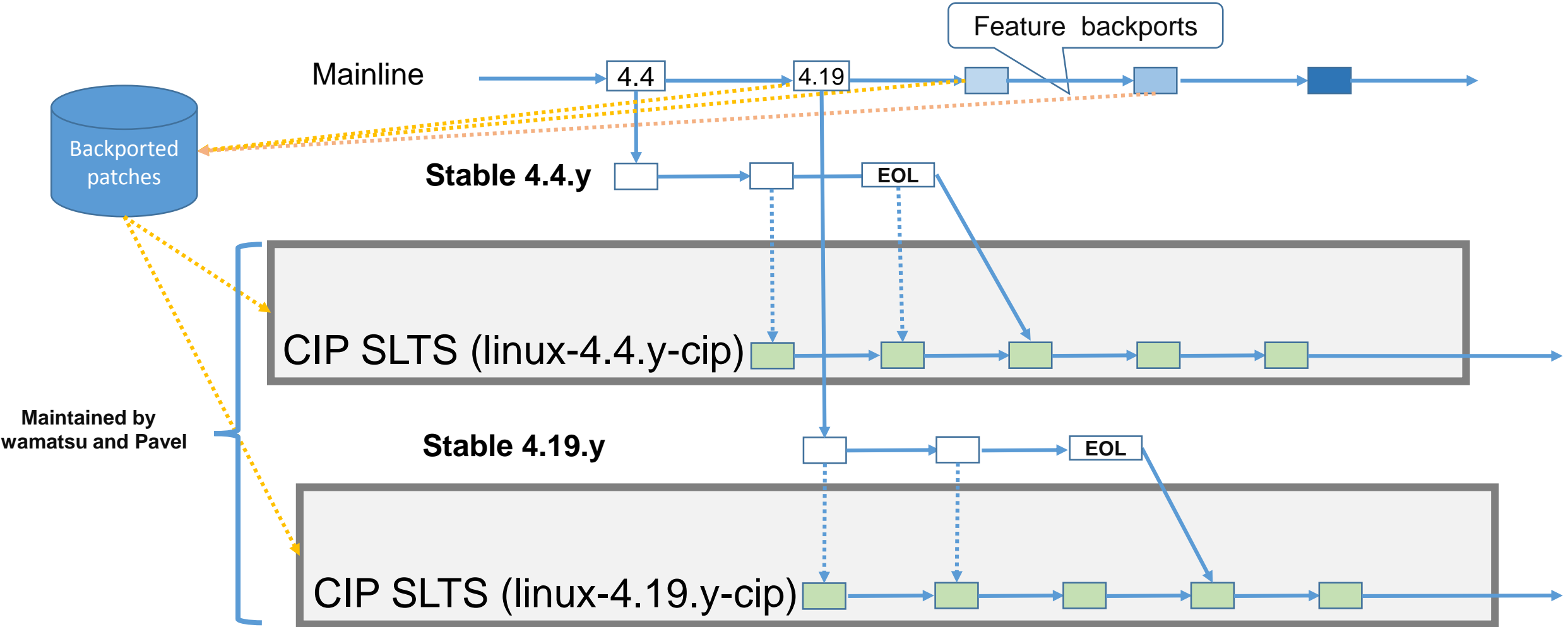
- <https://wiki.linuxfoundation.org/civilinfrastructureplatform/cipkernelmaintenance>
- Follow the stable kernel development rule as the basis
- Validation will be done by CIP test infrastructure and/or members
- Feature backports are acceptable
 - All features has to be in upstream kernel before backport 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



