

内核 Patch FAQ

NOTICE

1. 将文档在浏览器中打开，可以在左侧看到**目录**。
2. 在 `git format-patch` 生成patch文件以后，使用 `./scripts/checkpatch.pl ${PATCH_FILE}` 检查patch文件时，**warning是不可忽略的**。
3. 在使用学校的smtp服务器 `git send-email` 之前，需要首先设置自己的密码，没有默认密码。



4. `git send-email` 时出现的很多错误可以用**重发几次**的方法解决。（这里的“很多错误”或许需要大家来补充）
5. <https://docs.kernel.org/process/submitting-patches.html>
6. 可以在<https://groups.google.com/g/hust-os-kernel-patches>查看团队过去发送的patch。
7. 请在 `gitconfig` 中将 `name` 设置成自己的真名。

Issue 修复参考

Missing unwind goto ...

参考 Dan 的 Blog：内核错误处理规范

<https://staticthinking.wordpress.com/2022/04/28/free-the-last-thing-style/>

Xxx is an error pointer or valid

Return from `debugfs_create_dir`

- 1 Debugfs is kind of weird. The correct thing here is just to delete the
- 2 NULL check.
- 3

```
4 There are some times where a driver will do something weird with the
5 dir->inode pointer so in that case we need to check for errors. But if
6 the driver doesn't dereference the dir pointer then we can just delete
7 the error handling.
8
9 The commit message would say:
10
11 [PATCH] wifi: mt7601u: delete dead code checking debugfs returns
12
13 Smatch reports that:
14     drivers/net/wireless/mediatek/mt7601u/debugfs.c:130
15     mt7601u_init_debugfs() warn: 'dir' is an error pointer or valid".
16
17 Debugfs code is not supposed to need error checking so instead of
18 changing this to if (IS_ERR()) the correct thing is to just delete
19 the dead code.
20
21 The other question about this code is when are the debugfs files
22 deleted? I don't see that happening and it seems like it would lead to
23 a use after free or something.
24
25 Debugfs can only be written to by root and driver unload is a privileged
26 operation so this is not a security issue but it would be a bug.
27
28 regards,
29 dan carpenter
30
```

Patch 相关常见问题

我应该何时添加 Reviewed-by 标签?

- 只有当大家看见这样的回复邮件时，你才可以在下一版本的patch中加上reviewed-by tag，否则，贸然加上别人的reviewed-by tag会很不礼貌。



error27@gmail.com

收件人 Li Yang、hust-os-ker...@googlegroups.com

On Thu, Mar 30, 2023 at 06:50:57PM +0800, Li Yang wrote:

> Smatch reports:

>

> drivers/soc/renesas/renesas-soc.c:536 renesas_soc_init() warn:

> 'chipid' from ioremap() not released on lines: 475.

>

> If soc_dev_attr allocation is failed, function renesas_soc_init()

> will return without releasing 'chipid' from ioremap().

>

> Fix this by adding function iounmap().

>

> Fixes: cb5508e47e60 ("soc: renesas: Add support for reading product revision for RZ/G2L family")

> Signed-off-by: Li Yang <lida...@hust.edu.cn>

> ---

> v1 -> v2: Judge chipid is not NULL before call iounmap.

Reviewed-by: Dan Carpenter <err...@gmail.com>

regards,

dan carpenter

Commit Title 前面的 xxx: yyy: 是什么？如何编写合适的 Commit Title？

- 可以使用 `git log --oneline ${FILE}` 查看其他人对该文件的commit标题，并仿照他们的标题。

```
sjy@pride-poweredge-r740:~/patchSubmit/linux$ git log --oneline drivers/clock/clock.c
35170e69703e Merge branch 'clk-cleanup' into clk-next
12ca59b91d04 clk: Print an info line before disabling unused clocks
08d5ecf554f4 Merge branch 'clk-cleanup' into clk-next
5b1a1c1ab1f9 clk: remove unnecessary (void*) conversions
d02fddf24f98 clk: Use of_property_present() for testing DT property presence
1ec35eadc3b4 Merge tag 'clk-for-linus' of git://git.kernel.org/pub/scm/linux/kernel/git/clk/linux
79200d5851c8 clk: Honor CLK_OPS_PARENT_ENABLE in clk_core_is_enabled()
ca502fc6d9d9 cpuidle, clk: Remove trace_*_rcuidle()
0e2c9884cbba Merge branches 'clk-mediatek', 'clk-trace', 'clk-qcom' and 'clk-microchip' into clk-next
49e62e0d96ba clk: Add trace events for rate requests
ef13f8b64728 clk: Store clk_core for clk_rate_request
774560cf28fe clk: Initialize max_rate in struct clk_rate_request
2079d029387a clk: Initialize the clk_rate_request even if clk_core is NULL
08a32902a56e clk: Remove WARN_ON NULL parent in clk_core_init_rate_req()
2fcd8f108f25 Merge tag 'clk-for-linus' of git://git.kernel.org/pub/scm/linux/kernel/git/clk/linux
a7b78befbce2 Merge branch 'clk-rate-range' into clk-next
096f2a0c6469 clk: Update req_rate on __clk_recalc_rates()
```

我在发送第二版 Patch 时需要注意什么？

- Change log.

