42 Variability Bugs in Linux

A Qualitative Study

Iago Abal Rivas
IT University of Copenhagen

Joint work with Claus Brabrand and Andrzej Wąsowski

ASE 2014

September 18 in Västerås, Sweden

Agenda

- Introduction
- Method
  - Example
- Database (visit http://VBDb.itu.dk :-)
- Findings
- Conclusion

Variability in Software

```c
#include <stdlib.h>

void foo(int a) {
    printf("%d\n",2/a);
}

int main(void) {
    int x = 1;
    #ifdef CONFIG_INCR
    x = x + 1;
    #endif
    #ifdef CONFIG_DECR
    x = x - 1;
    #endif
    foo(x);
}
```

Variability in Software (cont.)

(a) $\neg \mathit{INCR} \wedge \neg \mathit{DECR}$

```
    int x = 1;







    foo(x);   // x = 1
```

(b) $\neg \mathit{INCR} \wedge \mathit{DECR}$

```
    int x = 1;




    x = x - 1;

    foo(x);   // x = 0
```

(c) $\mathit{INCR} \wedge \neg \mathit{DECR}$

```
    int x = 1;

    x = x + 1;




    foo(x);   // x = 2
```

(d) $\mathit{INCR} \wedge \mathit{DECR}$

```
    int x = 1;

    x = x + 1;


    x = x - 1;

    foo(x);   // x = 1
```

Variability Bug

```
#include <stdlib.h>

void foo(int a) {                                  →(5)
    printf("%d\n",2/a);      // ERROR                (6) ×
}

int main(void) {             // START              ⇒(1)
    int x = 1;                                       (2)
    #ifdef CONFIG_INCR        // DISABLED              |
    x = x + 1;                                         |
    #endif                                             |
    #ifdef CONFIG_DECR        // ENABLED               ↓
    x = x - 1;                                       (3)
    #endif                                             ↓
    foo(x);                                          (4)→
}
```

Motivation & Contribution

1. *Get evidence of variability bugs in real-world software.*
   - **Identification of 42 variability bugs in the Linux kernel.**
2. *Understand the requirements for effective bug finding.*
   - **An in-depth analysis of the collection of bugs.**
3. *Create a challenge for ourselves, and for the community.*
   - **A database containing the results of our analysis.**
   - **Self-contained simplified C99 versions of all bugs.**

*The **V**ariability **B**ugs **D**atabase*

http://VBDb.itu.dk

Motivation & Contribution

1. *Get evidence of variability bugs in real-world software.*
   - **Identification of 42 variability bugs in the Linux kernel.**
2. *Understand the requirements for effective bug finding.*
   - **An in-depth analysis of the collection of bugs.**
3. *Create a challenge for ourselves, and for the community.*
   - **A database containing the results of our analysis.**
   - **Self-contained simplified C99 versions of all bugs.**

*The **V**ariability **B**ugs **D**atabase*

http://VBDb.itu.dk

Motivation & Contribution

1. *Get evidence of variability bugs in real-world software.*
    - **Identification of 42 variability bugs in the Linux kernel.**
2. *Understand the requirements for effective bug finding.*
    - **An in-depth analysis of the collection of bugs.**
3. *Create a challenge for ourselves, and for the community.*
    - **A database containing the results of our analysis.**
    - **Self-contained simplified C99 versions of all bugs.**

*The **V**ariability **B**ugs **D**atabase*
http://VBDb.itu.dk

Motivation & Contribution

1. *Get evidence of variability bugs in real-world software.*
   - **Identification of 42 variability bugs in the Linux kernel.**
2. *Understand the requirements for effective bug finding.*
   - **An in-depth analysis of the collection of bugs.**
3. *Create a challenge for ourselves, and for the community.*
   - **A database containing the results of our analysis.**
   - **Self-contained simplified C99 versions of all bugs.**

The ***V****ariability* ***B****ugs* ***D****ata****b****ase*

http://VBDb.itu.dk

Agenda

Introduction

Method

- Example

Database (visit http://VBDb.itu.dk :-)

Findings

Conclusion

Filter commits that *may be* variability-related

```
commit 6252547b8a7acced581b649af4ebf6d65f63a34b
Author: Russell King <rmk+kernel@arm.linux.org.uk>
Date:   Tue Feb 7 09:47:21 2012 +0000

 ARM: omap: fix broken twl-core dependencies and ifdefs

 In commit aeb5032b3f, a dependency on IRQ_DOMAIN was added, which causes
 regressions on previously working setups: a previously working non-DT
 kernel configuration now loses its PMIC support.  The lack of PMIC
 support in turn causes the loss of other functionality the kernel had.

 ...

 The result is that OMAP3 oopses in the vp.c code (fixed by a previous
 commit) due to the lack of PMIC support.

 However, even with IRQ_DOMAIN enabled, the driver oopses:

 Unable to handle kernel NULL pointer dereference at ...
```

Filter commits that *may be* variability-related

```
diff --git a/drivers/mfd/Kconfig b/drivers/mfd/Kconfig
index cd13e9f..f147395 100644
--- a/drivers/mfd/Kconfig
+++ b/drivers/mfd/Kconfig
@@ -200,7 +200,7 @@ config MENELAUS

 config TWL4030_CORE
        bool "Texas Instruments TWL4030/TWL5030/TWL6030/TPS659x0 Support"
-       depends on I2C=y && GENERIC_HARDIRQS && IRQ_DOMAIN
+       depends on I2C=y && GENERIC_HARDIRQS
        help
          Say yes here if you have TWL4030 / TWL6030 family chip ...
          This core driver provides register access and IRQ handling
diff --git a/drivers/mfd/twl-core.c b/drivers/mfd/twl-core.c
index e04e04d..8ce3959 100644
--- a/drivers/mfd/twl-core.c
+++ b/drivers/mfd/twl-core.c
@@ -263,7 +263,9 @@ struct twl_client {

 static struct twl_client twl_modules[TWL_NUM_SLAVES];

+#ifdef CONFIG_IRQ_DOMAIN
 static struct irq_domain domain;
+#endif
```

