

Common Test Report Database (KCIDB)

Linux Plumbers 2021

Nikolai Kondrashov



Nikolai Kondrashov

fn    **spbnick**

Sr. Software Engineer, Red Hat, CKI Project

KernelCI KCIDB maintainer

DIGImend maintainer

Electronics and embedded enthusiast

Born in Russia, living in Finland

Open the slides and
click the **links!**



Describes upcoming **v9 release**



KernelCI Testing systems



...

KernelCI Testing reports

Results from Linaro's test farm.
No regressions on arm64, arm, x86_64, and i386.

Summary

kernel: 5.7.12-rc3
git repo: <https://git.kernel.org/pub/scm/linux/kernel/git/stable/linux-stable-rc.git>
git branch: linux-5.7.12
git commit: 48223ab6f6c9d90b3b7f7f71de2a29226
git describe: v5.7.12-32-g48223ab6f6f
Test details: <https://qa-reports.linaro.org/ikfr/linux-stable-rc-5.7-oe/build/v5.7.12-32-g48223ab6f6f>
No regressions (compared to build v5.7.12)
No fixes (compared to build v5.7.12)

Run 34339 total tests in the following environments and test suites.

Environments

- dragonboard-410c
- hi3620-hikey
- i386
- juno-r2
- juno-r2-compat
- juno-r2-kasan
- nxp-ls2088
- qemu_arm
- qemu_arm64
- qemu_i386
- qemu_x86_64
- x15
- x86
- x86-kasan

Test Suites

KMSAN: kernel-usb-ufaleak in hif_usb_send

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

- To: glider@xxxxxxxxx, gregkh@xxxxxxxxxxxxxxxxxx, Ingrassia@xxxxxxxxxxxxxxxx, linux-kernel@xxxxxxxx
- Subject: KMSAN: kernel-usb-ufaleak in hif_usb_send
- From: syzbot+fb3a1df1ed4f7e8d8ad@xxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
- Date: Tue, 11 Aug 2020 10:06:18 -0700

Hello,

syzbot found the following issue on:

HEAD commit: c80856d1 wip: changed copy from user where instrumented
git tree: <https://github.com/google/kmsan.git> master
console output: <https://syzkaller.appspot.com/view.txt?id=246453c980000>
kernel config: <https://syzkaller.appspot.com/view.txt?id=246453c980000>
detailed Link: <https://syzkaller.appspot.com/view.txt?id=246453c980000>
compiler: clang version 10.0.0 (<https://github.com/llvm/llvm-project>) c2443155a0fb245c817f2c1c72b6a39186e81
userspace arch: i386

Unfortunately, I don't have any reproducer for this issue yet.

IMPORTANT: if you fix the issue, please add the following tag to the commit:
Reported-by: syzbot+fb3a1df1ed4f7e8d8ad@xxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

usb 4-1: at90k: Host: Transferred FW: at90k_hic/firmware-1.4.0-fw, size: 51008

BUG: KMSAN: kernel-usb-ufaleak in kasan_handle_urb->data me/kasan/kasan_hooks.c:267
CPU: 1 PID: 13922 Comm: kworker/1:0 Not tainted 5.8.0-rc5-syzkaller #0
Hardware name: Google Compute Engine/Google Compute Engine, BIOS Google 01/01/2011
Workqueue: events request_firmware_work_func
Call Trace:

stable-rc/linux-4.9.y build: 197 builds: 1 failed, 196 passed, 4 warnings (v4.9.232)

kernel.org hot [hot](#) at [kernel.org](#)

Sat Aug 1 08:24:58 UTC 2020

- Previous message: [\[PATCH stable\] MIPS: Loongson: Introduce and use loongson_llsc_mbf\(\)](#)
- Next message: [stable-rc/linux-4.9.y build: 197 builds: 1 failed, 196 passed, 4 warnings \(v4.9.232\)](#)
- Messages sorted by: [\[date\]](#) [\[thread\]](#) [\[subject\]](#) [\[author\]](#)

stable-rc/linux-4.9.y build: 197 builds: 1 failed, 196 passed, 4 warnings (v4.9.232)

Full Build Summary: <https://kernel.org/build/stable-rc/branch/linux-4.9.y/kernel/v4.9.232/>

Tree: stable-rc
Branch: linux-4.9.y
Git describe: v4.9.232
Git commit: 8686c4129c29293b2a7b80e1d5801abe183b
Git URL: <https://git.kernel.org/pub/scm/linux/kernel/git/stable/linux-stable-rc.git>
Built: 6 unique architectures

Build Failure Detected:

nips:
3272el defconfig: (gcc-8) FAIL

Warnings Detected:

src:

✓ PASS: Test report for kernel 5.7.12-3ff3d4f.cki (stable-queue)

CKI Project [cki-project](#) at [redhat.com](#)

Tue Aug 4 12:35:15 UTC 2020

- Previous message: [fixing 4.14-stable's broken DM cache writethrough support \[was: Re: \(resend\) PATCH v3: \(linux-4.14.y\) dm cache: submit writethrough writes in parallel to origin and cache\]](#)
- Next message: [stable-rc/linux-4.4.y build: 190 builds: 1 failed, 189 passed, 2 errors, 13 warnings \(v4.4.232-30-g52247eb98ebe\)](#)
- Messages sorted by: [\[date\]](#) [\[thread\]](#) [\[subject\]](#) [\[author\]](#)

Hello,

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

Kernel repo: <https://git.kernel.org/pub/scm/linux/kernel/git/stable/linux-stable-rc.git>
Commit: 3ff3d4f43856 - x86/8250: Use printk_deferred() to prevent deadlock

The results of these automated tests are provided below.

Overall result: PASS5D
Merge: OK
Compile: OK
Tests: OK

All kernel binaries, config files, and logs are available for download here:

<https://cki-artifacts.s3.us-east-2.amazonaws.com/index.html?prefix=datawarehouse/2020/08/03/611391>

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.

(C) (K)
Continuous
Kernel
Integration

Re: [PATCH V4] hwmon: add fan/pwm driver for corsair

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

- To: Jaap aarts <jaap.aarts1@xxxxxxxxxx>, Jean Delvare <jdelvare@xxxxxxxxxx>, Guenter Roeck <linux@xxxxxxxxxx>, Barnabas Pócs <pobrn@xxxxxxxxxxxxxxxx>
- Subject: Re: [PATCH V4] hwmon: add fan/pwm driver for corsair h100i platinum
- From: Kernel test robot <clpk@xxxxxxxxxx>
- Date: Sun, 16 Aug 2020 17:34:40 +0800
- Cc: kbuild-all@xxxxxxxxxxxxx, jaap aarts <jaap.aarts1@xxxxxxxxxx>
- In-reply-to: <20200815211617.86565-1-jaap.aarts1@gmail.com>
- References: <20200815211617.86565-1-jaap.aarts1@gmail.com>
- User-agent: Mutt/1.10.1 (2018-07-13)

Hi Jaap,

Thank you for the patch! Perhaps something to improve:

[auto build test WARNING on hwmon/hwmon-next]
[also build test WARNING on v5.8 next-20200814]
[If your patch is applied to the wrong git tree, kindly drop us a note.
And when submitting patch, we suggest to use '-base' as documented in
<https://git-scm.com/docs/git-format-patch>]

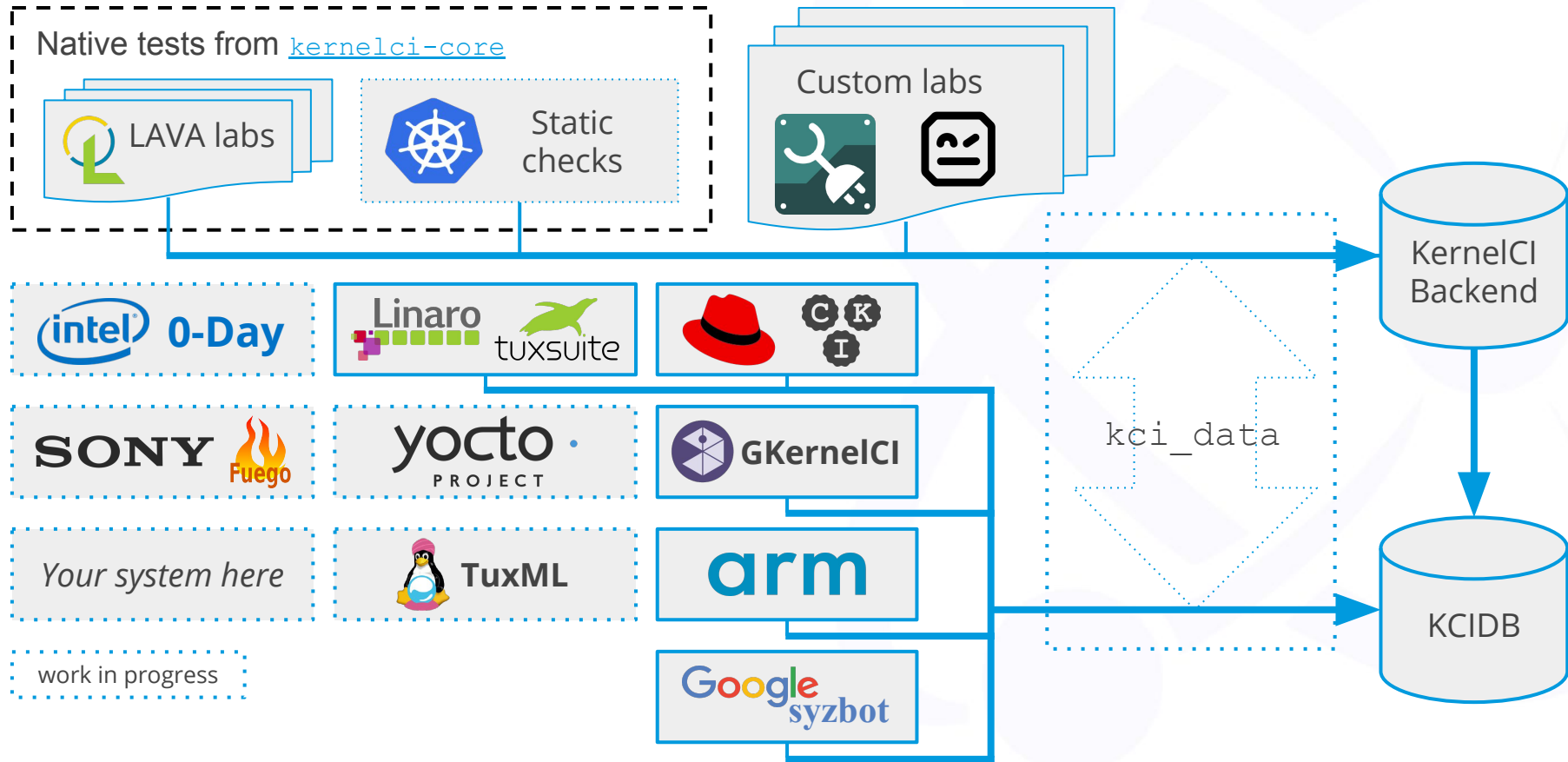
url: <https://github.com/8day-ci/linux/commit/jaap-aarts/hwmon-add-fan-pwm-driver-for-corsair-h100i-platinum/20200816-865292>
base: <https://git.kernel.org/pub/scm/linux/kernel/git/groeck/linux-staging.git> hwmon-next
config: nios2-defconfig-080-20200816 (attached as .config)
compiler: nios2-linux-gcc (GCC) 9.3.0



