



# Cookies for Kernel Developers



Nikolai Kondrashov, Red Hat  
@spbnick

Major Hayden, Red Hat  
@majorhayden

Photo credit: <https://pxhere.com/en/photo/225741>



# Who are we?



## Nikolai Kondrashov

Senior Software Engineer  
Red Hat

DIGImend project maintainer.  
Enjoys electronics and  
embedded as a hobby.



## Major Hayden

Principal Software Engineer  
Red Hat

Owner of too many domain  
names, including  
[icanhazip.com](http://icanhazip.com).

(Please do not give me any  
other ideas for domain names  
to buy.) ;)

# Continuous Kernel Integration project



“Cookie”

A team from Red Hat's  
Base Operating Systems and  
Global QE

- China
- Czech Republic
- Finland
- United States

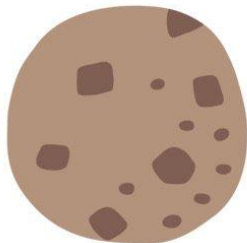
Apple



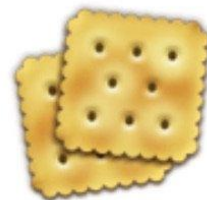
LG



Google



Samsung



Microsoft



HTC



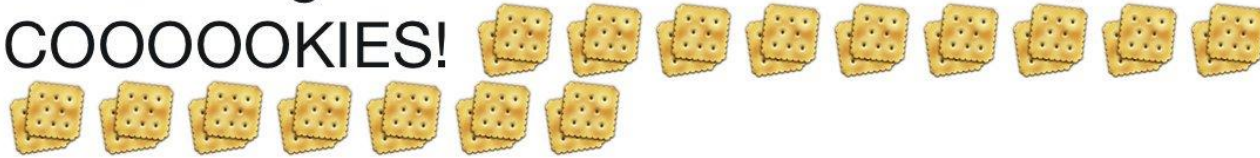


**Cookie Monster** ✓

@MeCookieMonster



IT #WORLDCHOCOLATEDAY!!! Me  
celebrating with lots of chocolate chip  
COOOOOKIES!







**Maintaining stable kernels  
is a complex and difficult task**

FIG. 1

A developer writes a patchset  
that gets merged into the  
mainline\* kernel tree

\* Patches that arrive here are included in the next major kernel release.

Time passes





The patchset becomes part of  
Greg's\* stable kernel release

\* Possibly the most efficient kernel developer on Earth

Time passes



A Linux distribution maintainer  
finds a bug or security problem\*  
with that patch

\* After a lengthy investigation, bisection, and debugging

Distribution maintainer contacts the  
original developer



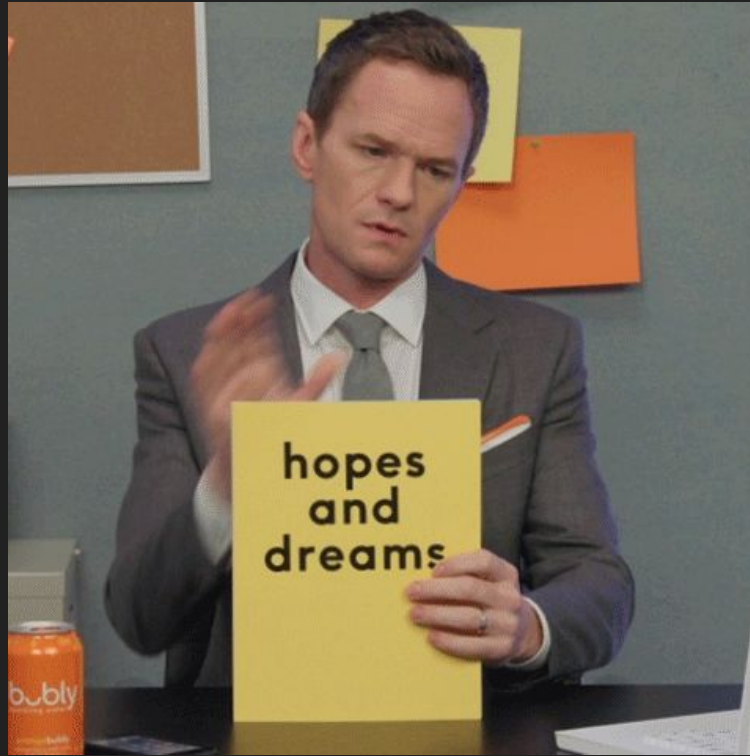


Time passes





By this time,  
the original developer can't remember  
why they wrote the patch  
or what is in the patch



This is a frustrating,  
time consuming process.

