

# Bugtracking in Debian und Ubuntu im Vergleich

Daniel Panayi

31. Juli 2015

## Inhaltsverzeichnis

<b>1</b>	<b>Einleitung</b>	<b>2</b>
<b>2</b>	<b>Bugtracking im Open-Source-Bereich</b>	<b>3</b>
<b>3</b>	<b>Debian</b>	<b>4</b>
3.1	Überblick . . . . .	4
3.2	Bugtracking in Debian . . . . .	5
<b>4</b>	<b>Ubuntu</b>	<b>7</b>
4.1	Überblick . . . . .	7
4.2	Bugtracking in Ubuntu . . . . .	7
<b>5</b>	<b>Fazit</b>	<b>9</b>

# 1 Einleitung

In dieser Arbeit wird verglichen, wie in den zwei GNU/Linux Distributionen Debian und Ubuntu Softwarefehler gemeldet und behoben werden. Insbesondere werden die verschiedenen Werkzeuge und Systeme betrachtet, die für diese Aufgabe entwickelt wurden und benutzt werden.

Zunächst wird grundlegendes über Bugtracking im Open-Source-Bereich erläutert. Daraufhin wird Debians Paketverwaltungssystem, das Bug Tracking System (BTS) und die Werkzeuge, mit denen Debian-Nutzer an das BTS Fehler melden können, dargestellt. Dann wird gezeigt, wie Debian-Entwickler das System nutzen, um mit Fehlermeldungen umzugehen. Ubuntu ist Debian-Derivat und benutzt ein Paketverwaltungssystem, das auf dem von Debian aufbaut. Allerdings wurden von den Ubuntu-Entwicklern andere Bugtracking-Tools für die speziellen Anforderungen von Ubuntu und seinen Nutzern entwickelt. Auch diese werden dargestellt. Anschließend werden die Vor- und Nachteile der zwei Systeme besprochen.

Die Distributionen Debian und Ubuntu sind sich sehr ähnlich, weisen jedoch fundamentale Unterschiede darin auf, wie in den Entwicklerteams gearbeitet wird. Deshalb ist dieser Vergleich sehr fruchtbar, und die Auseinandersetzung damit kann aufzeigen, welche Werkzeuge und Bugtracking-Systeme bei welchen Open-Source Projekten am sinnvollsten sind.

## 2 Bugtracking im Open-Source-Bereich

*Open Source Software* ist Software, deren Quelltext offenliegt. Strenger genommen, ist es Software, die unter einer Lizenz steht, die von der *Open Source Initiative (OSI)* anerkannt wird. *Freie Software* ist hingegen Software, die jedem Nutzer volle Nutzungsrechte gewährt.

Eric S. Raymond hat 1997 in seinem Essay *Die Kathedrale und der Basar* die im Open-Source-Bereich verbreitete Softwareentwicklungsmethode des *Basars* mit der zuvor gebräuchlichen Methode der *Kathedrale* verglichen.

Beim Kathedralen-Modell gibt es einen Chefarchitekten, der ein hierarchisch organisiertes Team von Spezialisten überwacht. Der Quellcode wird, wenn überhaupt, nur nach der Veröffentlichung einer neuen Softwareversion veröffentlicht. Sonst ist er nur von den beauftragten Entwicklern einsehbar. Es gibt einen Bauplan, der ein klares Ziel hat, und wenn das Ziel erreicht wurde, ist das Projekt fertig. Dieses Modell ist das übliche Entwicklungsmodell proprietärer Software, wird aber teilweise auch bei Projekten freier und Open-Source Software eingesetzt.

Das Basar-Modell hingegen ist dadurch ausgezeichnet, dass der Quellcode zu jedem Zeitpunkt für jeden im Internet einsehbar ist. Meistens gibt es eine Person, die als Wächter fungiert, und achtet, dass gewisse Standards eingehalten werden. Die Software besteht meist aus kleinen Teilen, und Entwickler können frei wählen, bei welchen Teil sie arbeiten möchten. Die Software ist nie wirklich fertig und kann immer verbessert werden. Die meisten Open-Source Projekte verlaufen nach diesem Entwicklungsmodell. Raymond behauptet, dieses Modell führt zu besserer Software.

Im selben Essay formulierte Raymond den *Linus' Law*, der besagt, dass "given enough eyeballs, all bugs are shallow." Der Satz soll aussagen, dass mit einer möglichst großen Menge an Beta-Testern und Mitentwicklern fast alle Probleme möglichst schnell entdeckt und gelöst werden. Dieser Satz gilt im Open-Source-Bereich als wichtiger Grundsatz. Nutzer von freier und Open Source Software werden auch als Mitentwickler betrachtet und werden oft ermutigt, Fehler und Probleme zu melden und sich in verschiedensten Arten bei der Entwicklung zu beteiligen.

