Fuzzing Linux Drivers with Syzkaller

Ricardo Cañuelo

ricardo.canuelo@collabora.com

Open First

Overview

- Fuzzing 101
  - Why it is a valuable kernel development tool
  - About Syzkaller
- Our goal: fuzzing kernel drivers
  - Tweaking Syzkaller
- Getting results

Fuzzing as a test tool

- There are many approaches to software testing
- Different techniques, different goals
- Fuzzing tries to uncover bugs by reaching execution paths that hard to cover with manual tests
- Produces semi-random inputs and code sequences automatically

Fuzzers: key features

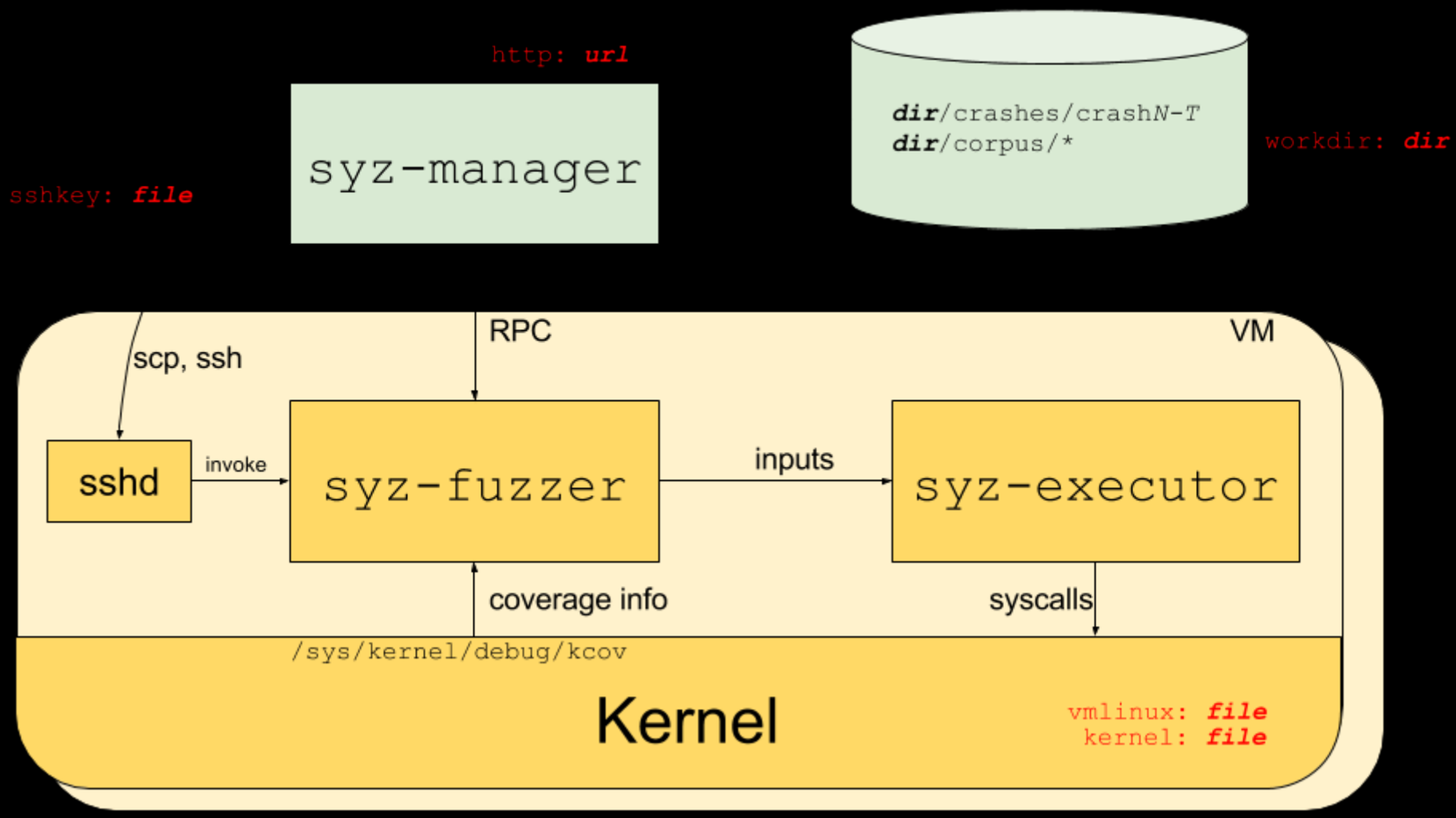
- Check code coverage
- Knowledge about the code base
- Smart data & code generation
- Useful reports (ideally with reproducers)

