Introduction To Linux Kernel Hacking

Kernel Hackathon, Bangalore

Vaishali Thakkar

(vaishali.thakkar@oracle.com, @kernel_girl)

Who Am I?

- Linux Kernel developer at Oracle
- Working in kernel security engineering group and memory management
- Interested in many different subsystems of the Linux Kernel
- Associated with the open source internship programs

Agenda

- Prerequisites
- Process of the Kernel Development
- Introduction of tools to find the bugs
- Conclusion

Prerequisites

- Linux-next source code:

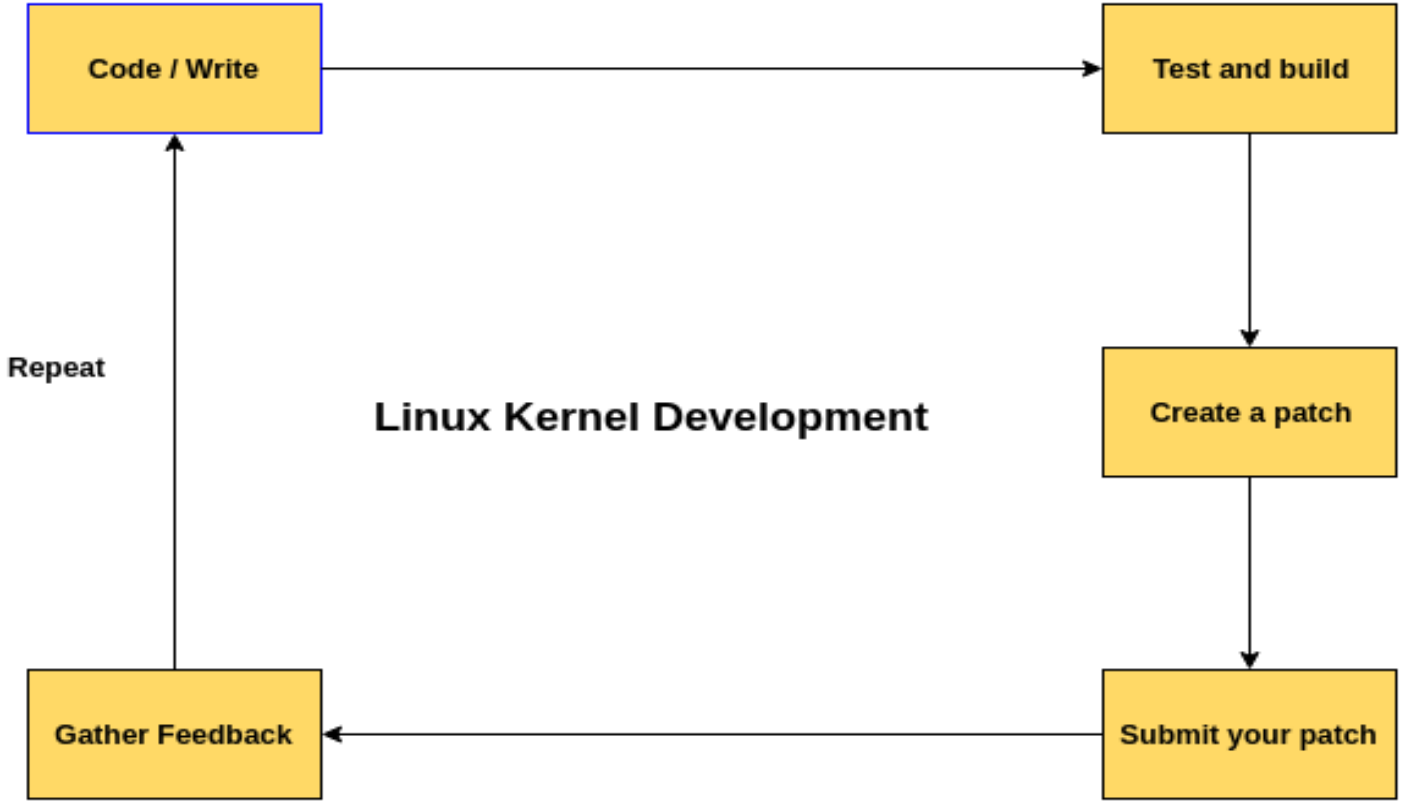
  git clone https://git.kernel.org/pub/scm/linux/kernel/git/torvalds/linux.git