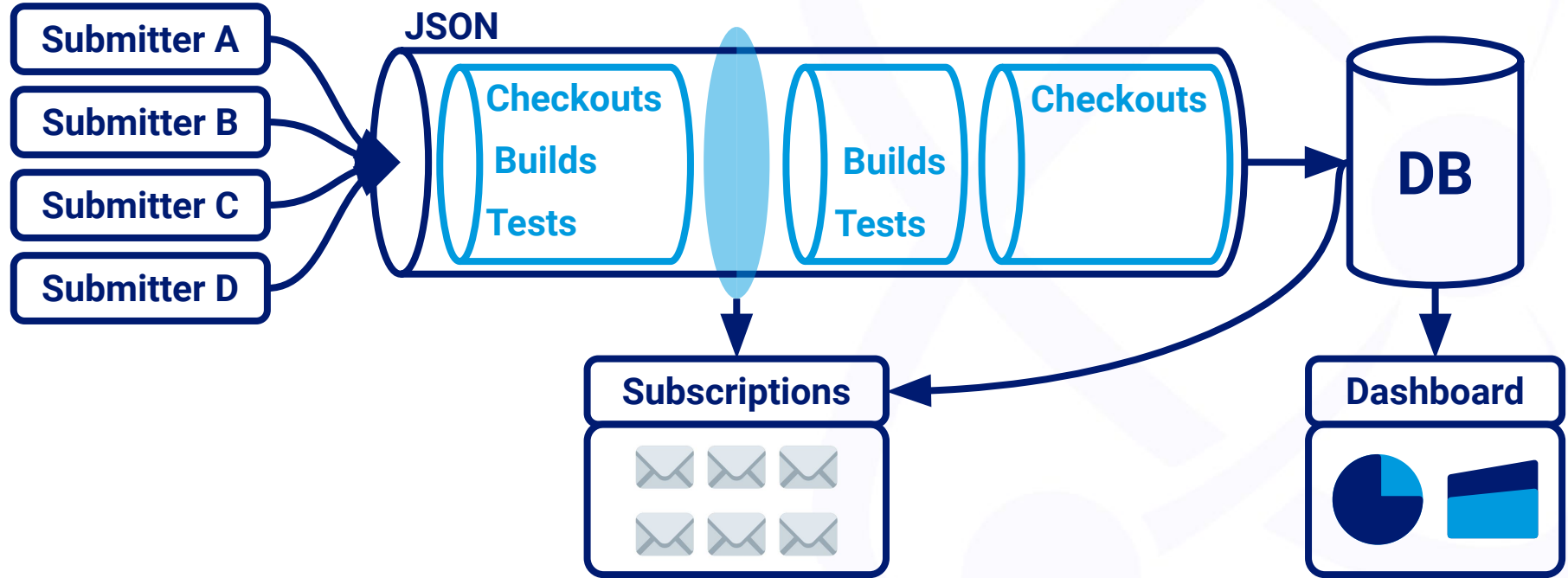


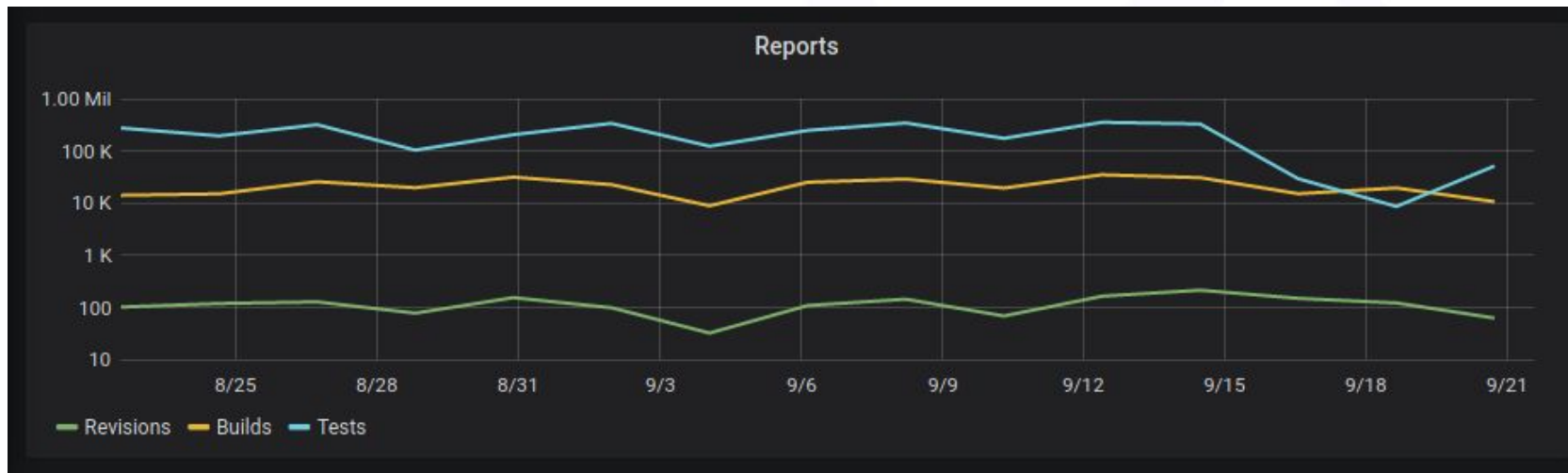
KCIDB

KernelCI The Bigger Picture

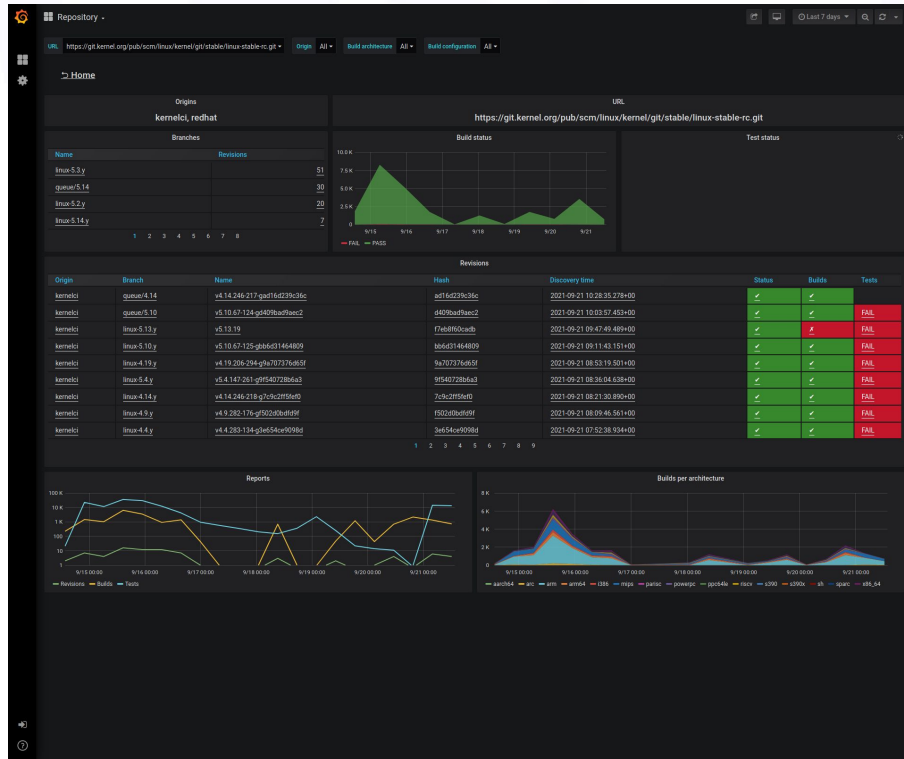
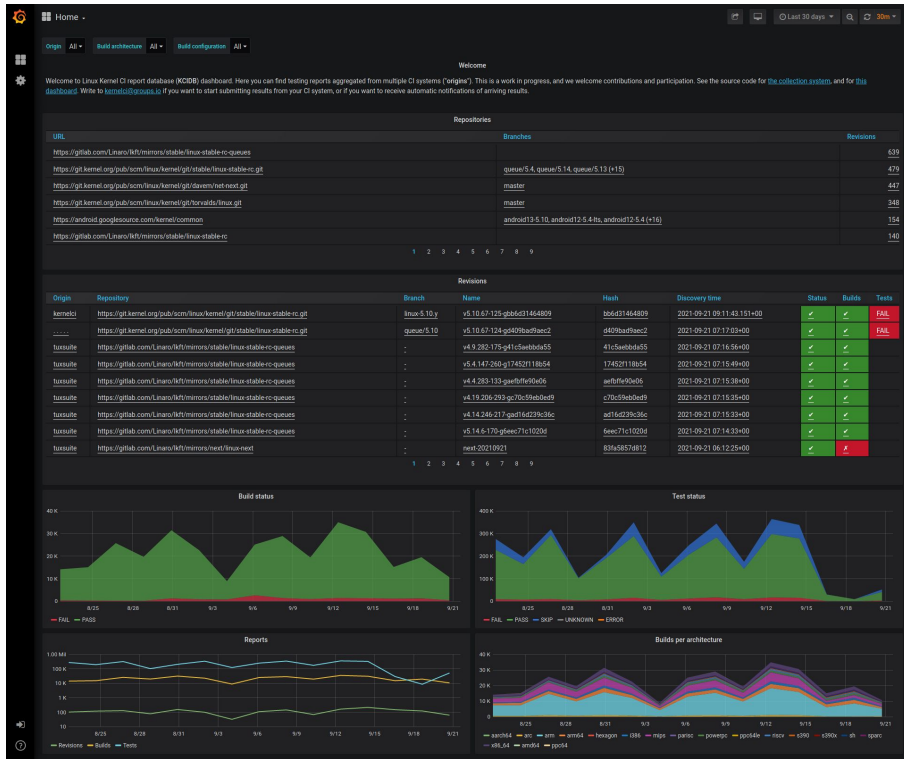


KernelCI The Smaller Bigger Picture

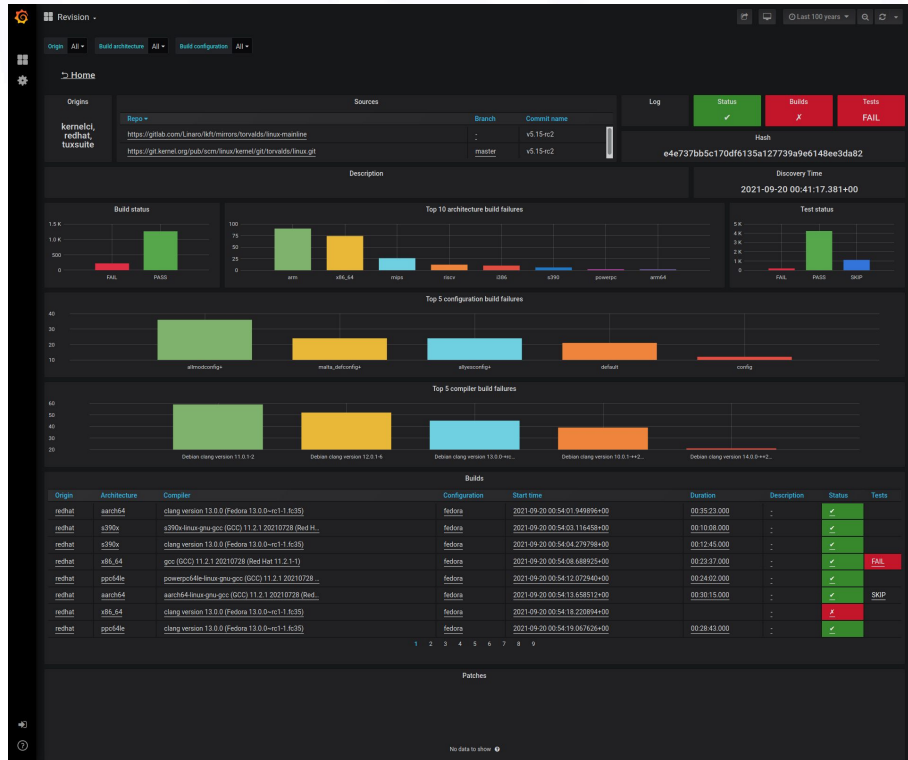
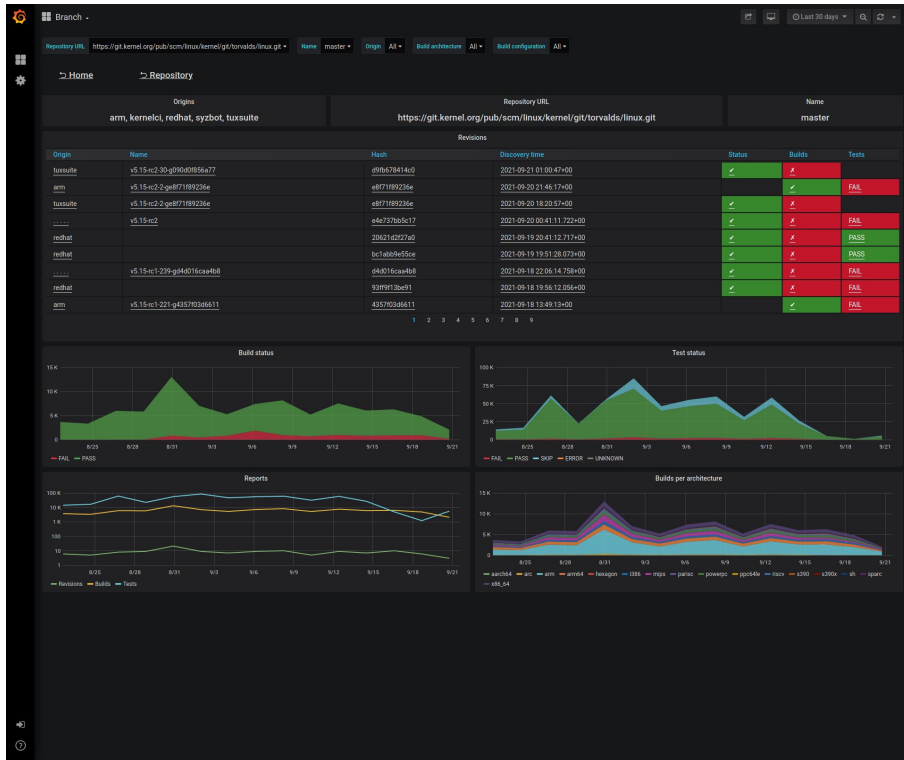




KernelCI Dashboard



KernelCI Dashboard





```
nkondras@ruby:~$ kcidb-
```

```
kcidb-count
```

```
kcidb-db-cleanup
```

```
kcidb-db-complement
```

```
kcidb-db-dump
```

```
kcidb-db-init
```

```
kcidb-db-load
```

```
kcidb-db-query
```

```
kcidb-describe
```

```
nkondras@ruby:~$ kcidb-
```

```
kcidb-ingest
```

```
kcidb-merge
```

```
kcidb-monitor-spool-wipe
```

```
kcidb-mq-io-publisher
```

```
kcidb-mq-io-subscriber
```

```
kcidb-mq-pattern-publisher
```

```
kcidb-mq-pattern-subscriber
```

```
kcidb-notify
```

```
kcidb-oo-query
```

```
kcidb-query
```

```
kcidb-schema
```

```
kcidb-submit
```

```
kcidb-summarize
```

```
kcidb-tests-validate
```

```
kcidb-upgrade
```

```
kcidb-validate
```



KernelCI Python 3 Library

```
40
41 class Client:
42     """Kernel CI reporting client"""
43
44     def __init__(self, database=None, project_id=None, topic_name=None):
45         """
46         Initialize a reporting client
47
48         Args:
49             database:      The database specification string to use for
50                           accessing the report database.
51                           Can be None to have querying disabled.
52             project_id:    ID of the Google Cloud project hosting the message
53                           queue accepting submissions.
54                           Can be None to have submitting disabled.
55             topic_name:    Name of the message queue topic to publish
56                           submissions to. The message queue should be
57                           located within the specified Google Cloud project.
58                           Can be None, to have submitting disabled.
59
```



KernelCI PoC Reports

Groups

Your Groups

Find or Create a Group

Help

Nikolai Kondrashov

Home

Subscription

Messages

Hashtags

New Topic

New Poll

Chats

Directory

Calendar

Photos

Files

Databases

Wiki

kernelci-results-staging@groups.io

Search

" failed for "

