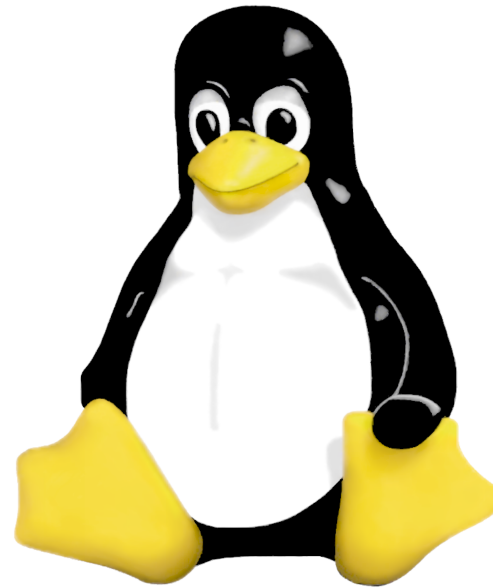


# FOSS Outreach Program for Women (OPW)



Linux Kernel  
Internship Report

# What is OPW?

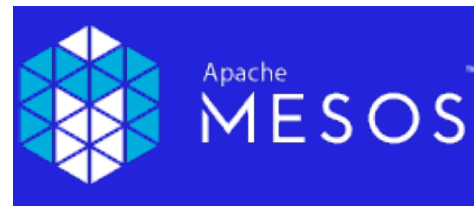
- Organized by the GNOME Foundation
- Goal: Get more women into open source
- Internship:
  - 3 months
  - \$5,500 stipend
  - Paired with mentor
- Program runs twice a year
  - May - Aug
  - Dec - March



# Who can apply as interns?

- Women, genderqueer, genderfluid, and genderfree people
- Don't have to be a student
- Must be able to work full-time
- Can work remotely
- Ascend Project alumni
  - Branching out into including other minorities

# Which projects are involved?

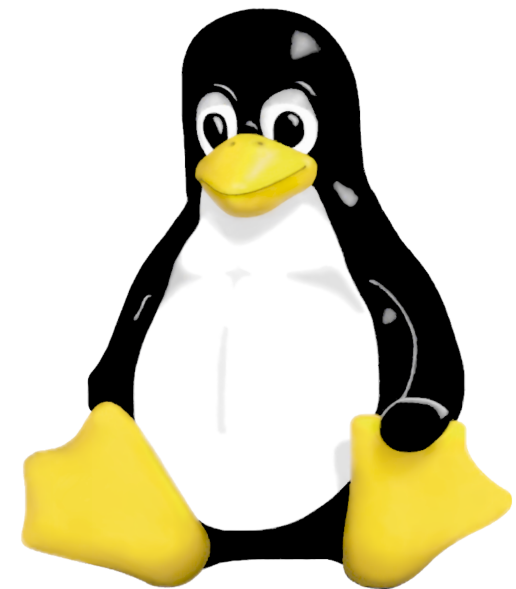


WIKIPEDIA  
The Free Encyclopedia

programming republic of



Perl®



# How are kernel internships paid?

Round 8 Promoter (3 interns)



Round 8 Includers (1 intern)



# Round 9

## applications open!

- This round:
  - applications opened September 22
  - applications due Oct 31
  - internships run Dec 9 - March 9

**[https://wiki.gnome.org/](https://wiki.gnome.org/OutreachProgramForWomen)**  
**OutreachProgramForWomen**



# OPW Kernel Internships Results

- 3 OPW rounds
- 16 alumni
- Top kernel contributors in 3.11, 3.12, 3.13, 3.14
- 1,157 patches from OPW interns & alumni
- diff stat: +44,148, -205,758



CC BY flickr Philo Nordlund

# Creating a Pipeline

- 4 alumni hired as Linux kernel developers
  - Intel, Linaro, Oracle, OnApp
- 1 alumni hired by Citrix
  - working on a proprietary project



CC BY-SA flickr flazingo



# How can I help out with OPW?

- Companies and individuals can:
  - Donate funds towards OPW interns
  - Talk to OPW coordinators  
<opw-admins@gnome.org>
- Linux kernel developers can:
  - Review application patches
  - Help out on IRC
  - Volunteer as mentors
  - Talk to Sarah Sharp <sarah.a.sharp@intel.com>
- Career counseling, job placement