Filter commits that *may be* fixing a bug

```
commit 6252547b8a7acced581b649af4ebf6d65f63a34b
Author: Russell King <rmk+kernel@arm.linux.org.uk>
Date:   Tue Feb 7 09:47:21 2012 +0000

 ARM: omap: fix broken twl-core dependencies and ifdefs

 In commit aeb5032b3f, a dependency on IRQ_DOMAIN was added, . . .

 . . . presumably to prevent a build error .

 The result is that OMAP3 oopses in the vp.c code ( fixed by . . .
 commit) due to the lack of PMIC support.

 However, even with IRQ_DOMAIN enabled, the driver oopses :

 Unable to handle kernel NULL pointer dereference at . . .
 pgd = c0004000
 [00000000] *pgd=00000000
 Internal error : Oops : 5 [#1] SMP
 . . .
```

ARM: omap: fix broken twl-core dependencies and ifdefs

```
static int twl_probe()
{
  int *ops = NULL;

  ops = &irq_domain_ops;

  irq_domain_add(ops);
}
```

```
void irq_domain_add(int *ops)
{
  int irq = *ops;
}
```

ARM: omap: fix broken twl-core dependencies and ifdefs

```
static int twl_probe()
{
  int *ops = NULL;
#ifdef CONFIG_OF_IRQ
  ops = &irq_domain_ops;
#endif
  irq_domain_add(ops);
}
```

```
void irq_domain_add(int *ops)
{
  int irq = *ops;
}
```

ARM: omap: fix broken twl-core dependencies and ifdefs

```
static int twl_probe()
{
  int *ops = NULL;
#ifdef CONFIG_OF_IRQ
  ops = &irq_domain_ops;
#endif
  irq_domain_add(ops);
}
```

```
void irq_domain_add(int *ops)
{
  int irq = *ops;
}
```

type: Null pointer dereference

descr: Null pointer on *!OF_IRQ* gets dereferenced if *IRQ_DOMAIN*.

In *TWL4030* driver, attempt to register an IRQ domain with a NULL ops structure: ops is de-referenced when registering an IRQ domain, but this field is only set when *OF_IRQ*.

config: *TWL4030_CORE && !OF_IRQ*

bugfix:

repo: git://git.kernel.org/pub/.../linux-stable.git

hash: 6252547b8a7acced581b649af4ebf6d65f63a34b

loc: model, mapping

trace:
```
!!trace |
. dyn-call drivers/mfd/twl-core.c:1190:twl_probe()
. 1235:  irq_domain_add(&domain);
..  call kernel/irq/irqdomain.c:20:irq_domain_add()
...  call include/linux/irqdomain.h:74:irq_domain_to_irq()
...  ERROR 77:  if (d->ops->to_irq)
```

links:
```
!!md |
* [I2C](http://cateee.net/lkddb/web-lkddb/I2C.html)
* [TWL4030](http://www.ti.com/general/docs/...)
* [IRQ domain](http://lxr.gwbnsh.net.cn/.../IRQ-domain.txt)
```

Agenda

Introduction

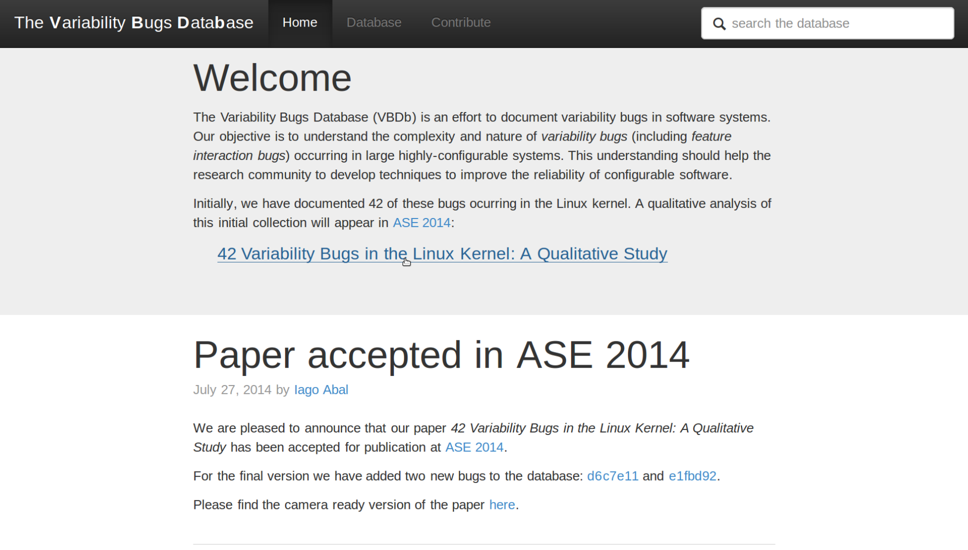
Method

- Example

Database (visit http://VBDb.itu.dk :-)

Findings

Conclusion



Welcome

The Variability Bugs Database (VBDb) is an effort to document variability bugs in software systems. Our objective is to understand the complexity and nature of *variability bugs* (including *feature interaction bugs*) occurring in large highly-configurable systems. This understanding should help the research community to develop techniques to improve the reliability of configurable software.