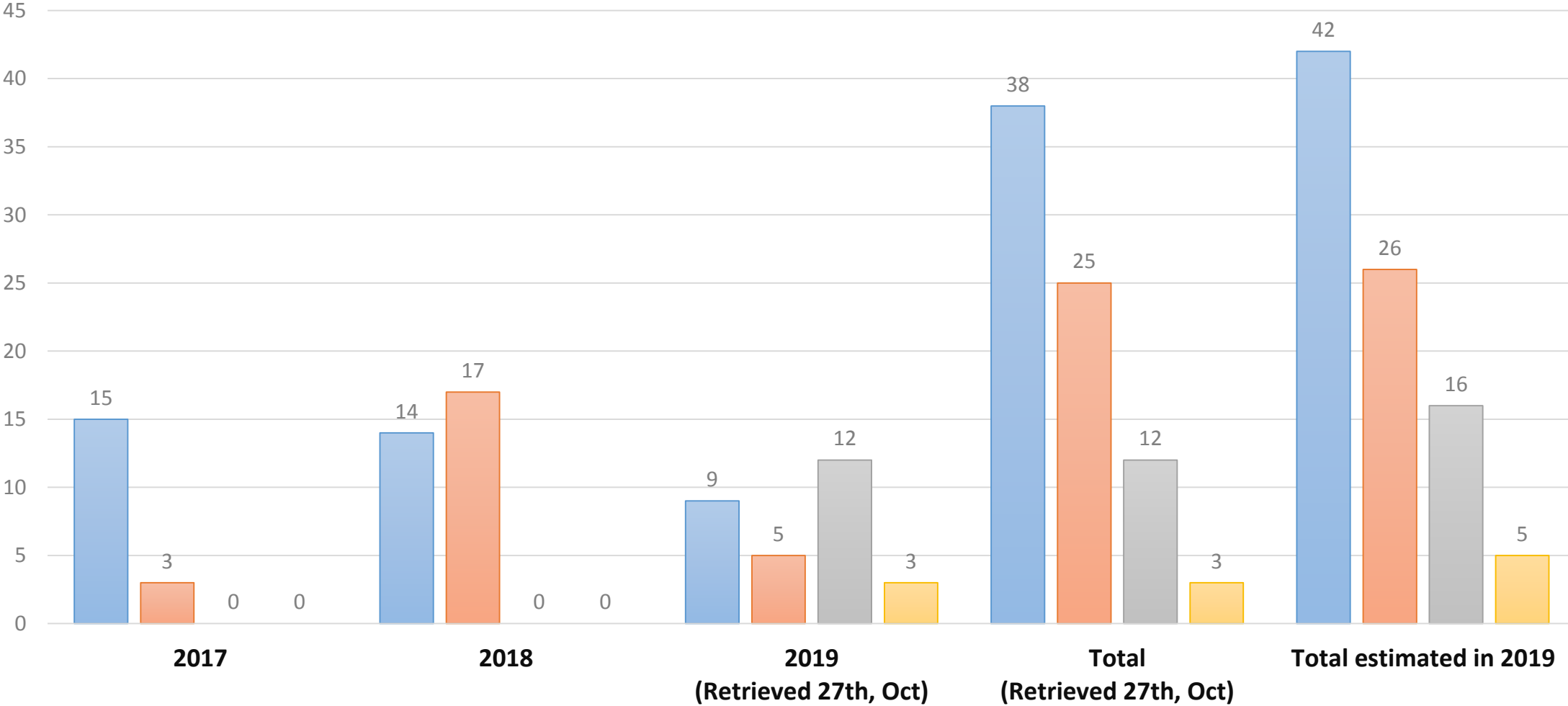
CIP SLTS Kernel Release Policy



<u>Release regularly</u>	<u>Release on demand</u>
Release 4.19 twice a month and 4.4 once a month (Effective June, 2019) <ul style="list-style-type: none">• Kernel 4.19<ul style="list-style-type: none">• second and fourth Fridays of the month• Kernel 4.4<ul style="list-style-type: none">• second Friday of the month	It depends on critical bug/ security fix
Release 4.19-rt once a month and 4.4-rt once every two months (Effective Nov, 2019)	Ditto