# **OPW Linux Kernel Alumni Presentations**

# Staging driver cleanups

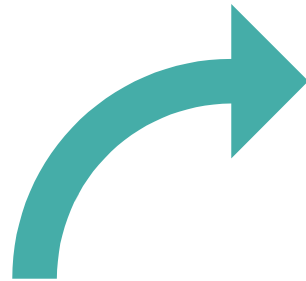
Kristina Martšenko

Mentor: Greg Kroah-Hartman

```
$ ls drivers/staging/
```

Kconfig	gdm72xx	panel	staging.c
Makefile	goldfish	phison	ste_rmi4
android	gs_fpgaboot	quickstart	tidsbridge
bcm	iio	rtl8188eu	unisys
ced1401	imx-drm	rtl8192e	usbip
comedi	keucr	rtl8192u	vme
cptm1217	line6	rtl8712	vt6655
crystalhd	lustre	rtl8723au	vt6656
cxtle1	media	rtl8821ae	winbond
dgap	mt29f_spinand	rts5139	wlags49_h2
dgnc	netlogic	rts5208	wlags49_h25
dgrp	nokia_h4p	sbe-2t3e3	wlan-ng
et131x	nvec	sep	xgifb
frontier	octeon	serqt_usb2	xillybus
ft1000	octeon-usb	silicom	
fwserial	olpc_dcon	slicoss	
gdm724x	ozwpan	speakup	

drivers/staging/media/



drivers/media/

Hi \_\_\_\_\_,

I'm helping Greg do a bit of cleanup in the staging tree. I noticed that nobody seems to have worked towards moving \_\_\_\_\_ out of staging in over a year. Are there any plans to clean it up and move it out soon? Because otherwise we're going to have to delete the driver, as we don't want staging to become a permanent place for unfinished code.

Thanks,  
Kristina

"I no longer have access to the hardware so I'm not planning on doing any more work on it."

"I recommend that kernel drop it. It's been an nice run but it's time to move on."

“I will continue working on this driver. Please leave this driver in staging if possible.”

“I think I can do this kind of cleaning stuff in the next few weeks, and I have needed hardware to test.”



“We are working on next set of patches for this driver, it is in final stage, will be posting it soon.”

?

```
$ ls drivers/staging/
```

Kconfig	gdm72xx	panel	staging.c
Makefile	goldfish	phison	ste_rmi4
android	gs_fpgaboot	quickstart	tidsbridge
bcm	iio	rtl8188eu	unisys
ced1401	imx-drm	rtl8192e	usbip
comedi	keucr	rtl8192u	vme
cptm1217	line6	rtl8712	vt6655
crystalhd	lustre	rtl8723au	vt6656
cxtle1	media	rtl8821ae	winbond
dgap	mt29f_spinand	rts5139	wlags49_h2
dgnc	netlogic	rts5208	wlags49_h25
dgrp	nokia_h4p	sbe-2t3e3	wlan-ng
et131x	nvec	sep	xgifb
frontier	octeon	serqt_usb2	xillybus
ft1000	octeon-usb	silicom	
fwserial	olpc_dcon	slicoss	
gdm724x	ozwpan	speakup	

```
$ ls drivers/staging/
```

Kconfig	gdm72xx	panel	staging.c
Makefile	goldfish	phison	ste_rmi4
android	gs_fpgaboot	quickstart	tidsbridge
bcm	iio	rtl8188eu	unisys
ced1401	imx-drm	rtl8192e	usbip
comedi	keucr	rtl8192u	vme
cptm1217	line6	rtl8712	vt6655
crystalhd	lustre	rtl8723au	vt6656
cxtle1	media	rtl8821ae	winbond
dgap	mt29f_spinand	rts5139	wlags49_h2
dgnc	netlogic	rts5208	wlags49_h25
dgrp	nokia_h4p	sbe-2t3e3	wlan-ng
et131x	nvec	sep	xgifb
frontier	octeon	serqt_usb2	xillybus
ft1000	octeon-usb	silicom	
fwserial	olpc_dcon	slicoss	
gdm724x	ozwpan	speakup	

# IIO

voltage

current

temperature

acceleration

humidity

light

proximity

...

# MXS LRADC

**ADC**

**touchscreen**

general-purpose ADC

die temperature

battery voltage

...

# MXS LRADC

**ADC**

**touchscreen**

general-purpose ADC

die temperature

battery voltage

...