A photograph of two firefighters in full gear, including helmets and jackets, standing in front of a large fire. The scene is dark and smoky, with bright orange flames and thick black smoke rising from the fire. The firefighters are silhouetted against the bright light of the fire. The text is overlaid in white on the dark background.

**What if we could find that problematic patch before it is ever merged?**

We built CI for  
kernel contributions

# High-level process overview



Watch git for commits,  
patchwork for patches



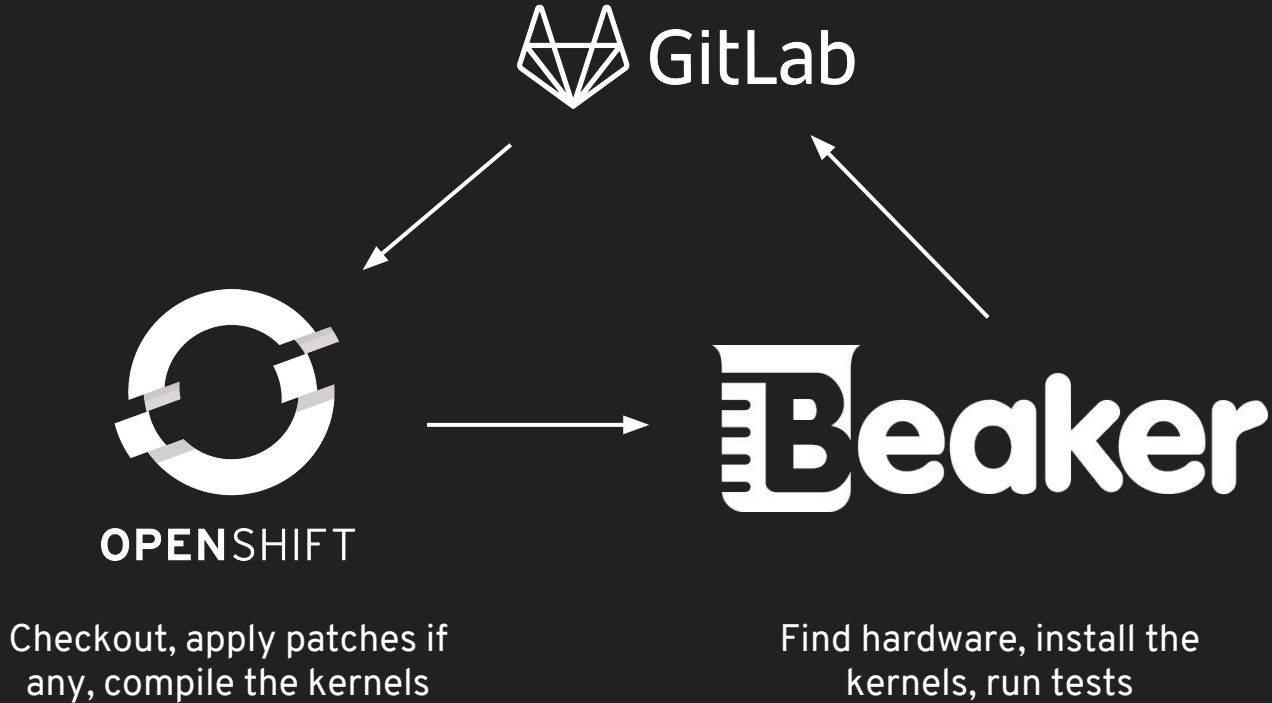
Clone the kernel tree,  
Apply patches (optional),  
Compile and test the kernel



CI results go to kernel  
mailing lists as part of the  
original email thread



# Building and testing





Red Hat maintains many kernel tests and can run them on hard-to-get hardware platforms

Including everyone's favorite: IA-64!

# Test suites onboarded so far

**Libhugetlbfs**

**LTP lite**

**KVM unit tests**

**PMTU discovery**

**USEX**

**Connectathon NFS**

...

# Architectures

AMD

INTEL

x86\_64

aarch64

AppliedMicro

Cavium

Qualcomm

Ampere

Hisilicon

IBM POWER8

ppc64le

s390x

IBM z12

IBM POWER9

IBM z13

# Platforms & Peripherals

**Servers**

**GPUs**

**Mainframes**

**Network cards**

**Workstations**

**Storage controllers**

**Laptops**

**Audio cards**

**Virtual machines**

**Infiniband adapters**



What are we doing  
for upstream kernels today?

It all started in a  
conversation with Greg KH...