提交第二（n）版Patch时，需要在COMMIT MSG后面三个横线下写上，最近的change log 写在最上面，格式为：

```
1 v2 -> v3: ...
2 v1 -> v2: ...
```

例子为：

```
Fixes: 96d6392b54db ("clk: imx: Add support for i.MX8MN clock driver")
Signed-off-by: Hao Luo <m202171776@hust.edu.cn>
---
v4 → v5: Add Fixes tag
v3 → v4: Corrected the subject
v2 → v3: Corrected devm_kzalloc() is used
v1 → v2: Change devm_platform_ioremap_resource() to devm_of_iomap()
The issue is discovered by static analysis, and the patch is not tested yet.
---
```

- Subject prefix.

使用[PATCH v2]而不是[PATCH]

我的 Patch 在内部审核通过了，在发送给内核 Maintainer 之前需要注意什么？

1. 准备好通过内部审核的 patch，删除其中的 changelog 以及版本 subject prefix
2. 发送 patch 至内核社区，不要忘记抄送至 HUST OS Kernel Contribution <hust-os-kernel-patches@googlegroups.com>
3. 等待内核维护者的意见，如需要发送 Patch v2，可以在发送给内核社区之前发送到 patch copilot 处（并抄送至 hust-os-kernel-patches@googlegroups.com）进行检查
 - a. 如果是我们通过静态分析扫描出来的问题，要在 changelog 的位置写清楚，可以是这样的字段：

```
1 ---
2 This issue is found by static analyzer. The patched code has passed
3 Smatch checker, but remains untested on / by ... SoC/ real device / concrete
   running
```

把 Patch 发送到内核邮件列表之后很长时间没有回复怎么办？

可以回复“ping?”，但不要发送 PATCH V2 或 PATCH RESEND

我的第一版 Patch 存在问题，如何在第一版的基础上生成第二版的 format-patch？

- 还没有 `git send-email` 出去的 patch，此时你的 Patch 依旧是第一版，不需要添加 v2 之类的信息
 - 如果是对代码的修改有问题，将代码重新修改后，使用 `git commit --amend -asev` 命令重新提交 commit

- 如果是commit message有误，则直接使用 `git commit --amend -asev` 命令，修改commit message即可
- 修改完毕后，`git format-patch -1` 即可生成新的patch
- 已经发送出去的patch
 - 第一步和没有发送出去的patch一样，但是在format-patch的时候要指定subject `git format-patch -1 --subject-prefix="PATCH v2"`

内核邮件列表中 PATCH 主题的 [PATCH xx] 都是什么意思？我要怎么使用？

准备建立你的 patch 文件：Subject 前缀

当发送邮件时，需要为邮件主题添加一个前缀，这个前缀保存在你的 patch 文件中，格式为

[PATCH (vP) (M/N)]¹⁰

以下前缀是规范的：

[PATCH]

[PATCH v2]

[PATCH 2/5]

[PATCH v2 01/27]

¹⁰括号内的内容表示可以省略，P 为补丁版本号，M 和 N 表示补丁系列中有 N 个补丁，该补丁为第 M 个。M 和 N 高位需要用 0 填充，以便其字典序为其真实顺序。

为了修改 patch 的前缀，可以使用 `--subject-prefix` 选项

```
1 git format-patch -1 --subject-prefix="PATCH v2"
```

生成多个 patch (patch set)

```
1 $ git format-patch -2 --subject-prefix="PATCH v2"
2 0001-clk-imx-clk-imx8mp-use-dev-managed-memory-to-avoid-.patch
3 0002-clk-imx-clk-imx8mp-add-error-handling-of-of_clk_add_.patch
4
```

```
5 $ sed -n 4p ./0001-clk-imx-clk-imx8mp-use-dev_managed-memory-to-avoid-.patch
6 Subject: [PATCH v2 1/2] clk: imx: clk-imx8mp: use dev_managed memory to avoid
```

我对分配给我的任务感到困难，我应该向谁寻求帮助？

1. 先问自己啦，参考我们内部的邮件列表中的讨论，由于我们的 issue 大多数是由 smatch 扫描出来的，里面有很多相同类型的问题，也许你的问题就可以在邮件列表里找到答案
2. 参考历史 patch 的写法。用 `git blame` 看看这个文件的修改历史，或者同一文件夹下其他文件的修改历史，可能可以得到启发
3. 向 Copilot 或直接在群里提问