`drivers/input/touchscreen/`

# MXS LRADC

**ADC**

**touchscreen**

general-purpose ADC

die temperature

battery voltage

drivers/iio/

drivers/input/touchscreen/



drivers/mfd/



drivers/iio/

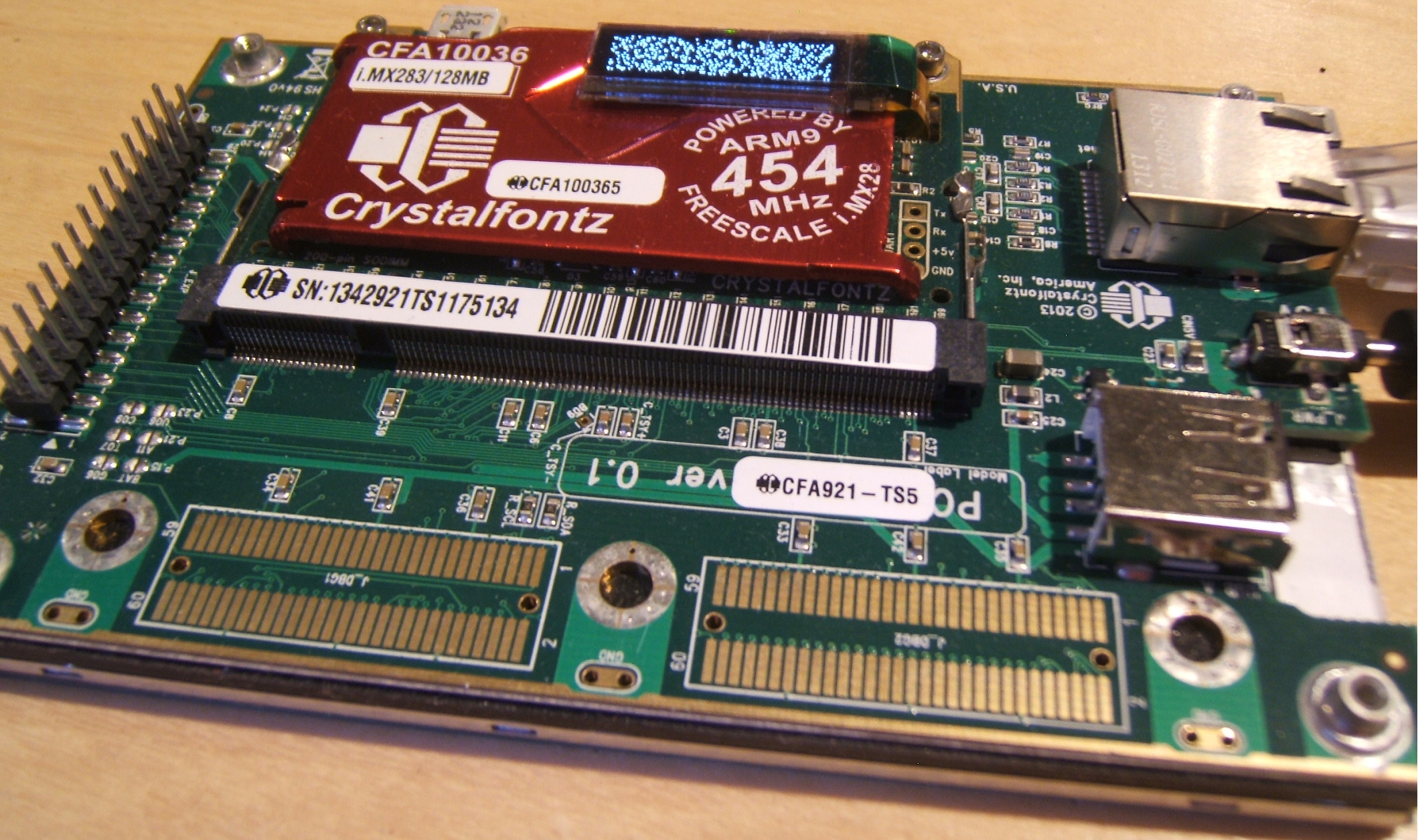


drivers/input/touchscreen/

A word cloud featuring various Linux kernel subsystems and drivers. The words are arranged in a non-uniform, overlapping manner. The colors are a mix of light green and teal. The words include: interrupt handling, touchscreens, device trees, MFD, input, hwmon, IIO, ADCs, locking, sysfs, udev, debugfs, clocks, tracing, and embedded devices. The word 'IIO' is notably large and centered.

interrupt handling touchscreens  
device trees MFD input  
hwmon IIO ADCs locking  
sysfs udev  
debugfs clocks  
tracing embedded devices





CFA10036  
i.MX283/128MB

  
CFA100365  
**CrystalFontz**

POWERED BY  
ARM9  
**454**  
MHZ  
i.MX28  
FREESCALE

 SN:1342921TS1175134



Ver 0.1

CFA921-TS5

Model Label

© 2013  
CrystalFontz  
America, Inc.



Desktop



06:46

### Applications



**Fotowall**

Photo collection creativ...



**Image Viewer**

Browse and rotate ima...



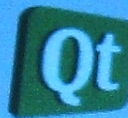
**Music Player**

Play your favourite songs



**QT Browser**

Qt Browser



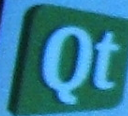
**QT Demo Launc...**

Qt Demo Launcher



**QT Media Player**

Qt Media Player



**RamboRemote L...**

RamboRemote Launch...



**Slideshow**

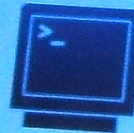


**smarthome**



**Video**

Video Player



**X11VNC Server**

Share this desktop by ...

ZK-1005



drivers/mfd/



drivers/iio/



drivers/input/touchscreen/

Future plans

# Hacking the Linux kernel with Coccinelle



Himangi Saraogi, Linux kernel intern,  
FOSS Outreach Program for Women Round 8

LinuxCon Europe 2014

# Introduction

- Computer Science student from IIT-Hyderabad, India currently in my pre-final year.
- Summer Intern for the May-August 2014 round.
- Worked on cleaning up code in the Linux kernel using Coccinelle with Julia Lawall.