- Setting up mail client and text editor:

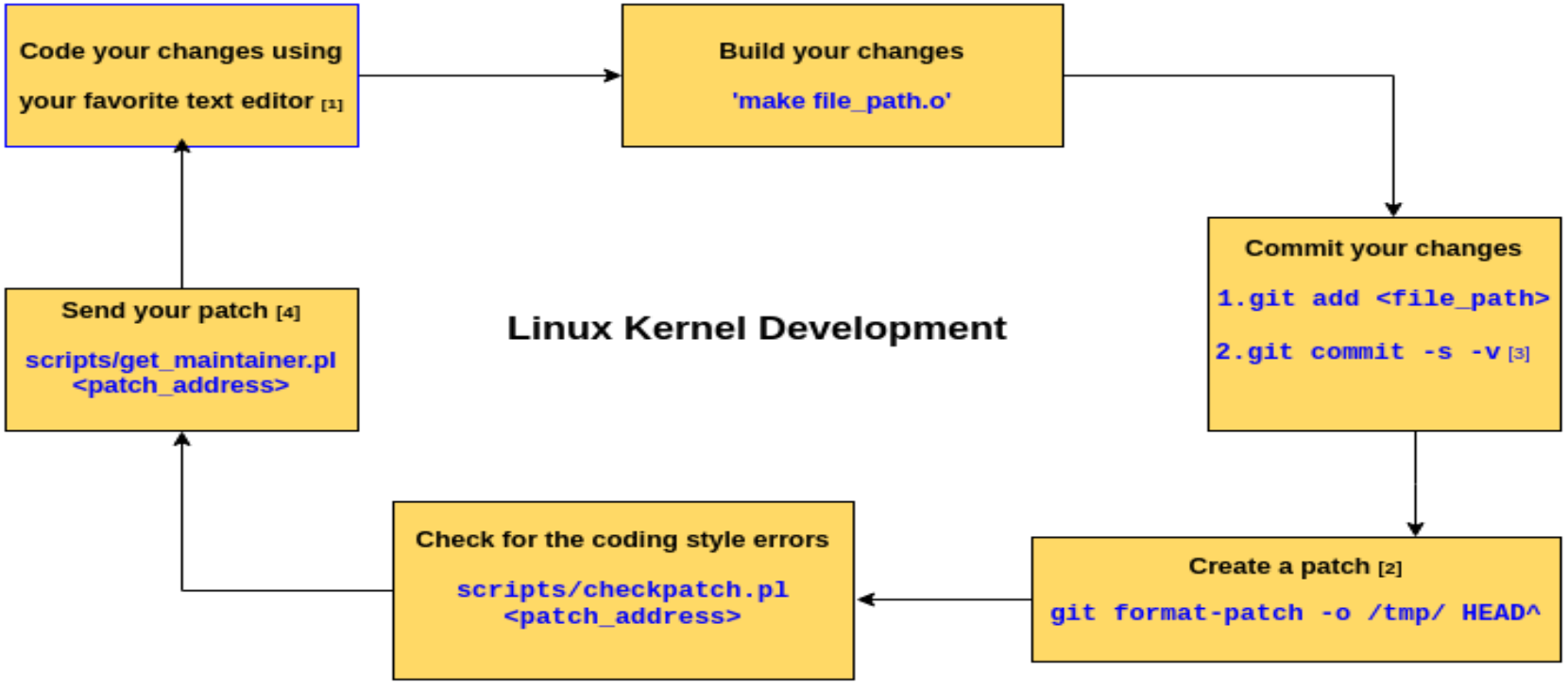
  https://kernelnewbies.org/FirstKernelPatch#head-17dd753ec497c8f7e2305ce78be8c6ca7cd1c92c