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



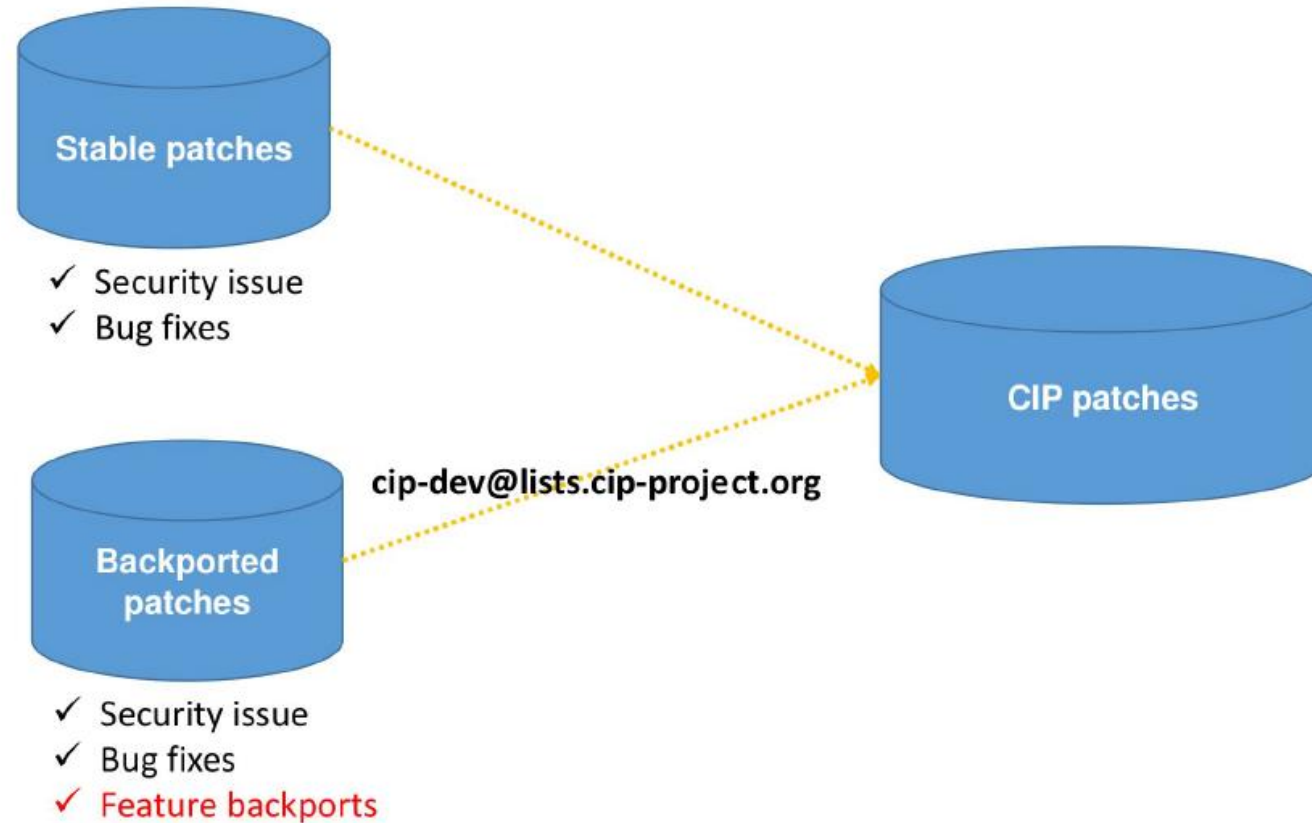
■ v4.4-cip ■ v4.4-cip-rt ■ v4.19-cip ■ v4.19-cip-rt

Patches Review

Patch Review



- CIP maintains two LTS-based kernels, 4.4.y and 4.19.y.
- We are reviewing patches for these kernels and patches that have been requested for feature backports.