# How I got Involved

- Linux had been my primary operating system at the university.
- While learning operating systems, I found some exercises to be practiced on the kernel.
- Motivation - Contributing with a great community to impact thousands of users and developers worldwide.

# My project

Develop/harden coccinelle semantic patches to integrate into the kernel.

## What Coccinelle can do?

- Static analysis to find patterns in C source code.
- Automatic transformation to fix bugs.
- Generate different information of bugs based on patch notation.



# What I did?

- Identify bugs that are prevalent across the kernel. (coccinellery)
- Send patches solving the bug to discuss whether it is an issue of concern.
- Develop coccinelle scripts to fix those bugs.
- Analyze results of the scripts.
- Send patches for the scripts to be accepted into the kernel.

# Other tasks

- Convert various device drivers to use the "devm" functions. Remove resource leaks.
- Improve the documentation of the devm functions.
- New devm functions - devm\_kasprintf and devm\_kvasprintf, which were necessary to introduce devm functions in several drivers.

# Current and future contributions

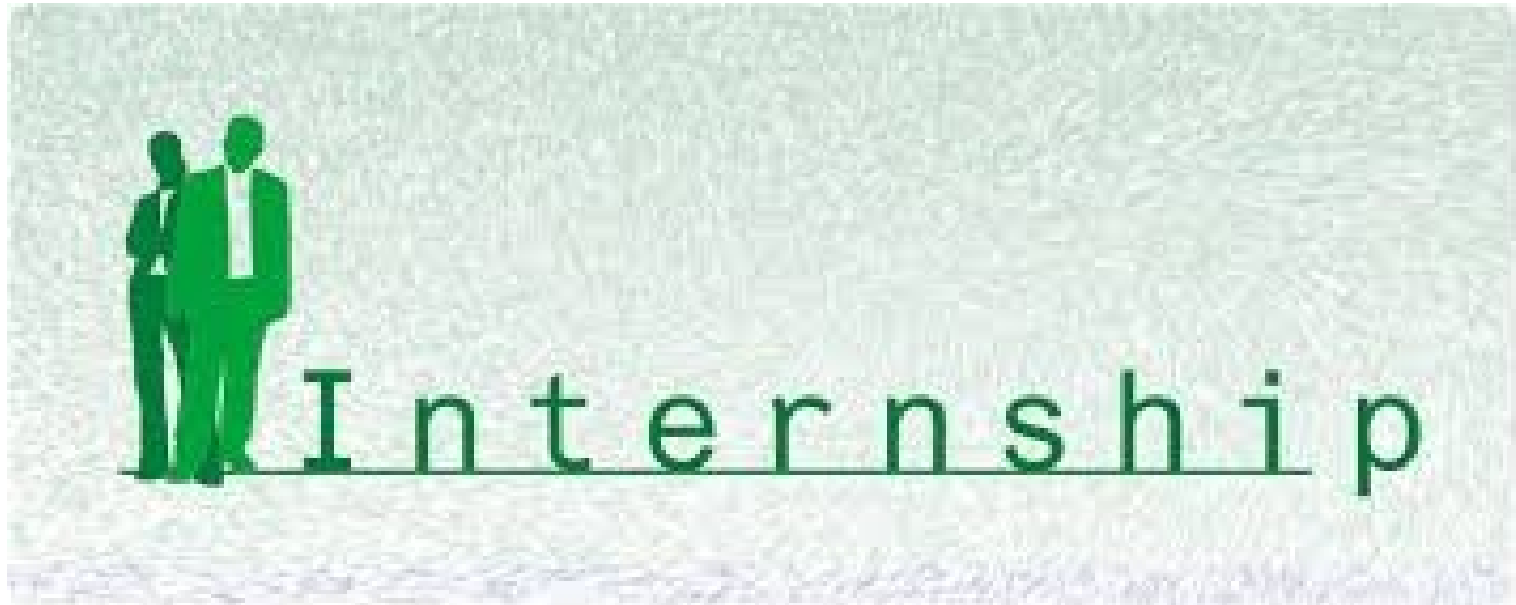
- A lot more can be done as bugs are omnipresent :)
- Continue to contribute to the kernel using Coccinelle and otherwise.
- Help newbies to enter the amazing open source domain.
- Spread awareness and help developers get started with Coccinelle
  - Linux Plumbers on 16<sup>th</sup> October 9:30 am.
  - Linux Conference Auckland during 12-16 January'15.

# What I have learnt

- How to learn
- The development process of Linux
- Time Management
- How to Understand source code
- Patch management process

Thus, it was an **amazing** experience to learn and code with awesome people.