Initially, we have documented 42 of these bugs ocurring in the Linux kernel. A qualitative analysis of this initial collection will appear in ASE 2014:

42 Variability Bugs in the Linux Kernel: A Qualitative Study

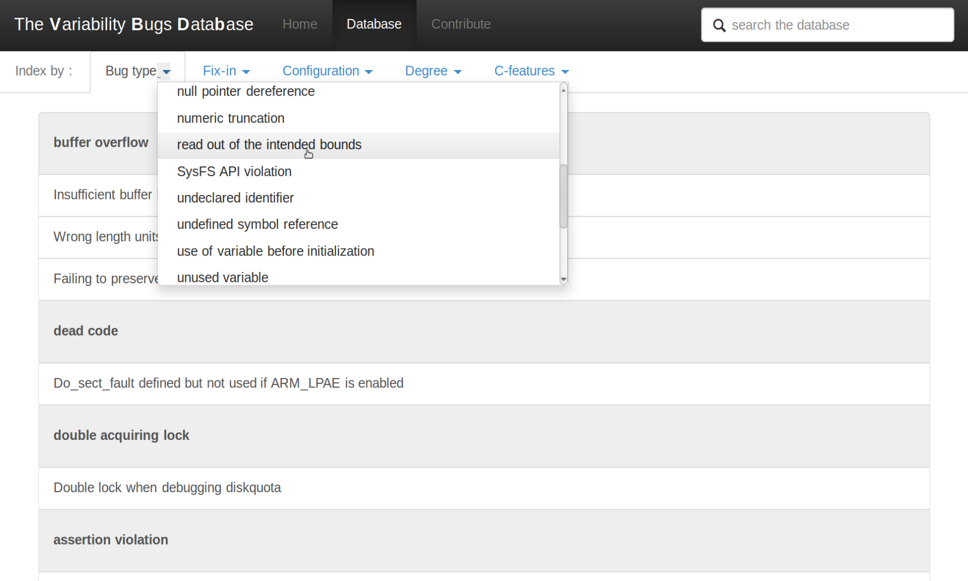
Paper accepted in ASE 2014

July 27, 2014 by Iago Abal

We are pleased to announce that our paper *42 Variability Bugs in the Linux Kernel: A Qualitative Study* has been accepted for publication at ASE 2014.

For the final version we have added two new bugs to the database: d6c7e11 and e1fbd92.

Please find the camera ready version of the paper here.



null pointer dereference
numeric truncation
read out of the intended bounds
SysFS API violation
undeclared identifier
undefined symbol reference
use of variable before initialization
unused variable

buffer overflow

Insufficient buffer

Wrong length units

Failing to preserve

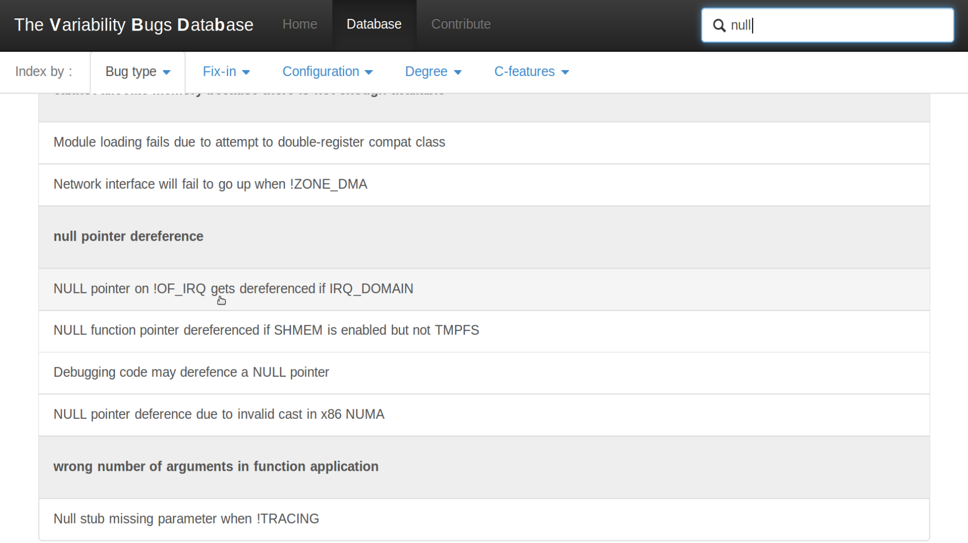
dead code

Do_sect_fault defined but not used if ARM_LPAE is enabled

double acquiring lock

Double lock when debugging diskquota

assertion violation



Module loading fails due to attempt to double-register compat class

Network interface will fail to go up when !ZONE_DMA

null pointer dereference

NULL pointer on !OF_IRQ gets dereferenced if IRQ_DOMAIN

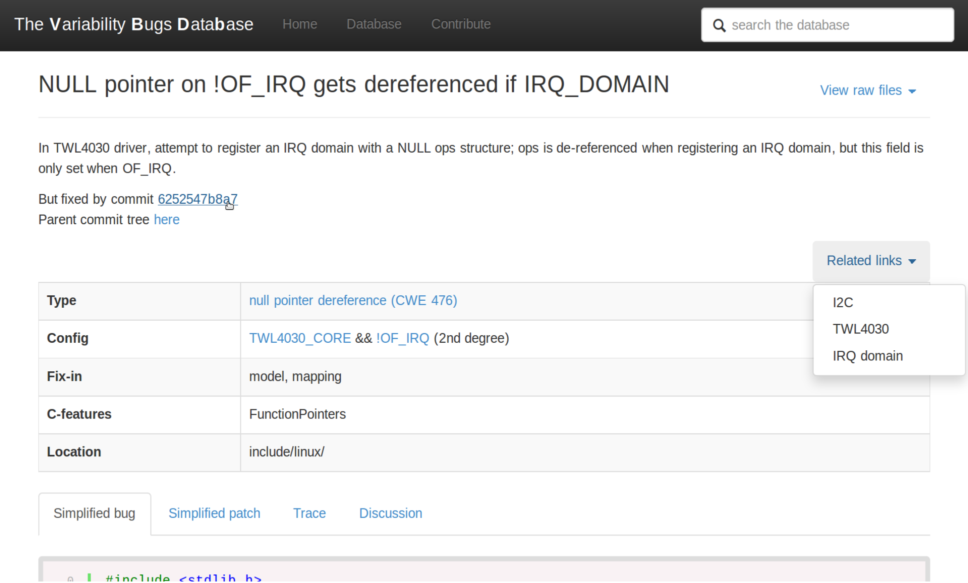
NULL function pointer dereferenced if SHMEM is enabled but not TMPFS