Patch Review



- There are two LTS patch review targets.
 - Already released
 - 4.19.y: Pavel Machek
 - 4.4.y: Me
 - Develop and review on linux-stable@vger.kernel.org.
- We both review and manage the results at repository[0].
- Also, commits with problems are reported to upstream or linux-stable.

[0]: <https://gitlab.com/cip-project/cip-kernel/lts-commit-list>

Patch Review

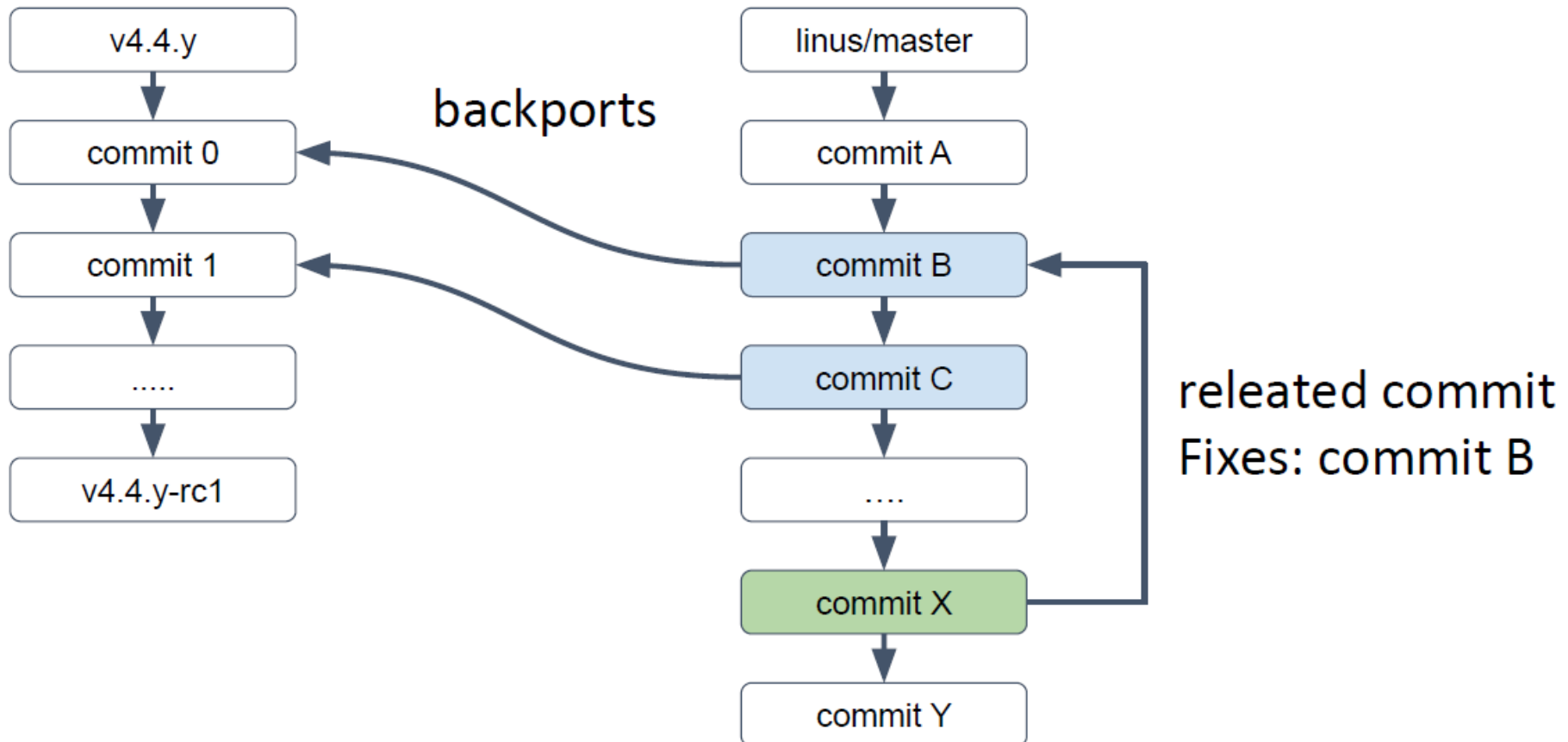


- Patch reviews are basically based on stable kernel rules.
Documentations/stable_kernel_rules.txt
- I carefully check the following:
 - Are there similar bugs elsewhere in similar code?
 - Are new bugs added to the newly code?
 - Is there a memory leak?
 - Is there any other good implementation?
 - Is the commit message correct?
 - Are there other related patches?