Q

Tools

Relevance · Date

41 - 60 of 869

<

>

Tests failed for linux.git:master@228a65d4544a

Below is the summary of testing results Kernel CI database has recorded for this revision so far. See complete and up-to-date report at: <https://staging.kernelci.org:3000/d/revision/revision?orgId=1&v>
By staging.kernelci.org bot · #3566 · Jan 25

Builds failed for linux.git:master@fdb80bdc436

Below is the summary of testing results Kernel CI database has recorded for this revision so far. See complete and up-to-date report at: <https://staging.kernelci.org:3000/d/revision/revision?orgId=1&v>
By staging.kernelci.org bot · #3565 · Jan 24

Tests failed for linux.git:master@e68061375f79

Below is the summary of testing results Kernel CI database has recorded for this revision so far. See complete and up-to-date report at: <https://staging.kernelci.org:3000/d/revision/revision?orgId=1&v>
By staging.kernelci.org bot · #3564 · Jan 24

Tests failed for linux.git:master@... v5.11-rc4-202-gfe75a21824e7 on linus/master

Below is the summary of testing results Kernel CI database has recorded for this revision so far. See complete and up-to-date report at: <https://staging.kernelci.org:3000/d/revision/revision?orgId=1&v>
By staging.kernelci.org bot · #3561 · Jan 24

Builds failed for linux.git:master@e1ae4b0be158

Below is the summary of testing results Kernel CI database has recorded for this revision so far. See complete and up-to-date report at: <https://staging.kernelci.org:3000/d/revision/revision?orgId=1&v>
By staging.kernelci.org bot · #3559 · Jan 23

Builds failed for linux.git:master@fe75a21824e7

Below is the summary of testing results Kernel CI database has recorded for this revision so far. See complete and up-to-date report at: <https://staging.kernelci.org:3000/d/revision/revision?orgId=1&v>
By staging.kernelci.org bot · #3548 · Jan 23

Builds failed for linux.git:master@faba877b3bcc

Below is the summary of testing results Kernel CI database has recorded for this revision so far. See complete and up-to-date report at: <https://staging.kernelci.org:3000/d/revision/revision?orgId=1&v>
By staging.kernelci.org bot · #3547 · Jan 23

Builds failed for linux.git:master@83d09ad4b950

Below is the summary of testing results Kernel CI database has recorded for this revision so far. See complete and up-to-date report at: <https://staging.kernelci.org:3000/d/revision/revision?orgId=1&v>
By staging.kernelci.org bot · #3541 · Jan 22

Builds failed for linux.git:master@36ada2502635

Below is the summary of testing results Kernel CI database has recorded for this revision so far. See complete and up-to-date report at: <https://staging.kernelci.org:3000/d/revision/revision?orgId=1&v>
By staging.kernelci.org bot · #3540 · Jan 22



CKI

Checkouts

Issues

CONFIDENCE

Tests

Hosts

>

Search by Checkout, Pipeline or Job ID

Anonymous

Checkouts

Actions

Latest Checkouts

ID	Tree	Branch	Start Time	Valid	Builds	Tests
redhat:374324880	net-next	master	Sept. 21, 2021, 9:28 a.	✓		
redhat:374228059	stable	linux-5.14.y	Sept. 21, 2021, 7:01 a.	✓	4 ✓	
redhat:374223329	stable-queue	queue/5.14	Sept. 21, 2021, 6:51 a.	✓	4 ✓	
redhat:374111090	bpf-next	master	Sept. 21, 2021, 12:33 ;	✓	2 ✓	5 ✓
redhat:374097679	mainline.kernel	master	Sept. 20, 2021, 11:36	✓	2 ✗	
redhat:374097660	mainline.kernel	master	Sept. 20, 2021, 11:36	✓	1 ✗	3 ✓
redhat:374087799	kernel-ark	os-build	Sept. 20, 2021, 11:13	✓		
redhat:374033479	mainline.kernel	master	Sept. 20, 2021, 8:26 p.	✓	2 ✗	
redhat:374033441	mainline.kernel	master	Sept. 20, 2021, 8:26 p.	✓	1 ✗	3 ✓
redhat:374003946	stable-queue	queue/5.14	Sept. 20, 2021, 7:21 p.	✓	4 ✓	62 ✓



ISSUE

ResolvedInfra

selinux-policy: Cannot download Packages/d/dosfstools-4.2-2.fc35.x86_64.rpm: All mirrors were tried

List of Checkouts

ID	Tree	Branch	Start Time
redhat:361953890	mainline.kernel	master	Aug. 30, 2021, 5:21 p.m.
redhat:360321446	mainline.kernel	master	Aug. 26, 2021, 8:36 p.m.

ISSUE

ResolvedKernel Bug

BUG: non-zero pgtables_bytes on freeing mm: -4096

ark-eln

List of Checkouts

ID	Tree	Branch	Start Time
redhat:365427185	ark-rawhide-cl	ark-latest	Sept. 5, 2021, 11:26 a.m.

ISSUE

ResolvedUnstable Test

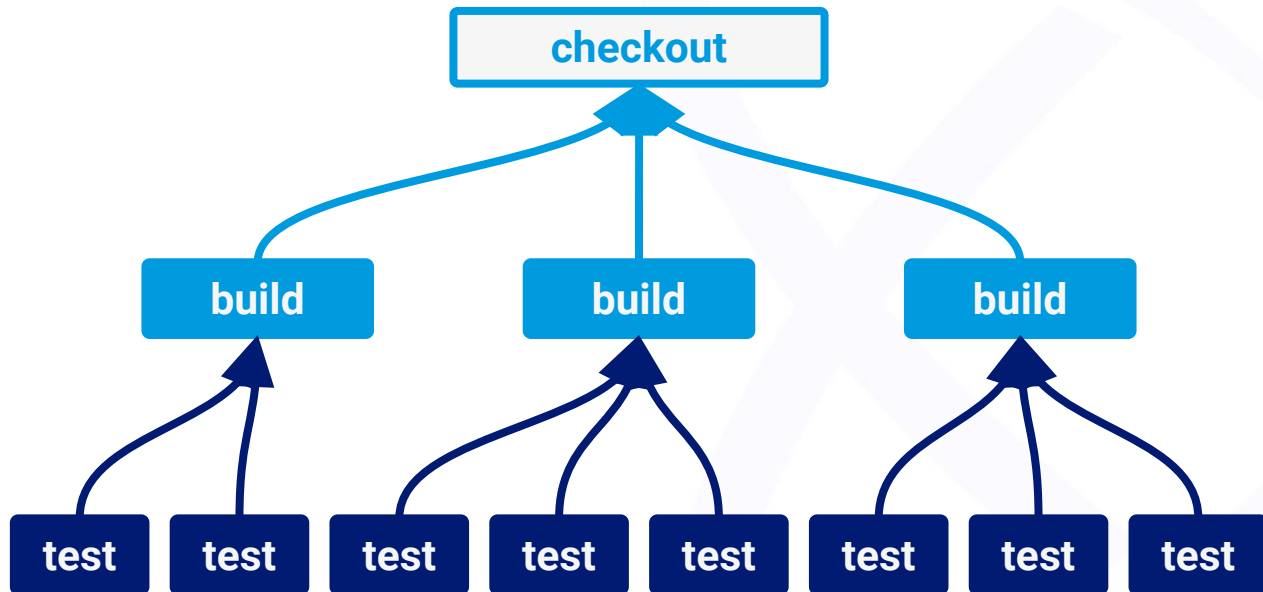
Podman test: no image found in manifest list for architecture arm64

List of Checkouts

ID	Tree	Branch	Start Time
----	------	--------	------------



I/O Schema







id

origin

git_repository_url

git_repository_branch

git_commit_hash

git_commit_name

patchset_files

patchset_hash

start_time

log_url

log_excerpt

valid

misc

comment



id

origin

checkout_id

architecture

compiler

command

config_name

config_url

input_files

output_files

start_time

duration

log_url

log_excerpt

valid

misc

comment





id

origin

build_id

path

environment

status

waived

output_files

start_time

duration

log_url

log_excerpt

misc

comment

Poll!

Submission Interface

```
nkondras@bard:~$ export GOOGLE_APPLICATION_CREDENTIALS=~/.kernelci-production-ci-cki.json
nkondras@bard:~$ kcidb-submit -p kernelci-production -t kernelci_new <<"EOF"
> {
>   "version": {
>     "major": 3,
>     "minor": 0
>   }
> }
> EOF
nkondras@bard:~$ █
```

```
#!/usr/bin/env python3

import kcidb

client = kcidb.Client(project_id="kernelci-production",
                      topic_name="kernelci_new")

report = dict(
    version=dict(major=3, minor=0)
)

client.submit(report)
```