# PASS: Test report for kernel 4.20.1.cki (linux-stable-rc)

[\[Date Prev\]](#)[\[Date Next\]](#)[\[Thread Prev\]](#)[\[Thread Next\]](#)[\[Date Index\]](#)[\[Thread Index\]](#) 

- 
- *Subject:* PASS: Test report for kernel 4.20.1.cki (linux-stable-rc)
  - *From:* CKI <cki-project@xxxxxxxxxxx>
  - *Date:* Wed, 9 Jan 2019 17:58:58 -0500
- 

Hello,

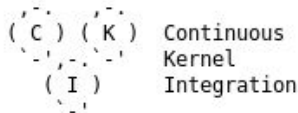
We ran automated tests on a recent commit from this kernel tree:

```
Kernel repo: git://git.kernel.org/pub/scm/linux/kernel/git/stable/linux-stable-rc.git
Commit: 8c3f48e8c288 Linux 4.20.1
```

The results of these automated tests are provided below.

```
Overall result: PASSED
Patch merge: OK
Compile: OK
Kernel tests: OK
```

Please reply to this email if you have any questions about the tests that we ran or if you have any suggestions on how to make future tests more effective.



---

## Compile testing

-----

We compiled the kernel for 4 architectures:

s390x:

make options: make INSTALL\_MOD\_STRIP=1 -j64 targz-pkg -j64

configuration: <https://artifacts.cki-project.org/builds/s390x/8c3f48e8c28823378274d2342a2ff1442a4af55f.config>

powerpc64le:

make options: make INSTALL\_MOD\_STRIP=1 -j64 targz-pkg -j64

configuration: <https://artifacts.cki-project.org/builds/ppc64le/8c3f48e8c28823378274d2342a2ff1442a4af55f.config>

aarch64:

make options: make INSTALL\_MOD\_STRIP=1 -j64 targz-pkg -j64

configuration: <https://artifacts.cki-project.org/builds/aarch64/8c3f48e8c28823378274d2342a2ff1442a4af55f.config>

x86\_64:

make options: make INSTALL\_MOD\_STRIP=1 -j64 targz-pkg -j64

configuration: [https://artifacts.cki-project.org/builds/x86\\_64/8c3f48e8c28823378274d2342a2ff1442a4af55f.config](https://artifacts.cki-project.org/builds/x86_64/8c3f48e8c28823378274d2342a2ff1442a4af55f.config)

## Hardware testing

-----

We booted each kernel and ran the following tests:

s390:

Boot test

- URL: <https://github.com/CKI-project/tests-beaker/archive/master.zip#distribution/kpkginstall/distribution/command>

LTP lite - release 20180926

- URL: <https://github.com/CKI-project/tests-beaker/archive/master.zip#distribution/ltp/lite>

Memory function: memfd\_create

- URL: [https://github.com/CKI-project/tests-beaker/archive/master.zip#/memory/function/memfd\\_create](https://github.com/CKI-project/tests-beaker/archive/master.zip#/memory/function/memfd_create)

Networking route: pmtu

- URL: <https://github.com/CKI-project/tests-beaker/archive/master.zip#/networking/route/pmtu>

AMTU (Abstract Machine Test Utility)

- URL: <https://github.com/CKI-project/tests-beaker/archive/master.zip#misc/amtu>

powerpc:

Boot test

- URL: <https://github.com/CKI-project/tests-beaker/archive/master.zip#distribution/kpkginstall/distribution/command>

LTP lite - release 20180926

- URL: <https://github.com/CKI-project/tests-beaker/archive/master.zip#distribution/ltp/lite>

xfstests: ext4

- URL: <https://github.com/CKI-project/tests-beaker/archive/master.zip#/filesystems/xfs/xfstests>

xfstests: xfs

- URL: <https://github.com/CKI-project/tests-beaker/archive/master.zip#/filesystems/xfs/xfstests>

Memory function: memfd\_create

- URL: [https://github.com/CKI-project/tests-beaker/archive/master.zip#/memory/function/memfd\\_create](https://github.com/CKI-project/tests-beaker/archive/master.zip#/memory/function/memfd_create)

Networking route: pmtu

- URL: <https://github.com/CKI-project/tests-beaker/archive/master.zip#/networking/route/pmtu>

AMTU (Abstract Machine Test Utility)

- URL: <https://github.com/CKI-project/tests-beaker/archive/master.zip#misc/amtu>

Usex - version 1.9-29

- URL: <https://github.com/CKI-project/tests-beaker/archive/master.zip#standards/usex/1.9-29>