Syzkaller

https://github.com/google/syzkaller

- Coverage-guided
- Uses many kernel debugging features
  - Kcov
  - Sanitizers
  - Fault injectors
- Great bug reporting
- Syzbot

Syzkaller: how it works



Syzkaller: how it works

Syzlang: system call description language

```
resource fd[int32]: -1
resource fd_dir[fd]: AT_FDCWD

open(file ptr[in, filename], flags flags[open_flags], mode flags[open_mode]) fd
open$dir(file ptr[in, filename], flags flags[open_flags], mode flags[open_mode]) fd_dir
close(fd fd)
read(fd fd, buf buffer[out], count len[buf])

open_flags = O_WRONLY, O_RDWR, O_APPEND, FASYNC, O_CLOEXEC, O_CREAT, O_DIRECT,
O_DIRECTORY, O_EXCL, O_LARGEFILE, O_NOATIME, O_NOCTTY, O_NOFOLLOW, O_NONBLOCK, O_PATH,
O_SYNC, O_TRUNC, __O_TMPFILE
open_mode = S_IRUSR, S_IWUSR, S_IXUSR, S_IRGRP, S_IWGRP, S_IXGRP, S_IROTH, S_IWOTH,
S_IXOTH
```

Syzkaller: how it works

Test configuration

```
{
    "target": "linux/amd64",
    "http": "myhost.com:56741",
    "workdir": "/syzkaller/workdir",
    "kernel_obj": "/linux/",
    "image": "/linux_image/wheezy.img",
    "sshkey": "/linux_image/ssh/id_rsa",
    "syzkaller": "/syzkaller",
    "disable_syscalls": ["keyctl", "add_key", "request_key"],
    "suppressions": ["some known bug"],
    "procs": 4,
    "type": "qemu",
    "vm": {
        "count": 16,
        "cpu": 2,
        "mem": 2048,
        "kernel": "/linux/arch/x86/boot/bzImage",
        "initrd": "linux/initrd"
    }
}
```

Running it

```
$ syzkaller_dir/bin/syz-manager -config=my_config.cfg
```

Syzkaller: how it works

syzkaller

Stats:

revision	baca2611
config	
uptime	3m45s
fuzzing	2m0s
corpus	47
triage queue	0
cover	3275
signal	4089
syscalls	22
crash types	0 (0/hour)
crashes	0 (0/hour)
exec candidate	94 (36/min)
exec fuzz	9 (209/hour)
exec gen	0 (0/hour)
exec hints	0 (0/hour)
exec minimize	14 (325/hour)
exec seeds	0 (0/hour)
exec smash	0 (0/hour)
exec total	505 (195/min)
exec triage	388 (150/min)
executor restarts	5 (116/hour)
max signal	4178 (26/sec)
new inputs	122 (47/min)
rotated inputs	0 (0/hour)
suppressed	0 (0/hour)
vm restarts	1 (23/hour)

Crashes:

Description	Count	Last Time	Report

Log:

```
2020/09/28 11:15:20 devlink PCI setup       : PCI device 0000:00:10.0 is not available
2020/09/28 11:15:20 USB emulation           : /dev/raw-gadget does not exist
2020/09/28 11:15:20 corpus                  : 47 (deleted 0 broken)
```