Debugging code may derefence a NULL pointer

NULL pointer deference due to invalid cast in x86 NUMA

wrong number of arguments in function application

Null stub missing parameter when !TRACING



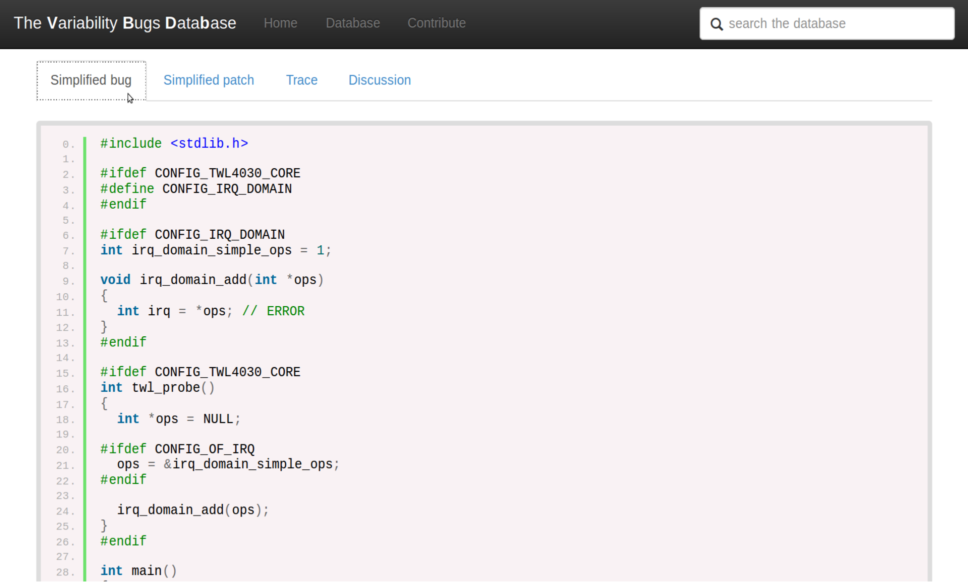
NULL pointer on !OF_IRQ gets dereferenced if IRQ_DOMAIN

In TWL4030 driver, attempt to register an IRQ domain with a NULL ops structure; ops is de-referenced when registering an IRQ domain, but this field is only set when OF_IRQ.

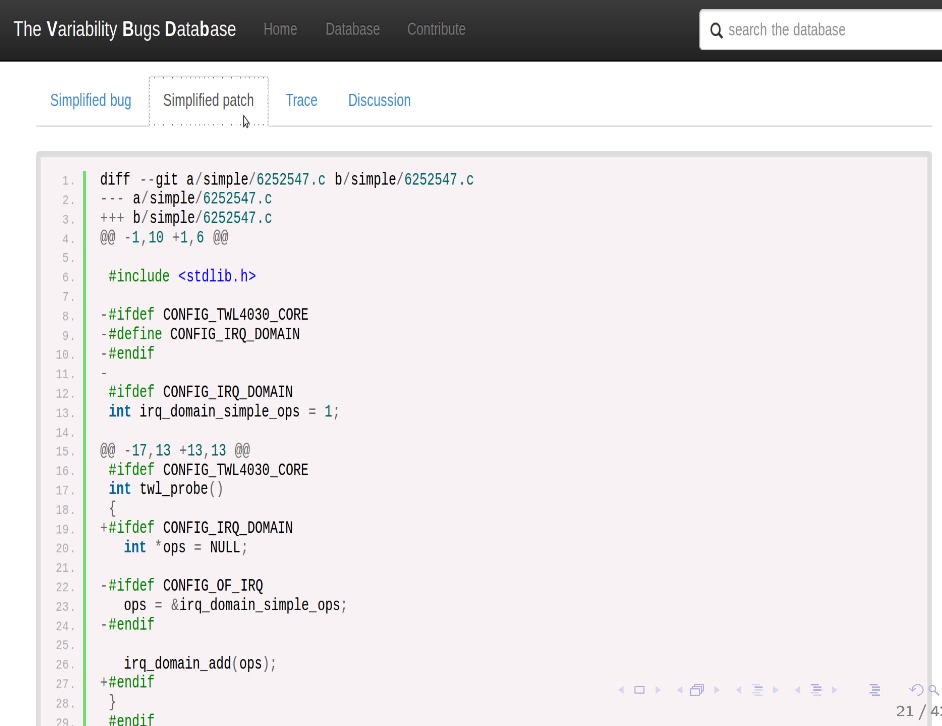
But fixed by commit 6252547b8a7
Parent commit tree here

Type	null pointer dereference (CWE 476)
Config	TWL4030_CORE && !OF_IRQ (2nd degree)
Fix-in	model, mapping
C-features	FunctionPointers
Location	include/linux/