# I'm looking for an internship



## Software Developer

# Acknowledgements

- **Julia Lawall** for being the most awesome mentor I have ever had.
- **Sarah Sharp** for always being supportive and coordinating the Linux kernel projects.
- **Greg K-H** for being the patient maintainer and offering feedback on patches.
- **Marina Z.** and **Karen S.** for organizing the OPW program.
- **The Linux Foundation** for giving us this opportunity at the LinuxCon.



# Thank you for listening

Himangi Saraogi

[himangi774@gmail.com](mailto:himangi774@gmail.com)

Website: <http://web.iit.ac.in/~himangi.saraogi>

<http://himangi99.wordpress.com/>



# Outreach Program for Women

Andreea-Cristina Bernat

Linux Kernel Intern

## Project

Automatically Locate Read-Copy-Update  
Abuses

## Mentor

Paul E. McKenney

# Project's purpose

- Automatically detect read-copy-update (RCU) usage issues using Coccinelle
- Provide fixes for the problems found

# Reasons

- Why use a function which does more than you want?

## Examples

- `rcu_access_pointer()` instead of `rcu_dereference()`
- `RCU_INIT_POINTER()` instead of `rcu_assign_pointer()`  
(in some cases)

# Reasons

The same problem in multiple files?

- Create a Coccinelle semantic patch and solve the problem automatically

# Reasons

- Use Read-Copy-Update mechanism correctly

# Examples of what I worked on

- Making use of an RCU-protected pointer after passing it to `call_rcu()`

```
/* BUG */
```

```
call_rcu(&p->head, func);
```

```
p->a = 1;
```

```
/* OK */
```

```
call_rcu(&p->head, func);
```

```
p = kmalloc(...);
```

```
p->a = 1;
```



# Examples of what I worked on

- Uses of `rcu_dereference()` whose return value is never dereferenced should be converted to `rcu_access_pointer()`

One simple case

```
if(rcu_dereference(p) == NULL) { ... }
```

# Before OPW

- Intention to contribute to Linux kernel
- Did not know how to create and send a patch
- Did not know what RCU or Coccinelle were

# Things learned during OPW

(application process + the internship)

- How to build the kernel
  - How to use git
- How to create and send patches
- How to interact with the open source community

# Things learned during OPW

(application process + the internship)

- How to read more easily code written by other people
  - A good understanding of how RCU works
- Achieved an intermediate experience with Coccinelle

# OPW experience

- Multiple patches accepted into the mainline Linux kernel
- Worked with many interesting people

# OPW experience

- The greatest way to start contributing to Linux kernel

# Current plans

- Finish my last year of university
- Continue to contribute to Linux kernel
- Open to any other opportunities

Thank you!





# TREE-WIDE WARNING FIXES AND STATIC ANALYSIS ENHANCEMENTS

**Rashika Kheria**

**Alumni - FOSS Outreach Program for Women**

**Mentor: Josh Triplett**  
**LinuxCon Europe 2014**

# ABOUT ME

- Final year student in Bachelor's program (CS Major) in **International Institute of Information Technology - Hyderabad**, India
- OPW Developer Intern for the Linux Kernel from December 2013 - March 2014
- **Interests:** Operating Systems (Linux), Computer Networks, Open Source Development
- Love Sleeping in spare time



# PROJECT OVERVIEW

## ◦ Goal

Eliminate all instances of `-Wmissing-prototypes` compiler warning tree-wide from the Linux Kernel

## ◦ Process

Analyzed the kernel build logs to find instances and causes of the warnings and submitted patches to eliminate them.



## WHY THIS WARNING?

- Fewer warnings and hence cleaner kernel build
- Removes unused code
- Shrinks the Kernel size
- Allows Optimization/Inlining
- Detect when prototype and definition don't match



# WHAT I DID

- Switched on occurrences of the warning by adding `-Wmissing-prototypes` to `KBUILD_CFLAGS` in the top-level Makefile
- For each warning, searched for the occurrences of the symbols causing the warning in the kernel code using git tools like “git grep”
- Applied the appropriate fix to remove the warning
- Submitted the changes in the form of a patch to appropriate recipients as indicated by `get_maintainer.pl` script.



# CHALLENGES

- Choosing the appropriate header file to put the prototype in.
- Proper handling of nested includes.
- Symbols hidden in macros and preprocessors

Example: In file drivers/scsi/be2iscsi/be\_main.c

```
#define beiscsi_disp_param(_name) \
ssize_t \
beiscsi_##_name##_disp(struct device *dev, \
    struct device_attribute *attrib, char *buf) \
{ \
    struct Scsi_Host *shost = class_to_shost(dev); \
    struct beiscsi_hba *phba = iscsi_host_priv(shost); \
    uint32_t param_val = 0; \
    param_val = phba->attr_##_name; \
    return snprintf(buf, PAGE_SIZE, "%d\n", \
        phba->attr_##_name); \
}
```



- Difficulty in adding header files in a single file due to incompatible definitions