Preparing the target machine

- Create the root filesystem image:
  - tools/create-image.sh
- Build the kernel
  - CONFIG_KCOV
  - CONFIG_KCOV_INSTRUMENT_ALL=y
  - CONFIG_KCOV_ENABLE_COMPARISONS=y
  - CONFIG_DEBUG_FS=y
  - CONFIG_DEBUG_INFO=y
  - CONFIG_CONFIGFS_FS=y
  - CONFIG_SECURITYFS=y
  - CONFIG_KASAN=y

Preparing the target machine

```
{
        "target": "linux/arm64",
        "http": "127.0.0.1:56741",
        "workdir": "/path_to/workdir",
        "kernel_obj": "/path_to_kernel/linux_rockpi",
        "sshkey": "/path_to_ssh_key/stretch.id_rsa",
        "rpc": "127.0.0.1:0",
        "syzkaller": "/path_to_syzkaller",
        "procs": 5,
        "reproduce": false,
        "sandbox": "none",
        "type": "isolated",
        "vm": {
                "targets": ["192.168.2.101"],
                "pstore": false,
                "target_dir": "/tmp",
                "target_reboot": false
        },
        "enable_syscalls": [...],
        "disable_syscalls": [...]
}
```

Telling Syzkaller about our driver

Enhancing syscall definitions

```
resource fd_hantro_dec[int32]: -1
resource fd_hantro_enc[int32]: -1

fd_hantro [
    enc fd_hantro_enc
    dec fd_hantro_dec
]

openat$hantro_enc(fd const[AT_FDCWD], file ptr[in, string["/dev/hantro_enc"]], flags const[O_RDWR],
                  mode const[0]) fd_hantro_enc
openat$hantro_dec(fd const[AT_FDCWD], file ptr[in, string["/dev/hantro_dec"]], flags const[O_RDWR],
                  mode const[0]) fd_hantro_dec
openat$hantro_media(fd const[AT_FDCWD], file ptr[in, string["/dev/hantro_media"]], flags
                    const[O_RDWR], mode const[0]) fd_media
```

Telling Syzkaller about our driver

Targeting the appropriate device

Generic device file name parameter:

```
syz_open_dev$video(dev ptr[in, string["/dev/video#"]], id intptr, flags flags[open_flags]) fd_video
```