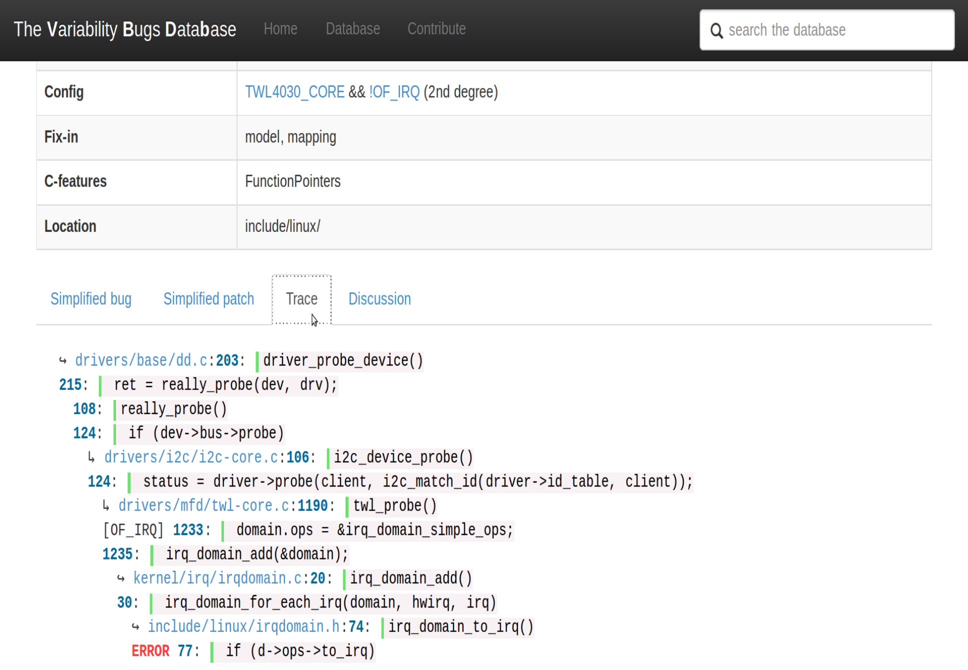
```
#include <stdlib.h>
```



```c
#include <stdlib.h>

#ifdef CONFIG_TWL4030_CORE
#define CONFIG_IRQ_DOMAIN
#endif

#ifdef CONFIG_IRQ_DOMAIN
int irq_domain_simple_ops = 1;

void irq_domain_add(int *ops)
{
  int irq = *ops; // ERROR
}
#endif

#ifdef CONFIG_TWL4030_CORE
int twl_probe()
{
  int *ops = NULL;

#ifdef CONFIG_OF_IRQ
  ops = &irq_domain_simple_ops;
#endif

  irq_domain_add(ops);
}
#endif

int main()
```

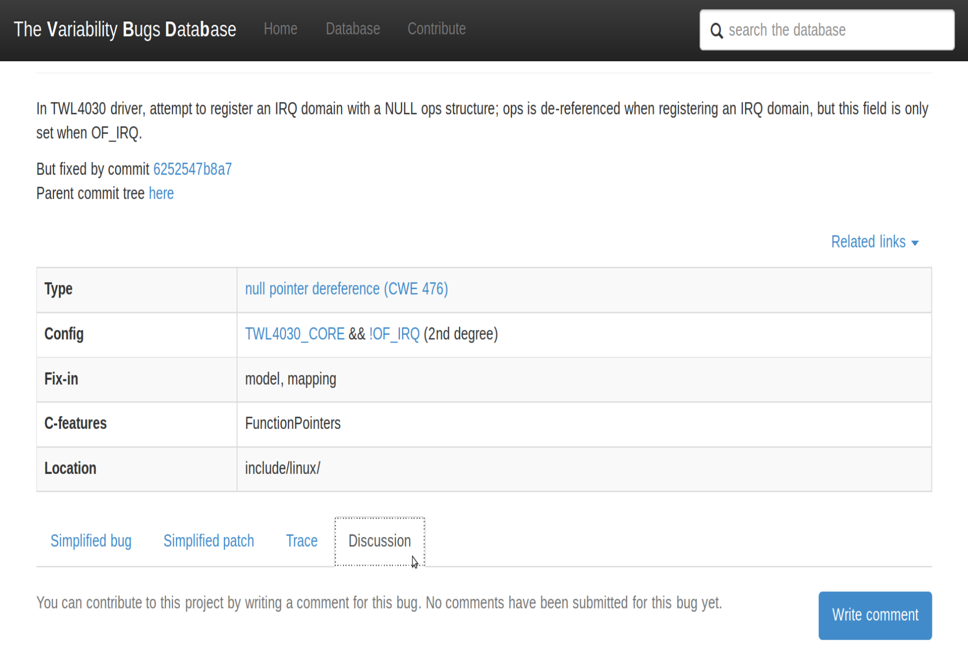


```
diff --git a/simple/6252547.c b/simple/6252547.c
--- a/simple/6252547.c
+++ b/simple/6252547.c
@@ -1,10 +1,6 @@
 
 #include <stdlib.h>
 
-#ifdef CONFIG_TWL4030_CORE
-#define CONFIG_IRQ_DOMAIN
-#endif
-
 #ifdef CONFIG_IRQ_DOMAIN
 int irq_domain_simple_ops = 1;
 
@@ -17,13 +13,13 @@
 #ifdef CONFIG_TWL4030_CORE
 int twl_probe()
 {
+#ifdef CONFIG_IRQ_DOMAIN
   int *ops = NULL;
 
-#ifdef CONFIG_OF_IRQ
   ops = &irq_domain_simple_ops;
-#endif
 
   irq_domain_add(ops);
+#endif
 }
 #endif
```



Config	TWL4030_CORE && !OF_IRQ (2nd degree)
Fix-in	model, mapping
C-features	FunctionPointers
Location	include/linux/

```
↳ drivers/base/dd.c:203: driver_probe_device()
215:   ret = really_probe(dev, drv);
  108: really_probe()
  124:   if (dev->bus->probe)
    ↳ drivers/i2c/i2c-core.c:106: i2c_device_probe()
    124:   status = driver->probe(client, i2c_match_id(driver->id_table, client));
      ↳ drivers/mfd/twl-core.c:1190: twl_probe()
      [OF_IRQ] 1233:   domain.ops = &irq_domain_simple_ops;
      1235:   irq_domain_add(&domain);
        ↳ kernel/irq/irqdomain.c:20: irq_domain_add()
        30:   irq_domain_for_each_irq(domain, hwirq, irq)
          ↳ include/linux/irqdomain.h:74: irq_domain_to_irq()
          ERROR 77:   if (d->ops->to_irq)
```