Process of the Kernel Development

- linux kernel hacking==creative cycle



Process of the Kernel Development



[1] Do the changes in your local branch: https://kernelnewbies.org/FirstKernelPatch#head-4fc0349738a61ed254bcbef7a980321c77495014

[2] You should see the command output with a filename in /tmp/

[3] Philosophy of Linux kernel patches: https://kernelnewbies.org/PatchPhilosophy

[4] Using mutt to send patches: https://kernelnewbies.org/FirstKernelPatch#head-dc6a8aa0be0d0e8ed9dc03726d0b5a1fb0f65e1f
Using git-send-email to send patches: https://burzalodowa.wordpress.com/2013/10/05/how-to-send-patches-with-git-send-email/

Contributing to the Linux Kernel

- Use bug finding tools [static checkers, dynamic checkers, fuzzers etc]
- Run kmemleak, kasan and other debugging features. Report bugs in the mailing lists.[1]
- Work on devm_functions and their missing uses
- More advanced project:

  https://kernsec.org/wiki/index.php/Kernel_Self_Protection_Project

[1]Watch out for the False Positives. Check mailing list archieves.

scripts/checkpatch.pl

- Written by Andy Whitcroft, Joe Perches
- Checks for basic coding style issues and sometimes for incorrect API usuage
- Preferrable to run it for any new patch submissions
- Things to take care of:
  - Avoid sending 80 characters line warning
  - If you are sending the patch for the simple warnings, send them for the files in staging/next

scripts/checkpatch.pl

- Example output: perl scripts/checkpatch.pl -f <path_to_{directory, file}>

```
CHECK: spaces preferred around that '+' (ctx:VxV)
#1564: FILE: drivers/staging/media/bcm2048/radio-bcm2048.c:1564:
+       BUG_ON((index+4) >= BCM2048_MAX_RDS_RT);
                         ^

drivers/staging/media/bcm2048/radio-bcm2048.c:1539: CHECK: Avoid
crashing the kernel - try using WARN_ON & recovery code rather
than BUG() or BUG_ON()

drivers/staging/media/bcm2048/radio-bcm2048.c:1997: ERROR: Macros
with complex values should be enclosed in parentheses

drivers/staging/media/bcm2048/radio-bcm2048.c:2025: WARNING:
Prefer 'unsigned int' to bare use of 'unsigned'

drivers/staging/media/bcm2048/radio-bcm2048.c:2543: WARNING:
struct v4l2_ioctl_ops should normally be const
```

Sparse

- Written by Linus Torvalds, later maintained by Josh Triplett, Chris Li
- Essentially, sparse is a library that, like a compiler front end, provides convenient access to the abstract syntax tree and typing information of a C program.
- Provides a set of annotations designed to convey semantic information about types.
  - For example, what address space pointers point to or what locks a function acquires or releases.

Sparse

- Installation:
  - From the package manager of your linux distro:

    e.g. sudo apt-get install sparse
  - Manual installation: https://kernelnewbies.org/Sparse
- Running Sparse: make C=2 <path_to_directory>
- Documentation:
  - Wikipedia: https://en.wikipedia.org/wiki/Sparse
  - Kernel Documentation: Documentation/sparse.txt

Sparse

- Example output:

```
drivers/staging/wlan-ng/p80211conv.c:132:25: warning: cast to
restricted __be16

drivers/staging/wlan-ng/p80211conv.c:154:38: warning: incorrect
type in assignment (different base types)
drivers/staging/wlan-ng/p80211conv.c:154:38: expected unsigned
short [unsigned] [usertype] type
drivers/staging/wlan-ng/p80211conv.c:154:38: got restricted
__be16 [usertype] <noident>

drivers/staging/wlan-ng/prism2fw.c:251:15: warning: memset with
byte count of 120000

drivers/staging/lustre/lnet/selftest/rpc.c:764:9: warning: context
imbalance in 'srpc_shutdown_service' - different lock contexts for
basic block
```

Smatch

- Written by Dan Carpenter
- Uses sparse as a C parser.
- Useful for finding many security[use-after-free, buffer overflow, off-by-one, double locks/unlocks, missing locks etc] related and other bugs.