Example: In file drivers/gpu/drm/radeon/radeon\_drv.h

```
# define RADEON_CSQ_PRIBM_INDBM      (4 << 28)
# define RADEON_CSQ_PRIPIO_INDPIO    (15 << 28)

#define R300_CP_RESYNC_ADDR          0x0778
#define R300_CP_RESYNC_DATA          0x077c

#define RADEON_AIC_CNTL              0x01d0
# define RADEON_PCIGART_TRANSLATE_EN (1 << 0)
```

Example: In file drivers/gpu/drm/radeon/radeon\_reg.h

```
#          define RADEON_CSQ_PRIBM_INDBM      (4      << 28)
#          define RADEON_CSQ_PRIPIO_INDPIO    (15      << 28)

#define R300_CP_RESYNC_ADDR          0x778
#define R300_CP_RESYNC_DATA          0x77c

#define RADEON_CP_CSQ_STAT           0x07f8
#          define RADEON_CSQ_RPTR_PRIMARY_MASK (0xff << 0)
```



## ○ Typo error in guard symbols

Example: In file drivers/isdn/hardware/eicon/xdi\_msg.h

```
#ifndef __DIVA_XDI_UM_CFG_MESSSGE_H__  
#define __DIVA_XDI_UM_CFG_MESSAGE_H__
```





# OPW LEARNINGS AND ACHIEVEMENTS

- First contact with FOSS community
- Learn about git tools – creating, sending and updating patches
- Reading, debugging and testing open source code and understanding build logs.
- Tweaking files across different subdirectories of the kernel handling challenges mentioned above.
- Was ranked 4<sup>th</sup> in the list of **most active developers** for the Linux Kernel Release 3.14  
<https://lwn.net/Articles/590354/>



# CURRENT AND FUTURE PLANS

- Graduating in April 2015
- Continue to contribute to Open Source projects
- On the lookout for job opportunities for a career in the same domain
- Particularly interested in the field of Operating Systems and Networks



# ACKNOWLEDGEMENTS

- Linux Foundation and GNOME's Outreach Program
- Linux Kernel mentors especially Josh Triplett, Sarah Sharp, Greg K-H
- Past OPW interns

Thank you for listening!  
Please feel free to contact me at  
[rashika.kheria@gmail.com](mailto:rashika.kheria@gmail.com) for any queries.