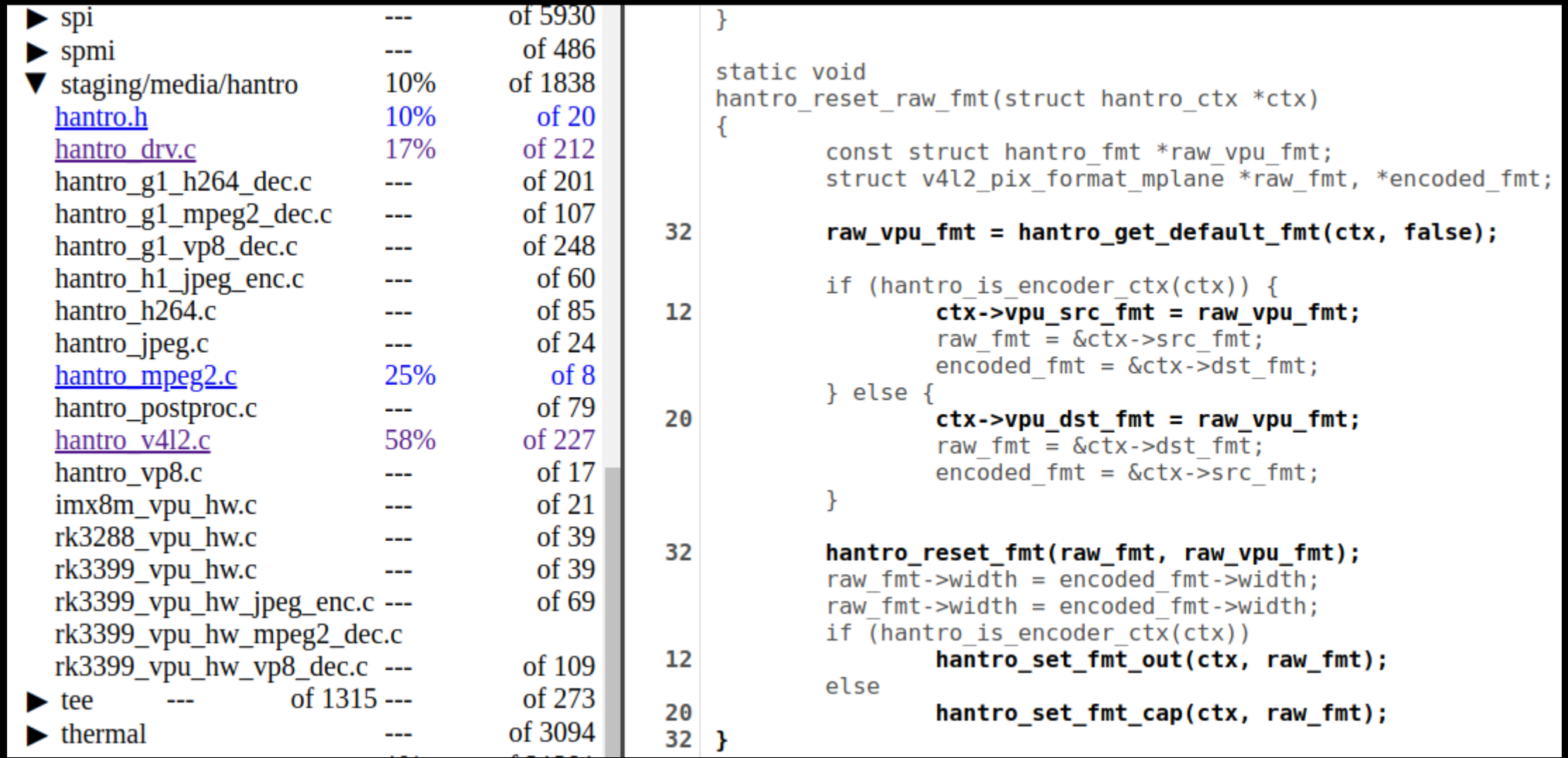
```
# Add udev rules for custom drivers.
# Create symlinks for the devices the hantro driver
echo 'ATTR{name}=="rockchip,rk3399-vpu-enc", SYMLINK+="hantro_enc"' \
     | sudo tee -a $DIR/etc/udev/rules.d/50-udev-default.rules
echo 'ATTR{name}=="rockchip,rk3399-vpu-dec", SYMLINK+="hantro_dec"' \
     | sudo tee -a $DIR/etc/udev/rules.d/50-udev-default.rules
echo 'ATTR{model}=="hantro-vpu", SYMLINK+="hantro_media"' \
     | sudo tee -a $DIR/etc/udev/rules.d/50-udev-default.rules
```

Telling Syzkaller about our driver

Add custom pseudo syscalls (discouraged)