Patch Review



A review of stable patches includes checking for missing related patches.



Patch Review



in 4.4.y:

commit **cc475966e5f704f36ccc74575640e743fec248ad**

Author: David Ahern <dsahern@gmail.com>

Date: Wed May 1 18:18:42 2019 -0700

neighbor: Call `__ipv4_neigh_lookup_noref` in `neigh_xmit`

[Upstream commit 4b2a2bfef3f056461a90bd621e8bd7d03fa47f60]

Commit `cd9ff4de0107` changed the key for `IFF_POINTOPOINT` devices to `INADDR_ANY` but `neigh_xmit` which is used for MPLS encapsulations was not updated to use the altered key. The result is that every packet Tx does a lookup on the gateway address which does not find an entry, a new one is created only to find the existing one in the table right before the insert since `arp_constructor` was updated to reset the primary key. This is seen in the allocs and destroys counters:

```
ip -s -4 ntable show | head -10 | grep alloc
```

which increase for each packet showing the unnecessary overhead.

Fix by having `neigh_xmit` use `__ipv4_neigh_lookup_noref` for `NEIGH_ARP_TABLE`.

Fixes: `cd9ff4de0107` ("ipv4: Make neigh lookup keys for loopback/point-to-point devices be `INADDR_ANY`")





A patch to fix commit [4b2a2bf3f0](#) is included in the linus tree, but it is not yet applied in the 4.4.y tree.

in linus/master:

commit [9b3040a6aafd7898ece7fc7efcbca71e42aa8069](#)

Author: David Ahern <dsahern@gmail.com>

Date: Sun May 5 11:16:20 2019 -0700

ipv4: Define __ipv4_neigh_lookup_noref when CONFIG_INET is disabled

Define __ipv4_neigh_lookup_noref to return NULL when CONFIG_INET is disabled.

Fixes: [4b2a2bf3f0](#) ("neighbor: Call __ipv4_neigh_lookup_noref in neigh_xmit")

Patch Review



in linux/master:

```
commit 9b3040a6aafd7898ece7fc7efcbca71e42aa8069  
Author: David Ahern <dsahern@gmail.com>  
Date: Sun May 5 11:16:20 2019 -0700
```

```
ipv4: Define __ipv4_neigh_lookup_noref when CONFIG_INET is disabled
```

```
Define __ipv4_neigh_lookup_noref to return NULL when CONFIG_INET is disabled.
```

```
Fixes: 4b2a2bfeb3f0 ("neighbor: Call __ipv4_neigh_lookup_noref in neigh_xmit")
```

in 4.4.y:

```
commit 647f72b0d75c9faeec36b88fc051339e73008435  
Author: David Ahern <dsahern@gmail.com>  
Date: Sun May 5 11:16:20 2019 -0700
```

```
ipv4: Define __ipv4_neigh_lookup_noref when CONFIG_INET is disabled
```

```
commit 9b3040a6aafd7898ece7fc7efcbca71e42aa8069 upstream.
```

```
Define __ipv4_neigh_lookup_noref to return NULL when CONFIG_INET is disabled.
```

```
Fixes: 4b2a2bfeb3f0 ("neighbor: Call __ipv4_neigh_lookup_noref in neigh_xmit")
```

report and backports
with commit text

Kernel Maintenance



- Again
 - CIP maintains kernels based on 4.4.y 4.19.y.
 - We do not include out of tree code, and all kernel modifications must be included in Upstream.
- Release
 - We are releasing 4.19.y on Fridays in the 2nd and 4th weeks, 4.4.y on Fridays in 4th weeks every month.
 - It depends on critical bug / security fix.