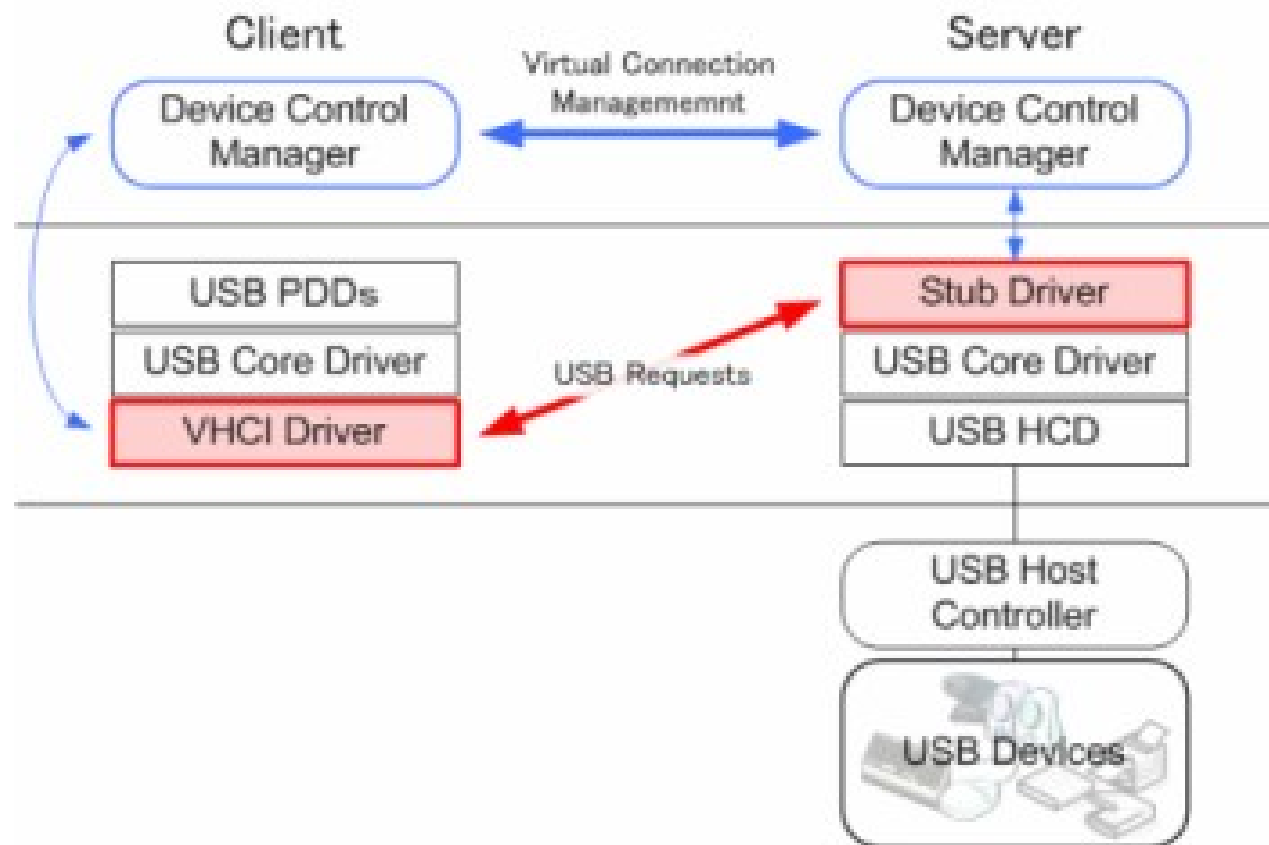
# USB IP Staging Driver

Valentina Manea  
Mentor: Andy Grover

# USB/IP (1)

- set of drivers in the kernel plus userspace utility
- allows sharing USB devices over network
- device is shared at driver level

# USB/IP (2)



# Improvements

- convert to interface driver
- migrate to libudev
- various fixes in the userspace utility
- move USB/IP out of staging

# Working within OPW

- mentorship
- initiative is valued
- feedback
- get introduced to the community



# Current Status

- maintainer for USB/IP
- Masters student
- looking for an internship :)

# OPW experience

---

From kernelnewbie to not so  
kernelnewbie

Teodora Băluță (teo)  
Mentor: PJ Waskiewicz

# Applying to OPW

LINUX: A TRUE STORY:

WEEK ONE

HEY, IT'S YOUR COUSIN  
I GOT A NEW COMPUTER  
BUT DON'T WANT WINDOWS.  
CAN YOU HELP ME  
INSTALL "LINUX"?

SURE.



WEEK TWO

IT SAYS MY XORG  
IS BROKEN. WHAT'S  
AN "XORG"? WHERE  
CAN I LOOK THAT UP



HMM,  
LEMME  
SHOW YOU  
MAN PAGES.

WEEK SIX

DUE TO AUTO-  
CONFIG ISSUES, I'M  
LEAVING UBUNTU  
FOR DEBIAN.



UH  
OR  
GENTOO.  
UHOH.

WEEK TWELVE

YOU HAVEN'T ANSWERED  
YOUR PHONE IN DAYS.

CAN'T SLEEP.  
MUST COMPILE  
KERNEL.



I'M  
TOO  
LATE.



PARENTS: TALK TO YOUR  
KIDS ABOUT LINUX..  
BEFORE SOMEBODY ELSE DOES.

# Project

---

- generate and decode QR codes for Oops messages and crashes

# Project Specifics

---

- research existing QR code libraries
- integrate/write own library in the kernel
- write compressed QR code as output in the framebuffer

# QR codes & Linux

---

- idea first came in 2012
  - <http://lwn.net/Articles/503677/>
- open source licenses
- research QR codes and how they work
- libqrencode
  - <http://fukuchi.org/works/qrencode/>
- kernel Oops mechanism

# How does it work?

---

- add hook in `printk()`'s calls to write Oops messages in a buffer
- when Oops is over, compress using `zlib` from `lib/`
- write to framebuffer using the minimal interface in `fb.h`

# RFC aftermath

---

- discussion on [lkml.org](http://lkml.org)
- posted project on [github.com/teobaluta/qr-linux-kernel](https://github.com/teobaluta/qr-linux-kernel)
- community involvement
  - contributions by Levente Kurusa
- mobile app on Android as a POC



# Back to the future

---

- started working at Intel OTC Bucharest
  - sensors drivers using Industrial I/O interface
  - Android HAL layer
- Master's degree
- keep on hacking the Linux kernel

# TODOs

---

- merge the kernel to the latest stable
- cleanup & test
- better alternative to zlib
- reporting Oops crashes needs a better workflow

# The Outreach Program For Women

Ana  
Rey

## The nftables project

Mentor: Pablo Neira

# Who am I?

- Ana Rey (@anarb). I live in Sevilla (Spain)
- I have participated in the social aspect of the Free/Open Source software.
- I have extensive experience in troubleshooting, system administration, event organization, team coordination, and leadership.
- Thanks to the OPW, I could focus on the development side.

# The application period of the OPW

- Around 97 commits in the staging kernel  
(Cleanup the rtl8192u, rtl8187se and rts5139 drivers)
- The Linux kernel coding style.
- How to make the perfect commit.
- How to compile the Linux Kernel.
- How to accept a recommendation and how to apply it.