- Generate a static chunk of code
- Generate controlled input programatically

Getting results



▶ spi	---	of 5930
▶ spmi	---	of 486
▼ staging/media/hantro	10%	of 1838
hantro.h	10%	of 20
hantro_drv.c	17%	of 212
hantro_g1_h264_dec.c	---	of 201
hantro_g1_mpeg2_dec.c	---	of 107
hantro_g1_vp8_dec.c	---	of 248
hantro_h1_jpeg_enc.c	---	of 60
hantro_h264.c	---	of 85
hantro_jpeg.c	---	of 24
hantro_mpeg2.c	25%	of 8
hantro_postproc.c	---	of 79
hantro_v4l2.c	58%	of 227
hantro_vp8.c	---	of 17
imx8m_vpu_hw.c	---	of 21
rk3288_vpu_hw.c	---	of 39
rk3399_vpu_hw.c	---	of 39
rk3399_vpu_hw_jpeg_enc.c	---	of 69
rk3399_vpu_hw_mpeg2_dec.c		
rk3399_vpu_hw_vp8_dec.c	---	of 109
▶ tee --- of 1315	---	of 273
▶ thermal	---	of 3094

```
    }

    static void
    hantro_reset_raw_fmt(struct hantro_ctx *ctx)
    {
            const struct hantro_fmt *raw_vpu_fmt;
            struct v4l2_pix_format_mplane *raw_fmt, *encoded_fmt;

32          raw_vpu_fmt = hantro_get_default_fmt(ctx, false);

            if (hantro_is_encoder_ctx(ctx)) {
12                  ctx->vpu_src_fmt = raw_vpu_fmt;
                    raw_fmt = &ctx->src_fmt;
                    encoded_fmt = &ctx->dst_fmt;
            } else {
20                  ctx->vpu_dst_fmt = raw_vpu_fmt;
                    raw_fmt = &ctx->dst_fmt;
                    encoded_fmt = &ctx->src_fmt;
            }

32          hantro_reset_fmt(raw_fmt, raw_vpu_fmt);
            raw_fmt->width = encoded_fmt->width;
            raw_fmt->width = encoded_fmt->width;
            if (hantro_is_encoder_ctx(ctx))
12                  hantro_set_fmt_out(ctx, raw_fmt);
            else
20                  hantro_set_fmt_cap(ctx, raw_fmt);
32  }
```

Getting results

	Corpus:
Coverage	**Program**
2634	openat$hantro_dec-syz_hantro_start-openat$hantro_dec-syz_hantro_start-openat$hantro_dec-syz_han
2585	openat$hantro_dec-syz_hantro_start-openat$hantro_dec-openat$hantro_media-syz_hantro_start
2579	openat$hantro_dec-openat$hantro_media-syz_hantro_start
2565	openat$hantro_media-ioctl-openat$hantro_dec-openat$hantro_media-syz_hantro_start
2555	openat$hantro_dec-openat$hantro_media-syz_hantro_start-syz_hantro_start
2542	openat$hantro_media-openat$hantro_dec-ioctl$hantro_VIDIOC_TRY_FMT-syz_hantro_start
2537	openat$hantro_dec-openat$hantro_media-syz_hantro_start
2307	openat$hantro_dec-syz_hantro_start
2294	openat$hantro_dec-syz_hantro_start-ioctl$hantro_VIDIOC_EXPBUF
2270	openat$hantro_dec-syz_hantro_start-openat$hantro_dec-syz_hantro_start
2236	openat$hantro_dec-syz_hantro_start-syz_hantro_start
1850	openat$hantro_enc-syz_hantro_start-ioctl$hantro_VIDIOC_EXPBUF
1671	openat$hantro_enc-ioctl$hantro_VIDIOC_S_EXT_CTRLS
1666	openat$hantro_dec-ioctl$hantro_VIDIOC_S_EXT_CTRLS
1619	openat$hantro_enc-ioctl$hantro_VIDIOC_S_EXT_CTRLS
1598	openat$hantro_enc-ioctl$hantro_VIDIOC_S_EXT_CTRLS
1580	openat$hantro_enc-ioctl$hantro_VIDIOC_S_EXT_CTRLS
1256	openat$hantro_dec-ioctl$hantro_VIDIOC_CREATE_BUFS
1220	openat$hantro_enc-ioctl$hantro_VIDIOC_ENUM_FRAMESIZES
1219	openat$hantro_enc-ioctl$hantro_VIDIOC_S_EXT_CTRLS
1212	openat$hantro_dec-ioctl$hantro_VIDIOC_TRY_FMT