KernelCI

kernelci-backed

```

265 lines (235 sloc)  8.77 KB
Raw  Blame  Edit  Delete

1  # Copyright (c) Collabora Limited 2019, 2020
2  # Author: Guillaume Tucker <tucker@collabora.com>
3
4  #
5  # This program is free software; you can redistribute it and/or modify it under
6  # the terms of the GNU Lesser General Public License as published by the Free
7  # Software Foundation; either version 2.1 of the License, or (at your option)
8  # any later version.
9
10 # This program is distributed in the hope that it will be useful, but WITHOUT
11 # ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS
12 # FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more
13 # details.
14
15 # You should have received a copy of the GNU Lesser General Public License
16 # along with this library; if not, write to the Free Software Foundation, Inc.,
17 # 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
18
19 """Push data using kcidb-submit."""
20
21 import json
22 import os
23 import subprocess
24 import urlparse
25
26 import models
27 import utils
28 from utils.report.common import DEFAULT_STORAGE_URL as STORAGE_URL
29
30 # Mapping between kcidb revision keys and build documents
31 BUILD_REV_KEY_MAP = {
32     'git_repository.url': models.GIT_URL_KEY,
33     'git_repository.commit_hash': models.GIT_COMMIT_KEY,
34     'git_repository.commit_name': models.GIT_DISCOURSE_V_KEY,
35     'git_repository.branch': models.GIT_BRANCH_KEY,
36 }
37
38 def make_id(raw_id, ns):
39     return '%s.%s' % (ns, str(raw_id))
40
41
42 def get_build_doc(group, db):
43     keys = [
44         models.JOB_KEY,
45         models.KERNEL_KEY,
46         models.GIT_BRANCH_KEY,
47         models.ARCHITECTURE_KEY,
48         models.DEFCONFIG_KEY,
49         models.BUILD_ENVIRONMENT_KEY,
50     ]
51     spec = {}
52     for k in keys:
53         spec[k] = group[k]
54     return utils.db.find_one(db[models.BUILD_COLLECTION], spec)
55
56
57 def get_test_cases(group, db, hierarchy, ns):
58     case_collection = db[models.TEST_CASE_COLLECTION]
59     group_collection = db[models.TEST_GROUP_COLLECTION]
60
61     hierarchy = hierarchy + [group[models.NAME_KEY]]
62     tests = [
63         {
64             'id': make_id(test[models.ID_KEY], ns),
65             'path': '%s.%s' % (hierarchy + [test[models.NAME_KEY]]),
66         }
67     ]

```

```

68     'status': test[models.STATUS_KEY],
69     # TODO: get start and duration times from LAVA log timestamps
70     'start_time': test[models.CREATED_KEY].isoformat(),
71 }
72
73 for test in (
74     utils.db.find_one(case_collection, test_id)
75     for test_id in group[models.TEST_CASES_KEY]
76 ):
77     pass
78
79 for sub_group_id in group[models.SUB_GROUPS_KEY]:
80     sub_group = utils.db.find_one(group_collection, sub_group_id)
81     tests += get_test_cases(sub_group, db, hierarchy, ns)
82
83 return tests
84
85 def submit(data, kcidb_options):
86     json_data = json.dumps(data, indent=2)
87     if kcidb_options.get('debug'):
88         utils.LOG.info('Submitting with kcidb:')
89         utils.LOG.info(json_data)
90     local_env = dict(os.environ)
91     local_env['MOOSE_APPLICATION_CREDENTIALS'] = kcidb_options['credentials']
92     kcidb_path = kcidb_options.get('kcidb_path', '')
93     kcidb_submit = os.path.join(kcidb_path, 'kcidb-submit')
94     p = subprocess.Popen([kcidb_submit,
95                          '-p', kcidb_options['project'],
96                          '-t', kcidb_options['topic'],
97                          'stdin=subprocess.PIPE',
98                          env=local_env])
99     p.communicate(input=json_data)
100     if p.returncode:
101         utils.LOG.warn("Failed to push data kcidb")
102
103 def push_build(build_id, first, kcidb_options, db_options={}, db=None):
104     if db is None:
105         db = utils.db.get_db_connection(db_options)
106     origin = kcidb_options.get('origin', 'kernelci.org')
107     ns = kcidb_options.get('namespace', 'kernelci.org')
108     build = utils.db.find_one(db[models.BUILD_COLLECTION], build_id)
109     build_id = make_id(build[models.ID_KEY], ns)
110     revision_id = build[models.GIT_COMMIT_KEY]
111
112     kcidb_data = {
113         'version': '1',
114     }
115
116     if first:
117         kcidb_revision = {
118             'origin': origin,
119             'origin_id': revision_id,
120             'valid': True,
121             'discovery_time': build[models.CREATED_KEY].isoformat(),
122         }
123     kcidb_revision.update({
124         kcidb_key: build[build_key]
125         for kcidb_key, build_key in BUILD_REV_KEY_MAP.items()
126     })
127     kcidb_data['revisions'] = [kcidb_revision]
128
129     files = {
130         'kernel_image': build[models.KERNEL_IMAGE_KEY],
131         'modules': build[models.MODULES_KEY],
132         'system_map': build[models.SYSTEM_MAP_KEY],
133     }

```

```

134     }
135     output_files = []
136     for name, file in files.items():
137         url = '%s.%s' % (
138             STORAGE_URL,
139             build[models.FILE_SERVER_RESOURCE_KEY],
140             file)
141         output_files.append({'name': name, 'url': url})
142
143     kcidb_build = {
144         'revision_origin': origin,
145         'revision_origin_id': revision_id,
146         'origin': origin,
147         'origin_id': build_id,
148         'valid': build[models.CREATED_KEY].isoformat(),
149         'start_time': build[models.CREATED_KEY].isoformat(),
150         'description': build[models.GIT_DISCOURSE_V_KEY],
151         'duration': build[models.BUILD_TIME_KEY],
152         'architecture': build[models.ARCHITECTURE_KEY],
153         'compiler': build[models.COMPILER_VERSION_FULL_KEY],
154         'log_url': urlparse.urljoin(
155             STORAGE_URL,
156             '%s.%s' % (
157                 build[models.FILE_SERVER_RESOURCE_KEY],
158                 build[models.BUILD_LOG_FILE],
159             ))
160     }
161     'config_name': build[models.DEFCONFIG_FULL_KEY],
162     'config_url': urlparse.urljoin(
163         STORAGE_URL,
164         '%s.%s' % (
165             build[models.FILE_SERVER_RESOURCE_KEY],
166             build[models.KERNEL_CONFIG_KEY],
167         ))
168     }
169     'output_files': output_files,
170     'misc': {
171         'kernel_image_size': build[models.KERNEL_IMAGE_SIZE_KEY],
172         'vmlinux_bin_size': build[models.VMLINUX_BIN_SIZE_KEY],
173         'vmlinux_data_size': build[models.VMLINUX_DATA_SIZE_KEY],
174         'build_platform': build[models.BUILD_PLATFORM_KEY],
175     },
176 }
177 kcidb_data['builds'] = [kcidb_build]
178
179 _submit(kcidb_data, kcidb_options)
180
181 def get_test_misc(group, test):
182     misc = {
183         'plan': group[models.NAME_KEY],
184         'plan_variant': group[models.PLAN_VARIANT_KEY],
185         'kernelci_status': test['status']
186     }
187     return misc
188
189 def push_test(group_id, kcidb_options, db_options={}, db=None):
190     status_translate = {
191         'UNKNOWN': 'SKIP'
192     }
193     if db is None:
194         db = utils.db.get_db_connection(db_options)
195     collection = db[models.TEST_GROUP_COLLECTION]
196     group = utils.db.find_one(collection, group_id)
197     origin = kcidb_options.get('origin', 'kernelci')

```

```

198     ns = kcidb_options.get('namespace', 'kernelci.org')
199     test_cases = get_test_cases(group, db, [], ns)
200     build = get_build_doc(group, db)
201     if not build:
202         utils.LOG.warn("kcidb: Missing build, unable to push tests.")
203         return
204     build_id = make_id(build[models.ID_KEY], ns)
205     env_description = "%s in [%s].format(
206         group[models.DEVICE_TYPE_KEY],
207         group[models.LAB_NAME_KEY]
208     )
209
210 test_description = "%s on [%s] in [%s].format(
211     group[models.NAME_KEY],
212     group[models.DEVICE_TYPE_KEY],
213     group[models.LAB_NAME_KEY]
214 )
215
216 env_misc = {
217     'device': group[models.DEVICE_TYPE_KEY],
218     'lab': group[models.LAB_NAME_KEY],
219     'mach': group[models.MACH_KEY],
220     'rootfs_url': group[models.INITRD_KEY],
221     'instance': group[models.BOARD_INSTANCE_KEY],
222 }
223
224 files = {
225     'test': group[models.BOOT_LOG_KEY],
226     'test': group[models.BOOT_LOG_VML_KEY],
227     'lava_json': '%s.%s' % (group[models.DEVICE_TYPE_KEY],
228                             group[models.LAB_NAME_KEY]),
229 }
230
231 output_files = []
232 for name, file in files.items():
233     url = urlparse.urljoin(
234         STORAGE_URL,
235         '%s.%s' % (
236             build[models.FILE_SERVER_RESOURCE_KEY],
237             group[models.LAB_NAME_KEY],
238             file,
239         ))
240     output_files.append({'name': name, 'url': url})
241
242 kcidb_data = {
243     'version': '1',
244     'tests': [
245         {
246             'build_origin': origin,
247             'build_origin_id': build_id,
248             'origin': origin,
249             'origin_id': test['id'],
250             'environment': {
251                 'description': env_description,
252                 'misc': env_misc,
253             },
254             'path': test['path'],
255             'description': test_description,
256             'status': status_translate.get(test['status'], test['status']),
257             'passed': False,
258             'start_time': test['start_time'],
259             'output_files': output_files,
260             'misc': get_test_misc(group, test)
261         }
262     ]
263     for test in test_cases
264 }
265 _submit(kcidb_data, kcidb_options)