Projekte freier oder Open Source Software können in folgenden vier Kategorien aufgeteilt werden:

- Eigenständige Projekte, die einen spezifischen Zweck erfüllen. Bsp: der Linux Kern, Mozilla Firefox, OpenOffice.
- Distributionen. Sammlungen von Software, für einen gemeinsamen Zweck. Bsp: Cygwin, GNU/Linux Distributionen wie Debian, Ubuntu, Fedora usw.
- Komplette Betriebssysteme in einem Versionsverwaltungssystem. Bsp: BSD-Derivate wie FreeBSD, OpenBSD usw.
- Bücher oder Dokumente.

Im folgenden werden die Bugtracking-Systeme von zwei GNU/Linux Distributionen betrachtet.

## 3 Debian

### 3.1 Überblick

Debian ist eine GNU/Linux Distribution, die komplett aus freier und open source Software besteht. Auf seinen Servern enthält Debian über 43000 Softwarepakete, die von ca. 20100 Quellpaketen gebaut sind.

Die Pakete werden in drei Bereichen (Branches) angeboten, *stable*, *testing*, und *unstable*. Dadurch werden auch drei Varianten des Betriebssystems angeboten. Die Pakete in *stable* werden nur verändert, wenn es Sicherheitsaktualisierungen gibt. *testing* ist der Kandidat für die nächste *stable*, weshalb in *testing* nach und nach Aktualisierungen und neue Pakete eingebunden werden. *unstable* ist der erste Anlaufpunkt für neue Versionen von Paketen, bevor sie in *testing* integriert werden und ist weniger für den alltäglichen, produktiven Einsatz gedacht, als für das Testen auf Fehlern von neuen Paketen.

*dpkg* ist die Basis der Paketverwaltung des Betriebssystems Debian und das grundlegende Programm zum Installieren und Manipulieren von Debian-Binärpaketen (Dateiendung „.deb“). Um eine einfache Möglichkeit zur Suche, Installation und Aktualisierung von Programmpaketen anzubieten, kommt das Paketverwaltungssystem *Advanced Packaging Tool (APT)* zum Einsatz, das *dpkg* zur eigentlichen Paketverwaltung benutzt.

Leute, die an den hochgeladenen Paketen arbeiten, heißen *Maintainer*. Jedes Paket hat einen Maintainer. Debian Maintainer (DMs) haben eingeschränkte Rechte zum hochladen von bestimmten Paketen. Nur einige von ihnen sind offizielle Mitglieder des Debian Projekts (*Debian Developer* - Leute, die durch einen Bewerbungsprozess gegangen sind und mehr Rechte haben). Zu den Aufgaben von DMs gehört das Verfolgen von neuen *upstream* Veröffentlichungen (Quellpakete) und das Prüfen, ob Pakete die vom Debian Projekt festgelegten Qualitätsstandards erfüllen. Außerdem tragen sie die Verantwortung für die Bugs eines Pakets.

## 3.2 Bugtracking in Debian

*BTS* steht für *Bug Tracking System* und ist ein Softwaresystem, das vom Debian Projekt für die Bedürfnisse von Debian erstellt wurde.

Seine Hauptfeatures sind folgende:

- Nutzer müssen sich nicht authentifizieren, um Fehler zu melden.
- Es ist manövrierbar per E-Mail.
- Es hat eine Web-Schnittstelle, wo Bugs betrachtet werden können.
- Ein Filtersystem erlaubt verschiedene Klassen von Bugs zu betrachten.
- Es interagiert automatisch mit dem Archiv.

Fehler werden per E-Mail an `submit@bugs.debian.org` gemeldet. Im Betreff der E-Mail kommt eine Beschreibung des Fehlers, die später im System als Titel des Fehlers übernommen wird. Im Body der E-Mail werden die restlichen Informationen eingetragen. In den ersten zwei Linien gehört ein Pseudo-Header, der aus dem Namen und der Version des Pakets besteht. Dieser Pseudo-Header kann auch mit anderen Informationen erweitert werden, wie z.B. eine Einschätzung der Severität des Fehlers. Im restlichen Body gehört:

- Der Inhalt von etwaigen Fehlermeldungen des Programms.
- Eine Beschreibung des Verhaltens, das zum Fehler geführt hat.
- Eine Beschreibung des erwarteten und tatsächlichen Verhaltens.
- Ein Vorschlag zur Behebung, falls es einen gibt.
- Details über die Konfigurierung des Programms.
- Die Versionen der Pakete, von denen das Paket abhängt.
- Informationen über das System des Nutzers, wie die Hardware, Kernelversion und mehr.

