linux>REPORTING-BUGS

【linux-2.6.31】如何报 bug

Translated By: openspace
Date : 2009-11-2

[部分信息摘录自 Frohwalt Egerer 编写的 linux-kernel FAQ]

本文描述了报告 Linux bug 的流程。你不必非得使用这里描述 bug 的格式，这里所用的格式是为了描述哪些信息对开发人员有帮助——仅此而已。

如果发生故障时会在系统日志或者屏幕上显示一条“OOPS:”类型的消息，请在发送你的 bug 报告之前阅读“Documentation/oops-tracing.txt”。这篇文档描述了你该如何处理“Oops”信息以方便接收对象。

将故障发生时产生的输出发送给出现问题的内核部分的维护人员，并 cc 给相关的邮件列表。不要担心会错发给其他人。如果不确定，就将它发送给与你要做的事情相关的代码部分的负责人。如果故障重复出现，请描述如何可以重现。这比 oops 本身更重要。维护人员列表和邮件列表放在当前目录下的 MAINTAINERS 文件中。如果你知道引发问题的文件名，那么可以在当前目录中用下面的命令查找对应文件的维护人员：

```
perl scripts/get_maintainer.pl -f <filename>
```

如果是一个安全相关的 bug，请使用 MAINTAINERS 中的 Security Contact。他们会帮忙解决该 bug。查看 Documentation/SecurityBugs 获取更多信息。

如果确实不知道该给谁发送报告，就将它发送到 linux-kernel@vger.kernel.org（访问 http://www.tux.org/lkml/获取关于 linux-kernel 的更多信息）。

下面是一个发送到 Linux 内核邮件列表的 bug 报告的格式。使用标准的 bug 报告格式会防止你漏掉一些信息，而且有利于开发人员找到问题根源的一些相关信息。不过你可以采用其它格式。

首先运行脚本 scripts/ver_linux，该脚本会检测一些重要子系统的版本。执行该脚本时使用命令“sh scripts/ver_linux”。

根据收集到的信息填写 bug 报告表单，然后把它发送到邮件列表，标题格式采用“PROBLEM: <one line summary from [1.]>”的形式，这样方便开发人员辨认。

[1.] 问题概述，一行即可：

[2.] 问题/报告的完整描述：

[3.] 关键词（例如 modules、networking、kernel）：

[4.] 内核信息

[4.1.] 内核版本（可以从/proc/version 获取）：

[4.2.] 内核配置文件.config：

[5.] 没有此 bug 的最近的内核版本：

[6.] Oops 的输出。如果适当的话解析消息中的符号链接（参考 Documentation/oops-tracing.txt）

[7.] 产生问题的简短 shell 脚本或者示例程序（如果可以的话）

[8.] 环境

[8.1.] 软件（将脚本 ver_linux 的输出添加到这里）

[8.2.] 处理器信息（可以从/proc/cpuinfo 获取）：

[8.3.] 模块信息（可以从/proc/modules 获取）：

[8.4.] 加载的驱动以及硬件相关信息（/proc/ioports、/proc/iomem）

[8.5.] PCI 信息（以 root 用户身份执行‘lspci -vvv’）

[8.6.] SCSI 信息（可以从/proc/scsi/scsi 获取）

[8.7.] 其他可能与问题有关的信息（请查看/proc，记录所有你认为相关的信息）：

[X.] 其他附加信息、补丁、修复、问题的解决办法：

谢谢！

【原文】

[Some of this is taken from Frohwalt Egerer's original linux-kernel FAQ]

What follows is a suggested procedure for reporting Linux bugs. You aren't obliged to use the bug reporting format, it is provided as a guide to the kind of information that can be useful to developers - no more.

If the failure includes an "OOPS:" type message in your log or on screen please read "Documentation/oops-tracing.txt" before posting your bug report. This explains what you should do with the "Oops" information to make it useful to the recipient.

Send the output to the maintainer of the kernel area that seems to be involved with the problem, and cc the relevant mailing list. Don't worry too much about getting the wrong person. If you are unsure send it to the person responsible for the code relevant to what you were doing. If it occurs repeatably try and describe how to recreate it. That is worth even more than the oops itself. The list of maintainers and mailing lists is in the MAINTAINERS file in this directory. If you know the file name that causes the problem you can use the following command in this directory to find some of the maintainers of that file:

```
perl scripts/get_maintainer.pl -f <filename>
```

If it is a security bug, please copy the Security Contact listed in the MAINTAINERS file. They can help coordinate bugfix and disclosure. See Documentation/SecurityBugs for more information.

If you are totally stumped as to whom to send the report, send it to linux-kernel@vger.kernel.org. (For more information on the linux-kernel mailing list see http://www.tux.org/lkml/).

This is a suggested format for a bug report sent to the Linux kernel mailing list. Having a standardized bug report form makes it easier for you not to overlook things, and easier for the developers to find the pieces of information they're really interested in. Don't feel you have to follow it.

First run the ver_linux script included as scripts/ver_linux, which reports the version of some important subsystems. Run this script with the command "sh scripts/ver_linux".

Use that information to fill in all fields of the bug report form, and post it to the mailing list with a subject of "PROBLEM: <one line summary from [1.]>" for easy identification by the developers.

[1.] One line summary of the problem:
[2.] Full description of the problem/report:
[3.] Keywords (i.e., modules, networking, kernel):
[4.] Kernel information
[4.1.] Kernel version (from /proc/version):
[4.2.] Kernel .config file:
[5.] Most recent kernel version which did not have the bug:
[6.] Output of Oops.. message (if applicable) with symbolic information

```
     resolved (see Documentation/oops-tracing.txt)
[7.] A small shell script or example program which triggers the
     problem (if possible)
[8.] Environment
[8.1.] Software (add the output of the ver_linux script here)
[8.2.] Processor information (from /proc/cpuinfo):
[8.3.] Module information (from /proc/modules):
[8.4.] Loaded driver and hardware information (/proc/ioports, /proc/iomem)
[8.5.] PCI information ('lspci -vvv' as root)
[8.6.] SCSI information (from /proc/scsi/scsi)
[8.7.] Other information that might be relevant to the problem
       (please look in /proc and include all information that you
       think to be relevant):
[X.] Other notes, patches, fixes, workarounds:


Thank you
```