```



kernelci-backend

```

285 re = re.compile(options.get("namespace"), "recompile")
286 test_cases = get_test_cases(group, db, [ ], ns)
287 build = get_build_description(group, db, [ ], ns)
288 if not build:
289     util.raise_with("rebuild: Missing build, unable to push tests.")
290
291 util.raise_with("rebuild: Missing build, unable to push tests.")
292
293 build_id = _make_id(build[models.ID_KEY], ns)
294 return {
295     'description': "I'm in {}".format(
296         group[models.DESCR_KEY],
297         group[models.LAB_NAME_KEY]
298     )
299 }
300
301 test_description = "%i on %i (%i)".format(
302     group[models.DESCR_KEY],
303     group[models.DESCR_KEY],
304     group[models.LAB_NAME_KEY]
305 )
306
307 return {
308     'env_vars': {
309         'device': group[models.DEVICE_TYPE_KEY],
310         'lab': group[models.LAB_NAME_KEY],
311         'task': group[models.TASK_KEY],
312         'roots_dir': group[models.ROOTS_KEY],
313         'instance': group[models.BOARD_INSTANCE_KEY],
314     }
315 }
316
317 files = {
318     'text': group[models.ROOTS_KEY],
319     'html': group[models.ROOTS_KEY],
320     'device': group[models.DEVICE_TYPE_KEY],

```

265 lines (235 sloc)	8.77 KB
----------------------	---------

```

77 KB
242 output_directory = os.path.join(
243     kicad_data = {
244         'version': '1',
245         'tests': [
246             {
247                 'build_origin': origin,
248                 'build_origin_id': build_id,
249                 'origin': origin,
250                 'origin_id': test['id'],
251                 'environment': {
252                     'description': env.description,
253                     'misc': env.misc,
254                 },
255                 'path': test['path'],
256                 'description': test.description,
257                 'status': status_translate.get(test['
258                     'status': false,
259                     'start_time': test['start_time'],
260                     'output_files': output_files,
261                     'misc': _get_test_misc(group, test)
262                 }
263             }
264         ]
265     }
266     )
267     submit(kicad_data, kicad_options)

```



KernelCI GKernelCI/Ghelper

```
nkondras@ruby:~/projects/github.com/GKernelCI/Ghelper/kcidb$ wc -l *
```

136	create_object.py
0	get_kernel_hash.py
5	get_kernel_hash.sh
18	get_patches.py
5	get_patch_hash.sh
3	requirements.txt
12	send.sh
14	sendtokcidb
193	total

```
nkondras@ruby:~/projects/github.com/GKernelCI/Ghelper/kcidb$
```



KernelCI GKernelCI/Ghelper

```
nkondras@ruby:~/projects/github.com/GKernelCI/Ghelper/kcidb$ wc -l *
```

136	create_object.py
0	get_kernel_hash.py
5	get_kernel_hash.sh
18	get_patches.py
5	get_patch_hash.sh
3	requirements.txt
12	send.sh
14	sendtokcidb
193	total

```
nkondras@ruby:~/projects/github.com/GKernelCI/Ghelper/kcidb$
```



```

    err := kcid.Validate(data); err != nil {
        return err
    }
    }, err = c.Topic.Publish(c.ctx, Apubsub.Message(data)).Get(c.ctx)
    return err
}

// validate bool

kcid.Validate(data []byte) error {
    if !Validate {
        return nil
    }

    bin := "kcid-validate"
    if const err := exec.Command(bin).Run(); err != nil {
        fmt.Fprintf(os.Stderr, "%v is not found\n", bin)
        return nil
    }

    cmd := exec.Command(Sider)
    cmd.Stdin = bytes.NewReader(data)
    output, err := cmd.CombinedOutput()
    if err != nil {
        return fmt.Errorf("%v failed (%v) on %v\n\nerror: %s",
            bin, err, data, output)
    }

    return nil
}

// client convert(target *targets.Target, bug *dashapi.BugReport) *Kcid {
res := &Kcid{
    Version: &Version{
        Major: 3,
        Minor: 0,
    },
    Revisions: []*Revision{
        {
            Origin:      c.origin,
            ID:          bug.KernelCommit,
            GitRepositoryURL: normalizeRepo(bug.KernelRepo),
            GitCommitHash: bug.KernelCommit,
            GitRepositoryBranch: bug.KernelBranch,
            Description:   bug.KernelCommitTitle,
            PublishingTime: bug.KernelCommitDate.Format(time.RFC3339),
            DiscoverTime:   bug.PublishTime.Format(time.RFC3339),
            Valid:         true,
        }
    },
    Builds: []*Build{
        {
            Origin:      c.origin,
            ID:          c.exidID(bug.BuildID),
            RevisionID:  bug.KernelCommit,
            Architecture: target.KernelArch,
            Compiler:   bug.CompilerID,
            StartTime:  bug.BuildTime.Format(time.RFC3339),
            ConfigURL:  bug.KernelConfigLink,
            Valid:      true,
        }
    },
}

if strings.Contains(bug.Title, "build error") {
    build := res.Builds[0]
    build.Valid = false
}
}

```

32



KernelCI

syzkaller/client.go

```

186 lines (174 sloc) 5.25 KB
1 // Copyright 2020 syzkaller project authors. All rights reserved.
2 // Use of this source code is governed by Apache 2 LICENSE that can be found in the LICENSE file.
3
4 package kcidb
5
6 import {
7     "bytes"
8     "context"
9     "encoding/json"
10    "fmt"
11    "os"
12    "os/exec"
13    "strings"
14    "time"
15
16    "cloud.google.com/go/pubsub"
17    "github.com/google/syzkaller/dashboard/dashapi"
18    "github.com/google/syzkaller/sys/targets"
19    "google.golang.org/api/option"
20 }
21
22 type Client struct {
23     ctx context.Context
24     origin string
25     client *pubsub.Client
26     topic *pubsub.Topic
27 }
28
29 // NewClient creates a new client to send
30 // // Origin is how this system identified itself
31 // // Project is Kcidb GCE project name, e.g
32 // // Topic is pubsub topic to publish messages
33 // // Credentials is Google application credentials
34 func NewClient(ctx context.Context, origin string, project string, topic string, credentials []byte) (*Client, error) {
35     client, err := pubsub.NewClient(ctx, project, option.WithCredentials(credentials))
36     if err != nil {
37         return nil, err
38     }
39     c := &Client{
40         ctx: ctx,
41         origin: origin,
42         client: client,
43         topic: client.Topic(topic),
44     }
45     return c, err
46 }
47
48 func (c *Client) Close() error {
49     c.topic.Stop()
50     return c.client.Close()
51 }
52
53 func (c *Client) Publish(bug *dashapi.BugReport) error {
54     target := targets.List[bug.OS][bug.VMArch]
55     if target == nil {
56         return fmt.Errorf("unsupported OS/arch %v/%v", bug.OS, bug.VMArch)
57     }
58     data, err := json.MarshalIndent(c.convert(target, bug), "", " ")
59     if err != nil {
60         return fmt.Errorf("failed to marshal kcidb json: %v", err)
61     }

```

```

62     if err := kcidbValidate(data); err != nil {
63         return err
64     }
65     _, err = c.topic.Publish(c.ctx, &pubsub.Message{Data: data}).Get(c.ctx)
66     return err
67 }
68
69 var Validate bool
70
71 func kcidbValidate(data []byte) error {
72     if !Validate {
73         return nil
74     }
75     const bin = "kcidb-validate"
76     if _, err := exec.LookPath(bin); err != nil {
77         fmt.Fprintf(os.Stderr, "%v is not found\n", bin)
78         return nil
79     }
80     cmd := exec.Command(bin)
81     cmd.Stdin = bytes.NewReader(data)
82     output, err := cmd.CombinedOutput()
83     if err != nil {
84         return fmt.Errorf("%v failed (%v) on %v\n%v\nerror: %s",
85             bin, err, data, output)
86     }
87 }
88
89 ID: bug.KernelCommit,
90 GitRepositoryURL: normalizeRepo(bug.KernelRepo),
91 GitCommitHash: bug.KernelCommit,
92 GitRepositoryBranch: bug.KernelBranch,
93 Description: bug.KernelCommitTitle,
94 PublishingTime: bug.KernelCommitDate.Format(time.RFC3339),
95 DiscoverTime: bug.BuildTime.Format(time.RFC3339),
96 Valid: true,
97 },
98 },
99 BuildID: []BuildID{
100     {
101         Origin: c.origin,
102         ID: c.extID(bug.BuildID),
103         RevisionID: bug.KernelCommit,
104         Architecture: target.KernelArch,
105         CompilerID: bug.CompilerID,
106         StartTime: bug.BuildTime.Format(time.RFC3339),
107         ConfigURL: bug.KernelConfigLink,
108         Valid: true,
109     },
110 },
111 },
112 if strings.Contains(bug.Title, "build error") {
113     build := res.BuildID[0]
114     build.Valid = false
115 }