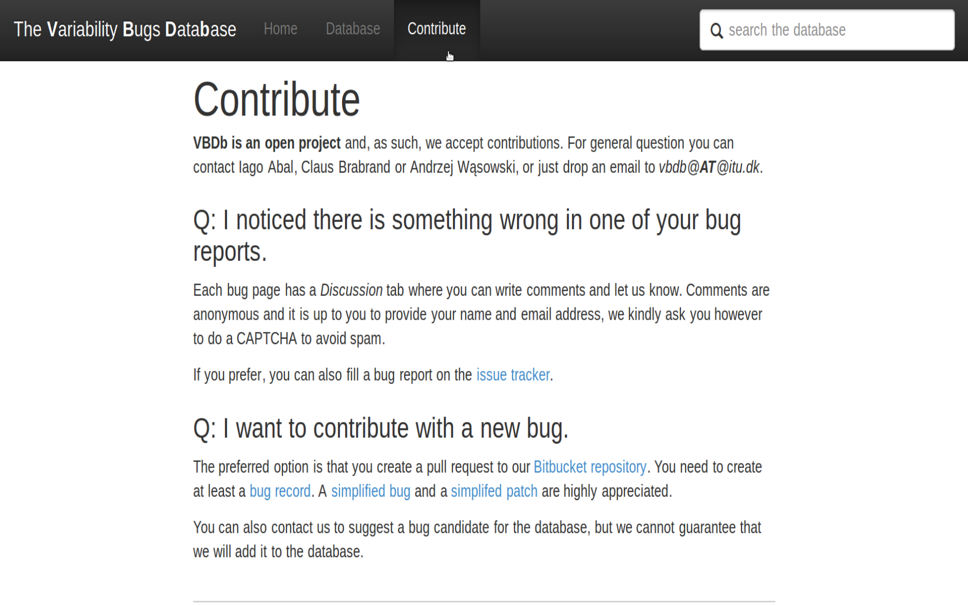
In TWL4030 driver, attempt to register an IRQ domain with a NULL ops structure; ops is de-referenced when registering an IRQ domain, but this field is only set when OF_IRQ.

But fixed by commit 6252547b8a7
Parent commit tree here

Related links

Type	null pointer dereference (CWE 476)
Config	TWL4030_CORE && !OF_IRQ (2nd degree)
Fix-in	model, mapping
C-features	FunctionPointers
Location	include/linux/

You can contribute to this project by writing a comment for this bug. No comments have been submitted for this bug yet.



Contribute

VBDb is an open project and, as such, we accept contributions. For general question you can contact Iago Abal, Claus Brabrand or Andrzej Wąsowski, or just drop an email to *vbdb@**AT**@itu.dk*.

Q: I noticed there is something wrong in one of your bug reports.

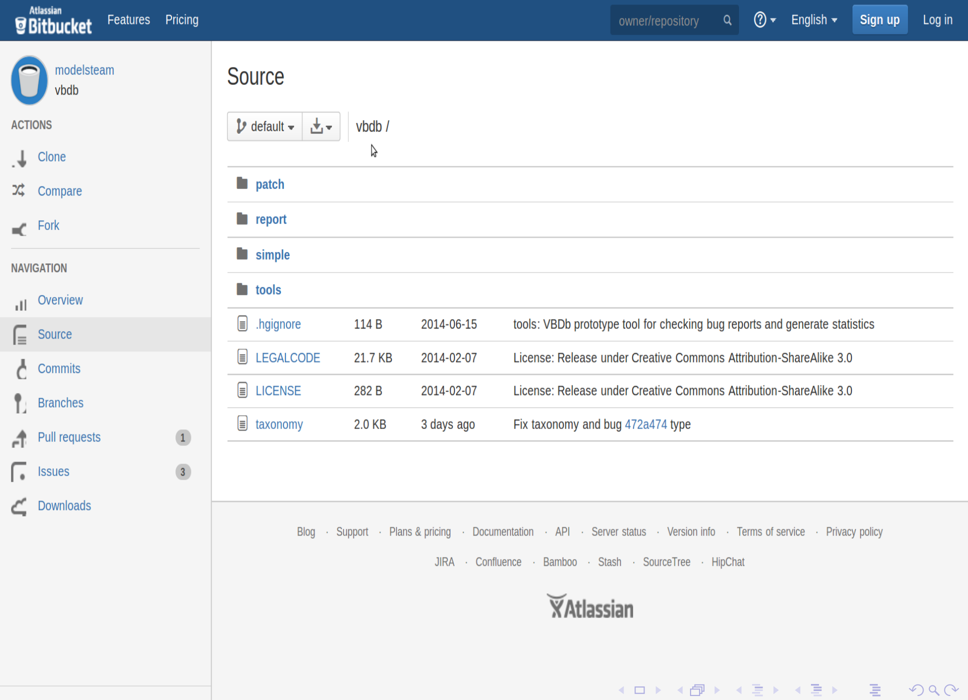
Each bug page has a *Discussion* tab where you can write comments and let us know. Comments are anonymous and it is up to you to provide your name and email address, we kindly ask you however to do a CAPTCHA to avoid spam.

If you prefer, you can also fill a bug report on the issue tracker.

Q: I want to contribute with a new bug.

The preferred option is that you create a pull request to our Bitbucket repository. You need to create at least a bug record. A simplified bug and a simplifed patch are highly appreciated.