我发现我得到的任务是 False Positive，我应该怎么办？

可以把问题发到邮件列表里，或者在群里大家一起讨论

Patch 中的 Fixes 标签是什么意思？我何时应当添加？要怎么添加？

1. Fixes: 表示这个 Patch 修复了一个之前 commit 引入的问题。这样做是为了更方便地使定位 bug 的起源，以帮助修复 Bug。这个标签同样辅助 stable 内核团队确定这个 patch 应该打到那些版本的 stable 内核上。This is the preferred method for indicating a bug fixed by the patch. See [Describe your changes](#) for more details.

2. 步骤

- a. 首先，如果之前使用 `git checkout -b` 新建分支，那么在查找 Fixes 之前需要切换到 master 分支，参考命令 `git switch master`

- b. 使用 `git blame {file_path}` 查看引入 bug 代码的 commit hash，例如

`git blame drivers/clock/imx/clock-imx8mn.c` 之后定位到进入代码的行数

```
daeb145455146 (Peng Fan      2019-12-12 02:59:17 +0000 326)      clk_hw_data = kzalloc(struct_size(clk_hw_data, hws,
```

- c. 第一列是 commit hash 的前 12 位，可以使用命令 `git log -p {commit hash} -1` 可以打印出指定 commit 从主分支角度存在的变化差异。其中，-p 是打印变化差异，-1 是只打印一条 commit

另外可以使用浏览器输入 <https://github.com/torvalds/linux/commit/{commit hash}> 查看 commit 内容，效果等同上面命令。注意：浏览器输入的 hash 需要是 40 个字母的 hash

检查 commit 内容，**确认 commit 是否引入了 patch 修改的 bug 代码**

- d. 使用 `git log --pretty="Fixes: %h (%s)" {commit hash} -1` 返回 Fixes tag

```
Fixes: daeb14545514 ("clk: imx: imx8mn: Switch to clk_hw based API")
```

将以上返回的内容，写入 commit message 中。

如何编译修改后的内核源代码？

首先 `make allyesconfig` ，打开所有的config。

然后 `make dir/file.[ois]` （如 `make drivers/spi/spi-mux.o` 即可编译 `drivers/spi/spi-mux.c` 文件）。

git clone很慢怎么办？

- 可以使用[中科大的镜像站](#)。

```
sjy@ubuntu20-sjy:~$ git clone git://mirrors.ustc.edu.cn/linux.git
Cloning into 'linux'...
remote: Enumerating objects: 11298145, done.
^Cceiving objects: 12% (1455090/11298145), 660.93 MiB | 3.21 MiB/s
```

- 使用有线网。如果使用无线网，wireless和auto会比2.4g快。
- 给笔记本插上电，网速也会快些。

我的虚拟机很卡怎么办？

- 有服务器的同学优先使用**服务器**。
- 改为使用wsl2（Windows Subsystem for Linux）。
- [ssh连接虚拟机](#)，不使用图形化界面，可能会快一些。
- 如果仍然打算使用图形化界面的虚拟机，可以在system monitor中查看一下当前瓶颈在哪以及是哪些程序占用了资源，杀掉无关的进程并尝试升级虚拟机配置（处理器、内存等）。



Processes Resources File Systems									
Process Name	User	% CPU	ID	Memory	Disk read tot	Disk write tot	Disk read	Disk write	Priority
gnome-shell	sjy	12	1821	148.4 MiB	10.4 MiB	12.0 KiB	N/A	N/A	Normal
gnome-system-monitor	sjy	12	2294	16.5 MiB	2.7 MiB	8.0 KiB	N/A	N/A	Normal
Xorg	sjy	8	1618	41.6 MiB	8.6 MiB	60.0 KiB	N/A	N/A	Normal
vmtoolsd	sjy	0	1997	11.9 MiB	9.8 MiB	N/A	N/A	N/A	Normal
gnome-calendar	sjy	0	4296	14.8 MiB	6.6 MiB	N/A	N/A	N/A	Normal
xdg-desktop-portal	sjy	0	2382	1.3 MiB	856.0 KiB	N/A	N/A	N/A	Normal
at-spi-bus-launcher	sjy	0	1786	948.0 KiB	N/A	N/A	N/A	N/A	Normal

虚拟机设置

硬件

选项

实测从1核改为8核快了不少

设备	摘要
内存	4 GB
处理器	8
硬盘 (SCSI)	40 GB
CD/DVD (SATA)	正在使用文件 D:\mysoftwares\...
网络适配器	NAT

处理器

处理器数量(P): 8

每个处理器的内核数量(C): 1

处理器内核总数: 8