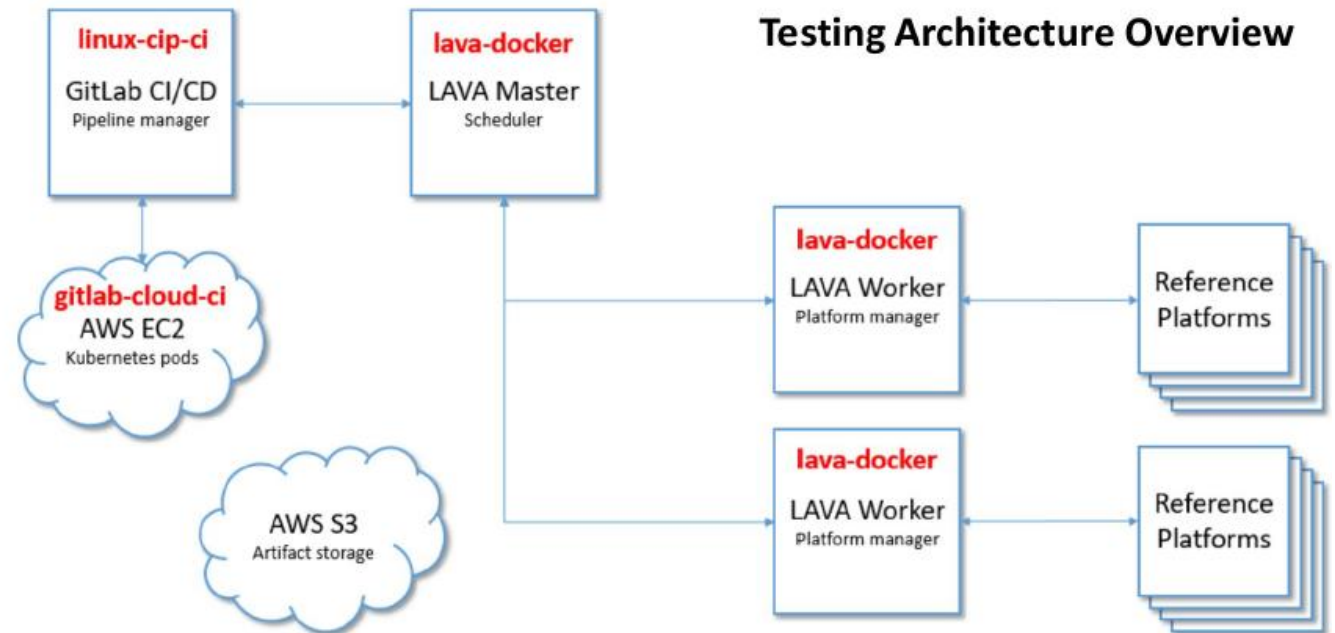
Testing and release

- Compile kernel tests for our reference board.
- Run the boot test, LTP, and kselftest using the created kernel image to confirm that there are no regression problems .

Kernel Maintenance



- At first, the local environment is checked for problems, but finally Gitlab and LAVA managed by CIP are used for kernel compiling and testing.
- Chris Paterson will talk about this later.