Smatch

- Installation:
  - git clone git://repo.or.cz/smatch.git
  - cd smatch
  - make
- Running Smatch: <path_to_smatch>/smatch_scripts/kchecker --spammy ./
- Documentation:

  https://blogs.oracle.com/linuxkernel/entry/smatch_static_analysis_tool_overview

Smatch

- Example output:

```
drivers/staging/xgifb/vb_setmode.c:3581 XGI_SetGroup2() warn: mask
and shift to zero

drivers/staging/xgifb/vb_setmode.c:5334 XGI_EnableBridge() warn:
we tested 'pVBInfo->VBInfo & 256' before and it was 'true'

drivers/staging/vt6656/rf.c:876 vnt_rf_table_download() error:
memcpy() 'addr1' too small (3 vs 48)

drivers/staging/rts5208/ms.c:2736 ms_build_l2p_tbl() error:
buffer overflow 'ms_start_idx' 17 <= s32max

drivers/staging/rts5208/ms.c:2594 ms_build_l2p_tbl() error: we
previously assumed 'ms_card->segment' could be null(see line 2586)

drivers/staging/rts5208/sd.c:4115 ext_sd_send_cmd_get_rsp() warn:
masked condition '(*ptr + 3 & 30) != 3' is always true.
```

Coccinelle

- Written by Julia Lawall
- Program matching and transformation tool. It can warn you about bugs [report mode] or suggest a fix for the bugs [patch mode].
- Spatch: Coccinelle binary in /usr/bin or /usr/local/bin that invokes the Coccinelle program.
- Semantic Patch Language(SmPL): Not another scripting language, aware of the structure of the C language

Coccinelle

- Coccicheck:
  - One of the targets of the Linux kernel
  - Provides a series of semantic patches written in SmPL and make use of the Coccinelle engine to interpret and complete these tests.
  - Each script has confidence - High, Moderate, Low
  - Can be run with four modes:
    - Patch - lets you fix the issues found
    - report - lets you generate a report
    - context - highlights lines of interest[indicated by -] and their context in a diff-like style.
    - org - generates a report in the Org mode format of Emacs

Coccinelle

- Installation:
  - From the package manager of your linux distro:
    e.g. sudo apt-get install coccinelle
  - Manual installation: https://github.com/coccinelle/coccinelle
- Running coccicheck:
  - All scripts under scripts/coccinelle: make coccicheck MODE=patch
  - On specific directory:make coccicheck MODE=report M=drivers/net/
  - Running specific tests:make coccicheck
    COCCI=scripts/coccinelle/locks/double_lock.cocci MODE=report
- Documentation: Documentation/coccinelle

Coccinelle

- Example output:

```
./security/integrity/ima/ima_template.c:192:29-35: ERROR:
application of sizeof to pointer

./drivers/power/supply/ab8500_charger.c:3676:8-28: ERROR:
Threaded IRQ with no primary handler requested without
IRQF_ONESHOT

./sound/soc/samsung/i2s.c:1269:2-4: ERROR: test of a variable
/field address

./drivers/block/loop.c:736:8-15: ERROR: PTR_ERR applied after
initialization to constant on line 728

./fs/btrfs/send.c:6335:22-39: ERROR: sctx is NULL but
dereferenced.

./drivers/misc/lkdtm_heap.c:38:1-5: ERROR: reference preceded
by free on line 37
```

Conclusion

- **First kernel patch tutorial:**
  https://kernelnewbies.org/FirstKernelPatch
- **LWN Articles:** https://lwn.net/Kernel/
- **Linux Kernel Documentation:**
  Documentation directory in the Linux kernel source code

Stop talking and start hacking!

Thank You