Getting results

```
ioctl$vim2m_VIDIOC_QUERYCAP(0xffffffffffffffff, 0x80685600, &(0x7f0000000100)={""/16, ""/32, ""/32, 0x0, @vim2m})
r0 = openat$vim2m(0xffffffffffffff9c, &(0x7f0000000200)='/dev/vim2m\x00', 0x2, 0x0)
ioctl$vim2m_VIDIOC_QUERYCAP(r0, 0x80685600, &(0x7f0000000240))
r1 = openat$vim2m(0xffffffffffffff9c, &(0x7f0000000040)='/dev/vim2m\x00', 0x2, 0x0)
ioctl$vim2m_VIDIOC_QUERYCAP(r1, 0x80685600, &(0x7f0000000180)={""/16, ""/32, ""/32, 0x0, @vim2m})
clock_gettime(0x4, &(0x7f0000000000))
ioctl$vim2m_VIDIOC_QUERYCAP(r1, 0x80685600, &(0x7f0000000080))
```

```
// autogenerated by syzkaller (https://github.com/google/syzkaller)

#define _GNU_SOURCE

#include <endian.h>
...

uint64_t r[2] = {0xffffffffffffffff, 0xffffffffffffffff};

int main(void)
{
        syscall(__NR_mmap, 0x1ffff000ul, 0x1000ul, 0ul, 0x32ul, -1, 0ul);
    syscall(__NR_mmap, 0x20000000ul, 0x1000000ul, 7ul, 0x32ul, -1, 0ul);
    syscall(__NR_mmap, 0x21000000ul, 0x1000ul, 0ul, 0x32ul, -1, 0ul);
                intptr_t res = 0;
    syscall(__NR_ioctl, -1, 0x80685600, 0x20000100ul);
memcpy((void*)0x20000200, "/dev/vim2m\000", 11);
    res = syscall(__NR_openat, 0xffffffffffffff9cul, 0x20000200ul, 2ul, 0ul);
    if (res != -1)
        r[0] = res;
    syscall(__NR_ioctl, r[0], 0x80685600, 0x20000240ul);
...
```

Syzbot

https://syzkaller.appspot.com

open (888):

Title	Repro	Bisected	Count	Last	Reported
BUG: unable to handle kernel paging request in wait_consider_task (2)			19	9d06h	3h07m
possible deadlock in f_getown			1	3h34m	3h17m
WARNING: proc registration bug in afs_manage_cell	C	cause	1	4d03h	3h17m
WARNING: filesystem loop0 was created with 512 inodes, the real m…	C	cause	2	1d00h	3h17m
BUG: unable to handle kernel paging request in diFree	C	cause	3	22h05m	3h17m
KASAN: use-after-free Read in __proc_create			1	1d20h	3h28m
general protection fault in mac80211_hwsim_tx_frame_no_nl			1	2d17h	3h28m
KASAN: null-ptr-deref Read in tcf_idrinfo_destroy			4	1h24m	3h28m
BUG: unable to handle kernel paging request in tcf_action_dump_terse			1	9h05m	3h29m
BUG: unable to handle kernel NULL pointer dereference in __lookup…			1	1d16h	3h29m
general protection fault in tcf_generic_walker			1	2d18h	3h29m
general protection fault in io_uring_flush	syz		1	1d12h	3h37m
KASAN: use-after-free Read in tcf_action_init	C	cause	1	2d20h	3h37m
INFO: task hung in tcf_action_init_1	C	cause	2	19h29m	3h37m
INFO: trying to register non-static key in exfat_cache_inval_inode	C	cause	2	2d07h	3h37m
kernel BUG at fs/erofs/inode.c:LINE!	C	cause	4	23h56m	3h37m
possible deadlock in do_fcntl			1	4h31m	3h47m
possible deadlock in io_write			1	1d03h	3h47m
INFO: task hung in nbd_ioctl (3)	syz		1	6h29m	3h47m
general protection fault in io_poll_double_wake (2)	C	cause	2	23h46m	3h47m
INFO: task hung in tcindex_partial_destroy_work	C	cause	8	10m	3h47m
WARNING: CPU: 1	C	cause	1	5d02h	1d02h
BUG: unable to handle kernel paging request in dqput	C	cause	2	5d14h	1d14h

Thank you

```
Message {
  config {
    priority: "high"
    body: "Collabora is hiring"  // Many open positions
    recipient: "you"             // Please join us
    calltoaction: "http://col.la/join"
  }
}
```