```

```

125 build.LogURL = bug.LogLink
126 build.Misc = &BuildMisc{
127     OriginURL: bug.Link,
128     ReportedBy: bug.CreditEmail,
129 }
130 } else {
131     var outputFiles []Resource
132     if bug.ReportLink != "" {
133         outputFiles = append(outputFiles, &Resource{Name: "report.txt", URL: bug.ReportLink})
134     }
135     if bug.LogLink != "" {
136         outputFiles = append(outputFiles, &Resource{Name: "log.txt", URL: bug.LogLink})
137     }
138     if bug.ReproLink != "" {
139         outputFiles = append(outputFiles, &Resource{Name: "repro.c", URL: bug.ReproLink})
140     }
141     if bug.ReproSyzLink != "" {
142         outputFiles = append(outputFiles, &Resource{Name: "repro.syz.txt", URL: bug.ReproSyzLink})
143     }
144     if bug.MachineInfoLink != "" {
145         outputFiles = append(outputFiles, &Resource{Name: "machine_info.txt", URL: bug.MachineInfoLink})
146     }
147     causeRevisionID := ""
148     if bug.BisectCause != nil && bug.BisectCause.Commit != nil {
149         causeRevisionID = bug.BisectCause.Commit.Hash
150     }
151 }
152
153 Misc: &BuildMisc{
154     OriginURL: bug.Link,
155     ReportedBy: bug.CreditEmail,
156     UserSpaceArch: bug.UserSpaceArch,
157     CauseRevisionID: causeRevisionID,
158 },
159 },
160 return res
161 }
162
163 func normalizeRepo(repo string) string {
164     // Kcidb needs normalized repo addresses to match reports from different
165     // origins and with subscriptions. "https:" is always preferred over "git:"
166     // where available. Unfortunately we don't know where it's available
167     // and where it isn't. We know that "https:" is supported on kernel.org
168     // and that's the main case we need to fix up. "https:" is always used
169     // for github.com and googlesource.com.
170     return strings.Replace(repo, "git://git.kernel.org", "https://git.kernel.org", -1)
171 }
172
173 func (c *Client) extID(id string) string {
174     return c.origin + "-" + id
175 }

```

186 lines (174 sloc)

5.25 KB

432 lines (324 sloc) | 13.6 KB

Raw Blame

How to submit reports with KCIDB

Here's what you need to do.

1. Get submission credentials and parameters.
2. Install KCIDB.
3. Generate some report data.
4. Submit report data.
5. Go to 3, if you have more.

You don't need to run a daemon, just execute command-line tools (or use the Python 3 library, if you're feeling fancy).

1. Get submission credentials and parameters

Write to kernelci@groups.io, introduce yourself, and explain what you want to submit (better, show preliminary report data). Once your request is approved, you will get a JSON credentials file, which you can use to authenticate yourself with KCIDB tools/library.

Export the file location (here `~/.kcldb-credentials.json`) into environment like this:

```
export GOOGLE_APPLICATION_CREDENTIALS=~/.kcldb-credentials.json
```

You will also need to specify some or all of the following to the tools/library (using current production parameters from here on):

- Google Cloud project: `kernelci-production`
- Submission queue topic: `kernelci_new`
- Dataset name: `kernelci04`

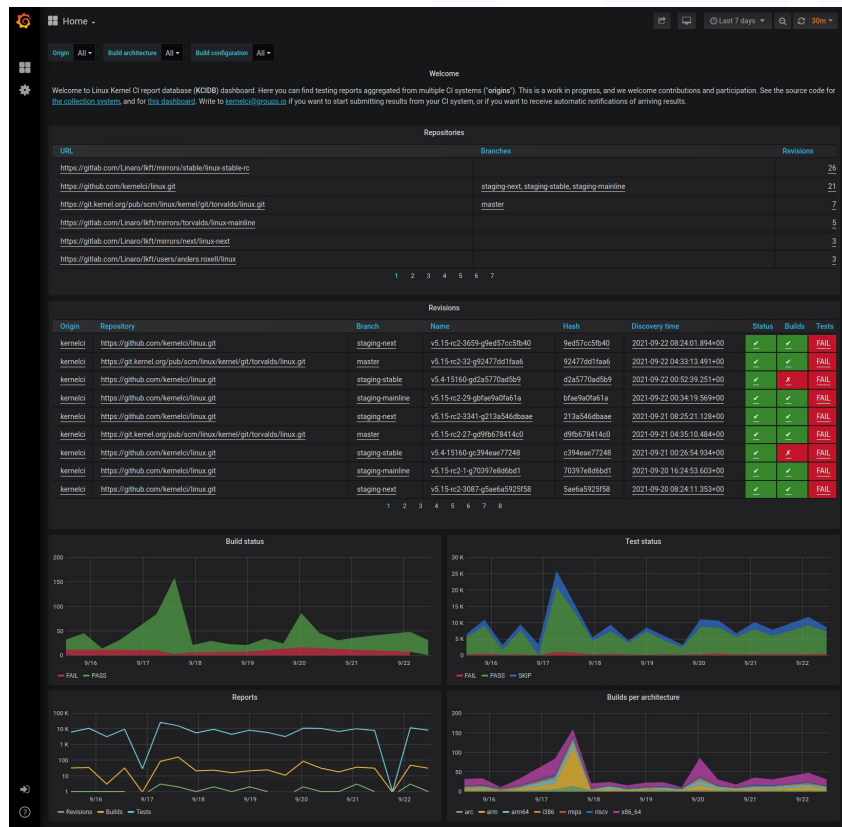
Finally, we'll need to agree on the "origin" string identifying your system among other submitters. We'll use `submitter` in examples below.

2. Install KCIDB

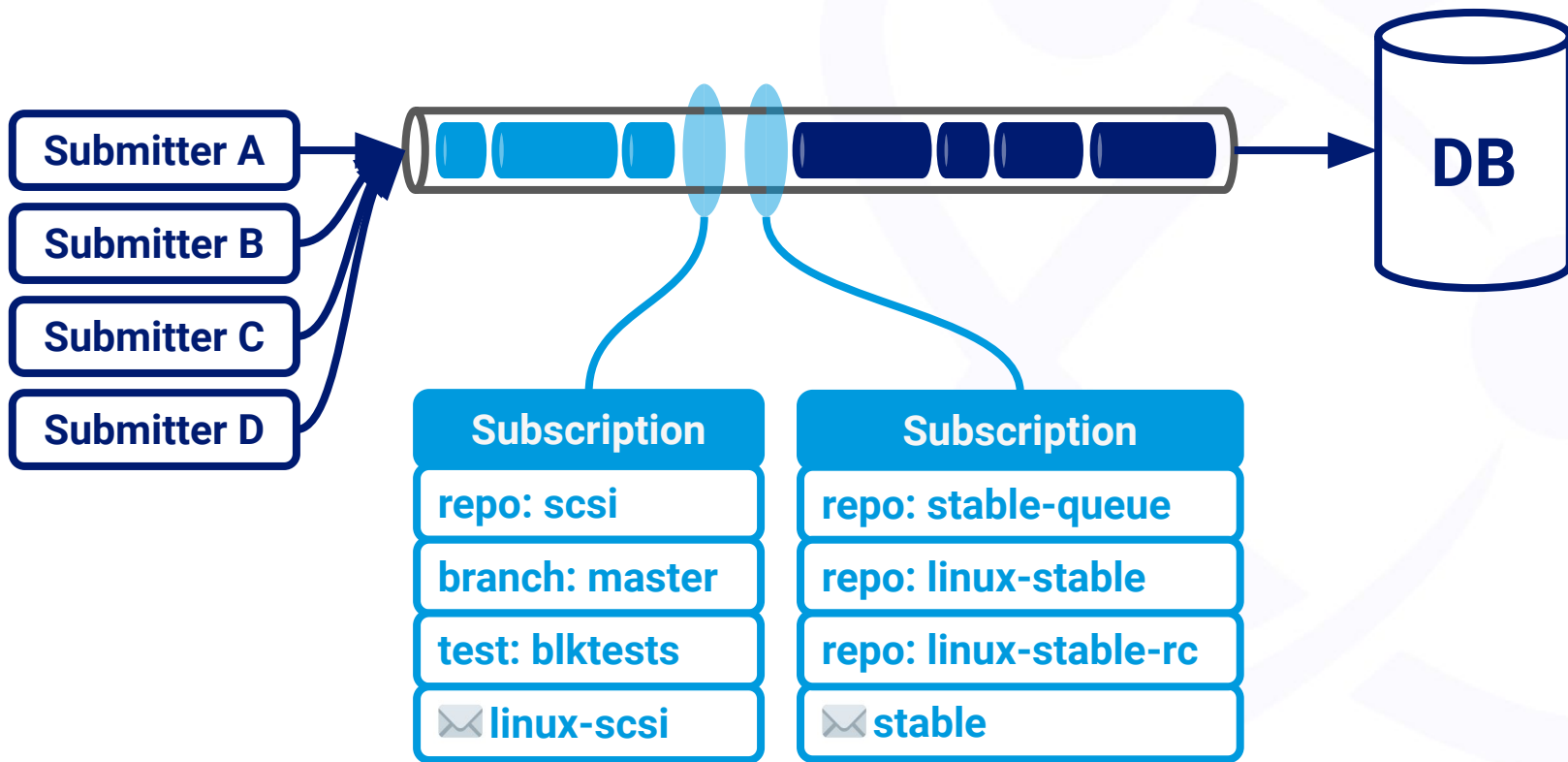
Pick a KCIDB [release](#). Latest would normally be best.