# The nftables project

- **nftables** is the project that aims to replace the existing {ip,ip6,arp,eb}tables framework.
- To use nftables:
  - nftables's Linux kernel tree
  - libmnl: the minimalistic Netlink library
  - libnftnl: the user-space library for low-level interaction with nftables Netlink's API over libmnl.
  - nftables userspace utility

# My accomplishments in nftables (1/4)

- Refactoring and adding some improvements to the libnftnl (around 80 patches)

# My accomplishments in nftables (2/4)

- Adding support to ***pkttype***, ***cpu***, ***groupdev*** and ***phydev*** iptables-extension in nftables tool. (and into libnftnl library and into kernel)
  - **pkttype**:  
nft add rule ip filter input meta pkttype PACKET\_TYPE (PACKET\_TYPE: unicast, broadcast and multicast.)
  - **cpu**  
nft add rule ip test input meta cpu 1 counter  
nft add rule ip test input meta cpu 1-3 counter  
nft add rule ip test input meta cpu { 1, 3} counter
  - **devgroup**:  
nft add rule ip test input meta iifgroup 2 counter  
nft add rule ip test output meta oifgroup 2 counter
  - **phydev**  
(This feature is not accepted for the maintainer yet.)

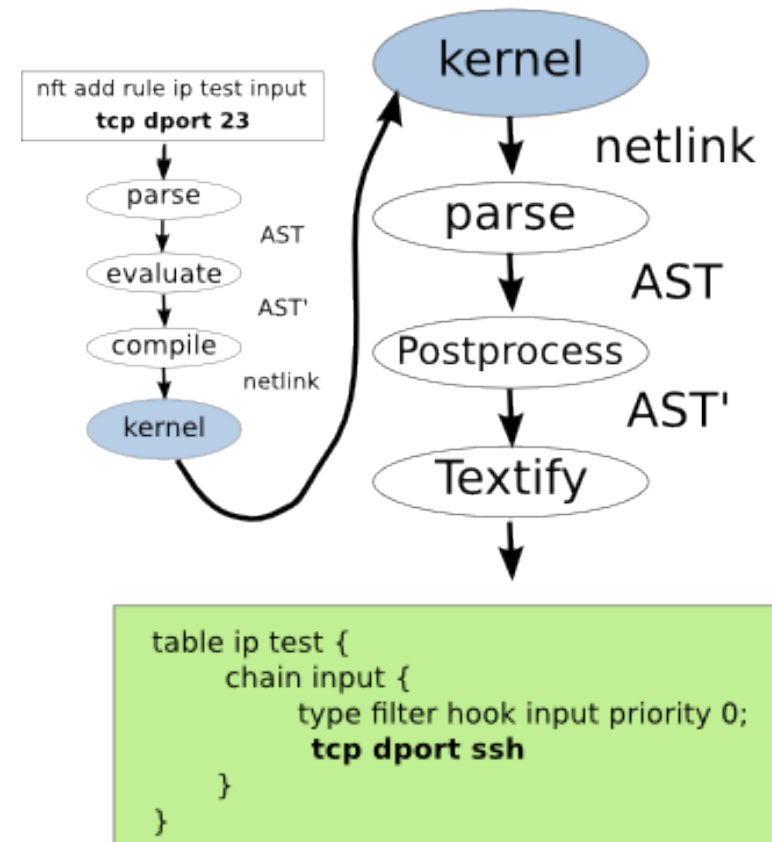


# My accomplishments in nftables (3/4)

- The automated regression testing of nftables.
  - This is a python script and some test files.
  - This script checks that the rule input and output of nft match.

Check inputs and outputs

`./nft-test.py ipv4/tcp.t`



# My accomplishments in nftables (3/4)

```
cat inet/tcp.t
```

```
1 *ip;test-ip4
2 *ip6;test-ip6
3 *inet;test-inet
4 :input;type filter hook input priority 0
5
6 tcp dport 22;ok
7 tcp dport != 233;ok
8 tcp dport 33-45;ok;tcp dport >= 33 tcp dport <= 45
```

any/meta.t: **WARNING**: line: 12: 'nft add rule -nnn ip test-ip4 input meta length 33-45': 'meta length 33-45' mismatches 'meta length >= 33 meta length <= 754974720'

any/ct.t: **ERROR**: line 10: nft add rule -nnn ip test-ip4 output ct state new,established, related, untracked: This rule should not have failed.

any/log.t: **OK**

any/limit.t: **OK**

**31 test files, 16 files passed, 797 unit tests, 2 error, 107 warning**



# My accomplishments in nftables (4/4)

- Reporting a lot of bug that I found when I was developing this automated regression testing.
- I was invited to the last Netfilter Workshop in France.

# Future plans for The Netfilter Project

- Testing the matching of packets:
  - Generating an artificial and specific packet
  - Checking that the nftable filters them correctly.

# Thanks

- Marina Zhurakhinskaya and Karen Sandler of The Gnome Foundation.
- Sarah Sharp.
- Pablo Neira for being my mentor during this internship.
- OPW Sponsors: Intel, Linux Foundation, Codethink



October 13 - 15, 2014. Düsseldorf, Germany