You can also contact us to suggest a bug candidate for the database, but we cannot guarantee that we will add it to the database.



Source

vbdb /

patch			
report			
simple			
tools			
.hgignore	114 B	2014-06-15	tools: VBDb prototype tool for checking bug reports and generate statistics
LEGALCODE	21.7 KB	2014-02-07	License: Release under Creative Commons Attribution-ShareAlike 3.0
LICENSE	282 B	2014-02-07	License: Release under Creative Commons Attribution-ShareAlike 3.0
taxonomy	2.0 KB	3 days ago	Fix taxonomy and bug 472a474 type

Agenda

Introduction

Method

Example

Database (visit http://VBDb.itu.dk :-)

Findings

Conclusion

Please note that ...

WARNING!

The following slides include some quantitative data, but this is used to characterize our sample and not to draw conclusions about the entire population of bugs in the Linux kernel.

Observation (1)

Variability bugs are not limited to any particular type of bugs.

15	**memory errors**	CWE ID
4	null pointer dereference	476
3	buffer overflow	120
3	read out of bounds	125
2	insufficient memory	-
1	memory leak	401
1	use after free	416
1	write on read only	-
8	**compiler warnings**	CWE ID
5	uninitialized variable	457
2	incompatible types	843
1	unused function (dead code)	561
1	unused variable	563
1	void pointer dereference	-
7	**type errors**	CWE ID
5	undefined symbol	-
1	undeclared identifier	-
1	wrong number of args to function	-
7	**assertion violations**	CWE ID
5	fatal assertion violation	617
2	non-fatal assertion violation	617
2	**API violations**	CWE ID
1	Linux *sysfs* API violation	-
1	double lock	764
1	**arithmetic errors**	CWE ID
1	numeric truncation	197

Observation (2)

Variability bugs appear to not be restricted to specific features.

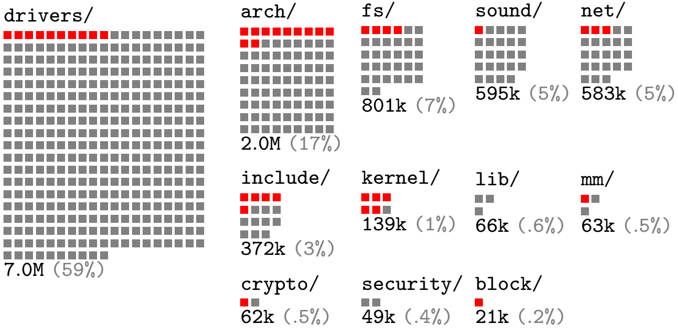
64BIT
ACPI_VIDEO
ACPI_WMI
AMIGA_Z2RAM
ANDROID
ARCH_OMAP2420
ARCH_OPAM3
ARM_LPAE
BACKLIGHT_CLASS_DEVICE
BCM47XX
BDI_SWITCH
BF60x
BLK_CGROUP
CRYPTO_BLKCIPHER
CRYPTO_TEST
DEVPTS_MULTIPLE_INSTANCES
DISCONTIGMEM
DRM_I915
EP93XX_ETH
EXTCON
FORCE_MAX_ZONEORDER=11
HIGHMEM
HOTPLUG
I2C
IOSCHED_CFQ
IPV6

IP_SCTP
JFFS2_FS_WBUF_VERIFY
KGDB
KPROBES
KTIME_SCALAR
LBDAF
LOCKDEP
MACH_OMAP_H4
MODULE_UNLOAD
NETPOLL
NUMA
OF
OF_IRQ
PARISC
PCI
PM
PPC64
PPC_256K_PAGES
PREEMPT
PROC_PAGE_MONITOR
PROVE_LOCKING
QUOTA_DEBUG
RCU_CPU_STALL_INFO
RCU_FAST_NO_HZ
REGULATOR_MAX8660
REISERFS_FS_SECURITY

S390
S390_PRNG
SCTP_DBG_MSG
SECURITY
SHMEM
SLAB
SLOB
SMP
SND_FSI_AK4642
SND_FSI_DA7210
SSB_DRIVER_EXTIF
STUB_POULSBO
SYSFS
TCP_MD5SIG
TMPFS
TRACE_IRQFLAGS
TRACING
TREE_RCU
TWL4030_CORE
UNIX98_PTYS
VLAN_8021Q
VORTEX
X86
X86_32
XMON
ZONE_DMA

Observation (3)

Variability bugs are not confined to any specific location.



Observation (4)

We have identified 29 bugs that involve non-locally defined features; i.e., features that are "remotely" defined in another subsystem than where the bug occurred.

drivers/extcon/extcon-class.c

```c
static int __init extcon_class_init(void)
{
    // Registers switch class using SysFS API.
}

static void __exit extcon_class_exit(void)
{
    // Does NOT deregister the switch class.
}
```

Observation (5)

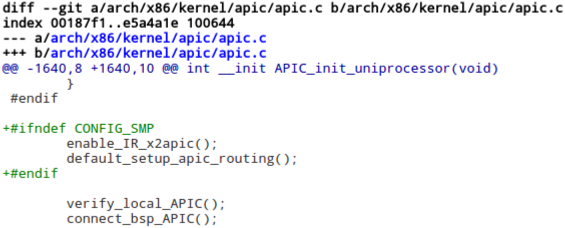
Variability can be implicit and even hidden in (alternative) configuration-dependent macro, function, or type definitions specified in (potentially different) header files.

include/linux/if_vlan.h

```
#if defined(CONFIG_VLAN_8021Q) || defined(CONFIG_VLAN_8021Q_MODULE)
  ...
#else
  ...
static inline bool vlan_hwaccel_do_receive(struct sk_buff **skb)
{
    BUG();
    return false;
}
  ...
#endif
```