Kernel Maintenance



- We also manage the -rebase branch.
 - This is to separate the commits we captured from the LTS commits for clarity.
 - It is also used for patch maintenance and management.

```
$ git log --decorate --oneline --graph linux-4.4.y-cip
* 83bce8ee46df (tag: v4.4.196-cip38) CIP: Bump version suffix
* a2cbf3a26dc8 Merge tag 'v4.4.196' into linux-4.4.y-cip
|\
| * c61ebb668f2c (tag: v4.4.196) Linux 4.4.196
| * 2e486758901d NFC: fix attrs checks in netlink interface
<snip>
| * 272e1835861e ALSA: hda - Fix potential endless loop at app
* | acb88309d814 gitlab-ci: Run tests on RZ/G1C iwg23s platfor
* | a5f1c421464f gitlab-ci: Always store job artifacts
* | 5253596c2e9b gitlab-ci: Increase test timeout to 60 minutes
* | 2d85c7070029 (tag: v4.4.192-cip37) CIP: Bump version suffi
* | aa8eb4f69a03 Merge tag 'v4.4.192' into linux-4.4.y-cip
|\
| |
| * 882f8791e141 (tag: v4.4.192)
```



```
$ git log --decorate --oneline --graph linux-4.4.y-cip-rebase
* eb744d702d93 (tag: v4.4.196-cip38-rebase) CIP: Bump version suffix
* ae3fb68390e0 gitlab-ci: Run tests on RZ/G1C iwg23s platform
* 34b0a906b19a gitlab-ci: Always store job artifacts
* 4a5f2f17da5c gitlab-ci: Increase test timeout to 60 minutes
* 458ab9722770 CIP: Bump version suffix to -cip37 after merge from st
* f5a80f24e5c4 ARM: dts: iwg23s-sbc: Enable HS-USB
* 76eafac95424 ARM: dts: r8a77470: Add HSUSB device nodes
* * ca75f91b47fd dt-bindings: usb: renesas_usbhs: Add support for r8a7
* 9bc064505191 dt-bindings: usb: renesas_usbhs: Add support for r8a7
* 41604073912f ARM: dts: iwg23s-sbc: Enable USB USB2.0 Host
* 249ef90d680e ARM: dts: iwg23s-sbc: Enable USB Phy[01]
* ac0e6ce8030f ARM: shmobile: Enable USB [EO]HCI HCD PLATFORM
* e7af62b1ffe8 ARM: shmobile: Enable PHY_RCAR_GEN3_USB2 in sh
* 9584390a673e ARM: dts: r8a77470: Add USB2.0 Host (EHCI/OHCI) d
<snip>
```


Documented Rules



- a) It or an equivalent fix must already exist in Linus' tree (upstream).
- b) It must be obviously correct and tested.
- c) It must fix a real bug that bothers people (not a, "This could be a problem..." type thing).
- 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.
- e) It cannot contain any "trivial" fixes in it (spelling changes, whitespace cleanups, etc).

Real Rules



a) It or an equivalent fix must already exist in Linus' tree (upstream).

[b) It must not be obviously broken]

Cc: stable



People actually follow the rules

- mostly

Auto-picked patches do not

Other Players



SuSE

- <https://kernel.suse.com/>
- 4.4 but not 4.19

Kinds of Patches



Crashes, bad errors

Run time `WARN_ON()`

Confusing printk messages

Compile time warnings

Regular Meeting

Weekly Regular Online Meeting



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

US-West	US-East	UK	DE	TW	JP
02:00	05:00	09:00	10:00	17:00	18:00

- **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.

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](https://twitter.com/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>

Join us

CIP for sustainable Smart Cities with Open Source Software



CIVIL
INFRASTRUCTURE
PLATFORM

RENESAS

SIEMENS

TOSHIBA

CodeThink

cybertrust

HITACHI
Inspire the Next

MOXA®

Plat'Home
There, we are. Internet of Things

Question?



Thank you



— CIVIL —
INFRASTRUCTURE
— PLATFORM —

References



- *How to make Smart Cities stay smart with Open Source Projects*, OSS-J 2019, **Yoshitake Kobayashi**
- *The Activity of the Security Working Group in the CIP Project*, OSS-J 2019, **Takehisa Katayama**
- *Debian and Yocto Project based Long-term Maintenance Approaches for Embedded Products*, ELCE 2019, **Jan Kiszka and Kazuhiro Hayashi**