Let's say you picked v8. Run this:

KernelCI Playground



Subscriptions and Notifications





KernelCI Subscription Example

22 lines (18 sloc) | 798 Bytes

Raw

Blame



```
1  """Mainline tree subscriptions"""
2
3  from kcidb.monitor.output import NotificationMessage as Message
4
5  REPO_URL = \
6      "https://git.kernel.org/pub/scm/linux/kernel/git/torvalds/linux.git"
7  RECIPIENTS = ["Linux Kernel Mailing List <linux-kernel@vger.kernel.org>"]
8
9
10 def match_revision(revision):
11     """Match revisions of interest to stable tree developers"""
12     if REPO_URL not in revision.repo_branch_checkouts:
13         return ()
14     if not revision.builds_valid:
15         return (Message(RECIPIENTS, "Builds failed for "),)
16     for test in revision.tests:
17         # Ignore syzbot until we have issues/incidents
18         if test.origin == "syzbot":
19             continue
20         if test.status == "FAIL" and not test.waived:
21             return (Message(RECIPIENTS, "Tests failed for "),)
22     return ()
```

Below is the summary of results Kernel CI database has recorded for this revision so far. See complete and up-to-date report at:

https://staging.kernelci.org:3000/d/checkout/checkout?orgId=1&var-git_commit_hash=7a

OVERVIEW

Builds: ❌ FAIL
Tests: ❌ FAIL

REVISION

Status

✅ PASS

Commit

name: v5.12-rc1-19-g7a7fd0de4a98

hash: 7a7fd0de4a9804299793e564a555a49c1fc924cb

Checked out from

<https://git.kernel.org/pub/scm/linux/kernel/git/torvalds/linux.git> master

By

kernelci, redhat, syzbot, tuxsuite

BUILDS

Status

✗ 1 ✓ 858

Architectures

arm	✗ 1	✓ 331
arm64	—	✓ 116
x86_64	—	✓ 115
mips	—	✓ 85
powerpc	—	✓ 62
riscv	—	✓ 36
i386	—	✓ 32
s390	—	✓ 29
sh	—	✓ 18
arc	—	✓ 10
sparc	—	✓ 9
parisc	—	✓ 9
aarch64	—	✓ 2
ppc64le	—	✓ 2
s390x	—	✓ 2

Failures

✗ 1 arm allnoconfig

By

kernelci, redhat, syzbot, tuxsuite

TESTS

Status

✗ 15 ✓ 27 ✗ 12 ✗ 7 ➡ 3

Failures

✗ baseline
✗ ltp-crypto
✗ baseline-nfs
✗ usb
✗ sleep
...

By

kernelci, redhat, syzbot

See complete and up-to-date report at:

https://staging.kernelci.org:3000/d/checkout/checkout?orgId=1&var-git_commit_hash=7a

LEGEND

- ✗ ERROR - Aborted. Test, tested code, or both might be faulty.
- ✗ FAIL - Failed. Tested code is likely faulty.
- ✓ PASS - Passed. Tested code is likely correct.
- OK DONE - Finished. Status of tested code is unknown.
- ➡ SKIP - Skipped. Planned, but didn't execute.
- ? UNKNOWN - In progress, or status unknown.
- ✗ WAIVED - Waived result. Test is too new or shows known failures.
- BLANK - No data, zero.

Thoughts?
Questions?

Join Us!



KernelCI

Main repos on GitHub

Repositories 41

Packages

People 22

Teams 2

Projects 1

Type: All

Language: All

New

3 results for repositories matching **kcidb**

Clear filter

kcidb

kernelci.org common database tools

Python

GPL-2.0

7

5

38

1

Updated 6 days ago

kcidb-io

kernelci.org common reporting - I/O data library

Python

2

1

7

2

Updated 8 days ago

kcidb-grafana

KCIDB Grafana docs and files

1

2

5

0

Updated on Jan 13

Top languages

Python

Shell

HTML

Makefile

SaltStack

People 22



is:issue is:open label:"good first issu" / Pull requests Issues Marketplace Explore

Repositories 2

Code 0

Commits 0

Issues 8

Discussions Beta 0

Packages 107K

Marketplace 0

Topics 815K

Wikis 0

Users 0

States

Open 8

Languages

Python 8

Advanced search Cheat sheet

8 issues or view all results on GitHub

kernelci/kcidb #157

[bigquery: Add and use a single function for queries](#)

To reduce code duplication implement a single querying function to be called by dump()/query()/oo_query()/etc. in the BigQuery database code. Add DEBUG-level logging of queries and their parameters.

good first issue

spbnick

kernelci/kcidb-io #9

[Mark all symbols not supposed to be exposed as private](#)

Figure out the correct way for, and mark all symbols not explicitly supposed to be exposed as private to avoid unintended use and unnecessary coupling.

good first issue

spbnick

kernelci/kcidb #111

[Upgrade google-cloud-secret-manager dependency](#)

The latest release of the google-cloud-secret-manager package breaks the existing KCIDB code, and was pinned to versions <2.0.0dev. Fix the incompatibilities and unpin the package. Or perhaps re-pin ...

good first issue

spbnick

kernelci/kcidb #110

[Upgrade google-cloud-pubsub dependency](#)

The latest release of the google-cloud-pubsub package breaks the existing KCIDB code, and was pinned to versions <2.0.0dev. Fix the incompatibilities and unpin the package. Or perhaps re-pin to <3.0.0dev. ...

good first issue

 Groups.io

Your Groups ▾

Find or Create a Group

Help

Nikolai Kondrashov ▾

Home

Subscription

Messages

Hashtags

New Topic

New Poll

Chats

Directory

Calendar

Photos

Files

Databases

Wiki



KernelCI

KernelCI

kernelci@groups.io

Discussion mailing list for the KernelCI project and kernelci.org

See also the [wiki](#) on [GitHub](#)

Group Information

 <https://kernelci.org>

 75 Members

 331 Topics, Last Post: Feb 9

 Started on 17.03.2018

 [Feed](#)

Group Email Addresses

Post: kernelci@groups.io

Subscribe: kernelci+subscribe@groups.io

Unsubscribe: kernelci+unsubscribe@groups.io

Group Owner: kernelci+owner@groups.io

Help: kernelci+help@groups.io

Top Hashtags

[\[See All\]](#)

- [#minutes](#) kernelci.org team meeting minutes 61
- [#KCI0B](#) 3

Group Settings

-  All members can post to the group.
-  Posts to this group do not require approval from the moderators.
-  Posts from new users require approval from the moderators.
-  Messages are set to reply to group and sender.
-  Subscriptions to this group do not require approval from the moderators.
-  Archive is visible to anyone.
-  Wiki is visible to members only.
-  Members cannot edit their messages.
-  Members can set their subscriptions to no email.

 Groups.io

 Your Groups

 Find or Create a Group

 Help

 Nikolai Kondrashov

 Home Subscription Messages Hashtags New Topic New Poll Chats Directory Calendar Photos Files Databases Wiki

KernelCI

KernelCI

kernelci@groups.io

Discussion mailing list for the KernelCI project and kernelci.org

See also the [wiki on GitHub](#)

Group Information

 <https://kernelci.org>

 75 Members

 331 Topics, Last Post: Feb 9

 Started on 17.03.2018

 Feed

Group Email Addresses

Post: kernelci@groups.io

Subscribe: kernelci+subscribe@groups.io

Unsubscribe: kernelci+unsubscribe@groups.io

Group Owner: kernelci+owner@groups.io

Help: kernelci+help@groups.io

Top Hashtags

[\[See All\]](#)

-  [kernelci.org team meeting minutes](#) 61
-  [KCIIDB](#) 3

Group Settings

-  All members can post to the group.
-  Posts to this group do not require approval from the moderators.
-  Posts from new users require approval from the moderators.
-  Messages are set to reply to group and sender.
-  Subscriptions to this group do not require approval from the moderators.
-  Archive is visible to anyone.
-  Wiki is visible to members only.
-  Members cannot edit their messages.
-  Members can set their subscriptions to no email.

Group Email Addresses

Post: kernelci@groups.io

Subscribe: kernelci+subscribe@groups.io

Unsubscribe: kernelci+unsubscribe@groups.io

Post: kernelci@groups.io
Subscribe: kernelci+subscribe@groups.io
Unsubscribe: kernelci+unsubscribe@groups.io
Group Owner: kernelci+owner@groups.io
Help: kernelci+help@groups.io

Top Hashtags [\[See All\]](#)

[#minutes](#) kernelci.org team meeting minutes **61**
[#KCI0B](#) **3**

- Members cannot edit their messages.
- Members can set their subscriptions to no email.

#kernelci at libera.chat

#kernelci The upstream kernel validation project: <http://kernelci.org> | Mailing list: <https://lists.linaro.org/mailman...> 71

Tuesday, February 16th, 2021 about 21 hours ago

A **alicef** none
spbnick: sending it 21:31
→ FFY00 nipped out 22:41

Wednesday, February 17th, 2021

→ sestebanz and kverz_broke joined ← alexandra, shuah and kveremitz quit → selvak1631189 popped in → csd nipped out 01:10

S **spbnick** Nikolai Kondrashov
alicef: Yep, I see it now, awesome :D 10:52
→ mwasilew joined ← veremitz quit → selvak1631189, gaulishcoin9 and gaulishcoin3 popped in • kverz_broke → kveremitz 10:57

S **spbnick** Nikolai Kondrashov
Everyone who's around, say hi for a channel screenshot for my talk on Friday 🙌! 12:19

G **gtucker** Guillaume Tucker
o/ KernelCI rules :) 12:33
→ vkabatova joined (~vkabatova@176-74-128-122.netdatacomm.cz) 12:40

M **montjoie**
hello 12:49
→ monstr joined 15:50

S **spbnick** 18:26

Thank you