Observation (6)

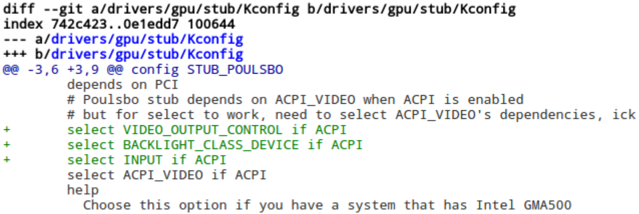
Variability bugs are fixed not only in the code; some are fixed in the mapping, some are fixed in the model, and some are fixed in a combination of these.



```
diff --git a/arch/x86/kernel/apic/apic.c b/arch/x86/kernel/apic/apic.c
index 00187f1..e5a4a1e 100644
--- a/arch/x86/kernel/apic/apic.c
+++ b/arch/x86/kernel/apic/apic.c
@@ -1640,8 +1640,10 @@ int __init APIC_init_uniprocessor(void)
        }
 #endif

+#ifndef CONFIG_SMP
        enable_IR_x2apic();
        default_setup_apic_routing();
+#endif

        verify_local_APIC();
        connect_bsp_APIC();
```

Observation (6)

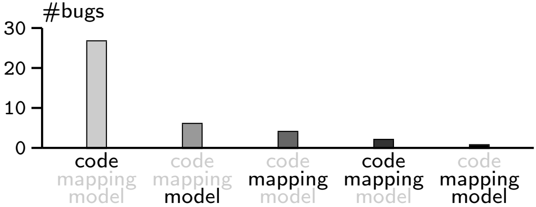
Variability bugs are fixed not only in the code; some are fixed in the mapping, some are fixed in the model, and some are fixed in a combination of these.



```
diff --git a/drivers/gpu/stub/Kconfig b/drivers/gpu/stub/Kconfig
index 742c423..0e1edd7 100644
--- a/drivers/gpu/stub/Kconfig
+++ b/drivers/gpu/stub/Kconfig
@@ -3,6 +3,9 @@ config STUB_POULSBO
        depends on PCI
        # Poulsbo stub depends on ACPI_VIDEO when ACPI is enabled
        # but for select to work, need to select ACPI_VIDEO's dependencies, ick
+       select VIDEO_OUTPUT_CONTROL if ACPI
+       select BACKLIGHT_CLASS_DEVICE if ACPI
+       select INPUT if ACPI
        select ACPI_VIDEO if ACPI
        help
          Choose this option if you have a system that has Intel GMA500
```

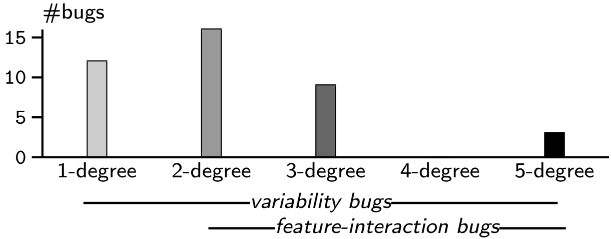
Observation (6)

Variability bugs are fixed not only in the code; some are fixed in the mapping, some are fixed in the model, and some are fixed in a combination of these.



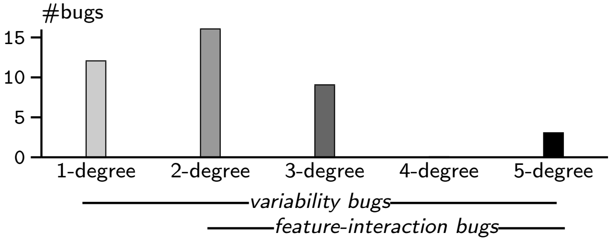
Observation (7)

We have identified as many as 28 feature-interaction bugs.



Observation (8)

We have identified 12 bugs involving three or more features.



Observation (9)

Presence conditions also involve disabled features.

19	***some-enabled***
6	a
7	$a \land b$
5	$a \land b \land c$
0	$a \land b \land c \land d$
1	$a \land b \land c \land d \land e$
19	***some-enabled-one-disabled***
4	$\neg a$
11	$a \land \neg b$
3	$a \land b \land \neg c$
0	$a \land b \land c \land \neg d$
1	$a \land b \land c \land d \land \neg e$
2	**other configurations**
1	$\neg a \land \neg b$
1	$a \land \neg b \land \neg c \land \neg d \land \neg e$

Observation (10)

There is an effective testing strategy for our sample of 42 bugs.

strategy	**cost**	**benefit**
all-yes	$O(1)$	48% (20/42)
all-yes-but-one	$O(\lvert F \rvert)$	95% (40/42)
exhaustive	$O(2^{\lvert F \rvert})$	100% (42/42)

Hypothesis

In practice, *all-yes-but-one* is a cost-effective testing strategy.

Observation (10)

There is an effective testing strategy for our sample of 42 bugs.

strategy	**cost**	**benefit**
all-yes	$O(1)$	48% (20/42)
all-yes-but-one	$O(\|F\|)$	95% (40/42)
exhaustive	$O(2^{\|F\|})$	100% (42/42)

Hypothesis

In practice, *all-yes-but-one* is a cost-effective testing strategy.

Agenda

Introduction

Method

Example

Database (visit http://VBDb.itu.dk :-)

Findings

Conclusion

Conclusion

- Variability bugs are diverse, and subtle.
- The implementation of features is intermixed.
- Features from different subsystems interact.
- Variability is implicit and even hidden in alternative definitions specified at spare locations.
- Stratification in three layers: C code, cpp, and Kconfig.
- *Disabling* features also triggers bugs.
- Conventional feature-insensitive tools are *not* enough.

Thank you

*The **V**ariability **B**ugs **D**ata**b**ase*
http://VBDb.itu.dk