# Stable queue: queue-4.20

[\[Date Prev\]](#)[\[Date Next\]](#)[\[Thread Prev\]](#)[\[Thread Next\]](#)[\[Date Index\]](#)[\[Thread Index\]](#) 

- 
- *Subject:* Stable queue: queue-4.20
  - *From:* CKI <cki-project@xxxxxxxxxxx>
  - *Date:* Fri, 11 Jan 2019 05:54:06 -0500
- 

Hello,

We ran automated tests on a patchset that was proposed for merging into this kernel tree. The patches were applied to:

Kernel repo: `git://git.kernel.org/pub/scm/linux/kernel/git/stable/linux.git`  
Commit: `8c3f48e8c288` Linux 4.20.1

The results of these automated tests are provided below.

Overall result: PASSED  
Patch merge: OK  
Compile: OK  
Kernel tests: OK

Please reply to this email if you have any questions about the tests that we ran or if you have any suggestions on how to make future tests more effective.



## Merge testing

-----

We cloned this repository and checked out a ref:

```
Repo: git://git.kernel.org/pub/scm/linux/kernel/git/stable/linux.git
Ref: 8c3f48e8c288 Linux 4.20.1
```

We then merged the following patches with `git am`:

```
scsi-zfcp-fix-posting-too-many-status-read-buffers-leading-to-adapter-shutdown.patch
scsi-lpfc-do-not-set-queue-page_count-to-0-if-pc_sli4_params.wqpcnt-is-invalid.patch
fork-record-start_time-late.patch
zram-fix-double-free-backing-device.patch
hwpoison-memory_hotplug-allow-hwpoisoned-pages-to-be-offlined.patch
mm-devm_memremap_pages-mark-devm_memremap_pages-export_symbol_gpl.patch
mm-devm_memremap_pages-kill-mapping-system-ram-support.patch
mm-devm_memremap_pages-fix-shutdown-handling.patch
memcg-oom-notify-on-oom-killer-invocation-from-the-charge-path.patch
sunrpc-fix-cache_head-leak-due-to-queued-request.patch
sunrpc-use-svc_net-in-svcauth_gss_-functions.patch
mm-devm_memremap_pages-add-memory_device_private-support.patch
mm-hmm-use-devm-semantics-for-hmm_devmem_-add-remove.patch
mm-hmm-replace-hmm_devmem_pages_create-with-devm_memremap_pages.patch
mm-hmm-mark-hmm_devmem_-add-add_resource-export_symbol_gpl.patch
mm-swap-fix-swapoff-with-ksm-pages.patch
media-cx23885-only-reset-dma-on-problematic-cpus.patch
alsa-cs46xx-potential-null-dereference-in-probe.patch
alsa-usb-audio-avoid-access-before-blength-check-in-build_audio_procunit.patch
alsa-usb-audio-check-mixer-unit-descriptors-more-strictly.patch
alsa-usb-audio-fix-an-out-of-bound-read-in-create_composite_quirks.patch
```

Coming upstream soon



Lint



Merge



Next

tests++

trees++

latency--

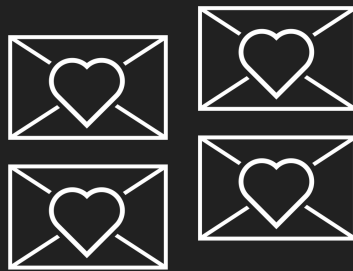
Next

`open(planning)`

`open(logs)`

`open(docs)`

# Get your commits tested

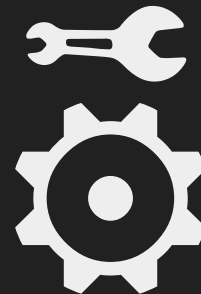


Write to  
[cki-project@redhat.com](mailto:cki-project@redhat.com)

We evaluate  
impact on RHEL

You get your  
reports!

# Have us run your tests



Write to  
[cki-project@redhat.com](mailto:cki-project@redhat.com)

We evaluate  
impact on RHEL

Together we add a  
wrapper for Beaker

You maintain it,  
we run it!



Why is Red Hat doing this?



We want better RHEL\*

\*Red Hat Enterprise Linux

# Get involved



<https://github.com/cki-project>



<https://gitlab.com/cki-project>



[cki-project@redhat.com](mailto:cki-project@redhat.com)

# Thank you!

Ask some questions and get  
some (real) cookies. 🍪 🍪 🍪

**Got questions after the talk?**

E-mail [cki-project@redhat.com](mailto:cki-project@redhat.com)

Visit <https://cki-project.org/>

(slides on the website soon!)