Ein Beispiel einer Fehlermeldung:

```
To: submit@bugs.debian.org
From: diligent@testing.linux.org
Subject: Hello says 'goodbye'
```

```
Package: hello
Version: 1.3-16
```

```
When I invoke 'hello' without arguments from an ordinary shell
prompt it prints 'goodbye', rather than the expected 'hello, world'.
Here is a transcript:
```

```
$ hello
goodbye
$ /usr/bin/hello
goodbye
$
```

```
I suggest that the output string, in hello.c, be corrected.
```

```
I am using Debian GNU/Linux 2.2, kernel 2.2.17-pre-patch-13
and libc6 2.1.3-10.
```

Um das Melden von Fehlern zu vereinfachen, wurde das Programm *reportbug* entwickelt, das mit einer Reihe von Nachfragen in der Kommandozeile einige Informationen für eine Fehlermeldung zusammenträgt. Es wird den Nutzern empfohlen, erst in der Datenbank des *BTS* zu prüfen, ob der Fehler schon gemeldet wurde, bevor sie *reportbug* benutzen.

Nachdem die Fehlermeldung abgesendet wurde, kriegt sie vom System eine Nummer. Der Titel des Fehlers bekommt im System einen Zusatz **Bug#*nnn*** wo ***nnn*** die Nummer des Fehlers ist. Anhand des Names des Pakets, der in der ersten Linie des Pseudo-Headers angegeben wurde, wird die Fehlermeldung zum Maintainer des Pakets gesendet. Nachdem der Fehler behoben wurde, kann der Maintainer des Pakets mit einer E-Mail an ***nnn-done@bugs.debian.org*** den Fehler als behoben markieren.

## 4 Ubuntu

### 4.1 Überblick

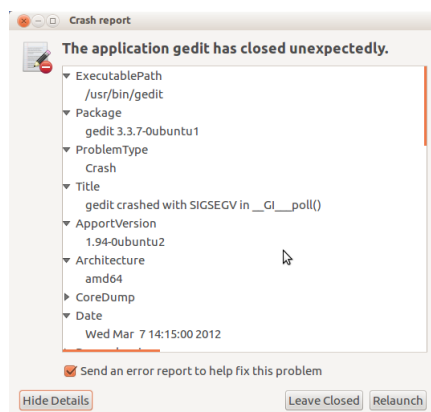
Ubuntu ist eine weitere GNU/Linux Distribution, die von der Firma Canonical entwickelt wird. Es ist ein Betriebssystem, das das Ziel hat, leicht installierbar und einfach zu bedienen zu sein. Inzwischen ist Ubuntu eine der bekanntesten und meistgenutzten Linux-Distributionen mit einer geschätzten Nutzerzahl von 25 Millionen. Pro Woche werden im Durchschnitt 1800 Bugs gemeldet. Somit steht Ubuntu vor anderen Herausforderungen als Debian.

Ubuntu ist auf der technischen Architektur und Infrastruktur von Debian aufgebaut. Das Paketformat (.deb), Werkzeuge der Paketverwaltung (APT) und diverse andere Strukturen wurden direkt übernommen. Der Arbeitsaufwand für die Wartung der vielen Pakete wird dadurch reduziert, dass Pakete vom *unstable* Bereich der Debian Pakete importiert werden, und bei Bedarf für Ubuntu modifiziert.

### 4.2 Bugtracking in Ubuntu

*Launchpad* ist eine Sammlung von Webanwendungen, die von Canonical für Ubuntu entwickelt wurden. Es dient Ubuntu und vielen anderen Projekten als Bugtracker. Um Fehler in Launchpad zu melden, wird ein Launchpad-Konto benötigt.

*apport* ist eine Anwendung für die Fehlerberichterstattung von Programmen, die auch von Canonical entwickelt wird und ist ein vorinstallierter Standard-Bestandteil von Ubuntu. *apport* erkennt Programmabstürze, startet danach automatisch und sammelt Informationen, die für einen Fehlerbericht zusammengetragen werden. Anschließend wird dem Nutzer eine Schnittstelle mit nützlichen Informationen rund um den Fehlerbericht präsentiert, der wiederum mit einem Knopfdruck an Launchpad gesendet werden können.



Sobald *apport* einen Fehlerbericht an *Launchpad* gesendet hat, wird ein neues Webbrowser-Fenster geöffnet. *Launchpad* sucht automatisch nach Fehlerbericht-

en mit ähnlichen Eckdaten. Die Ergebnisse dieser Suche werden wie im nächsten Bild zu sehen ist angezeigt.

Please describe the bug in a few words, for example, "weather applet crashes on logout":

**Summary:**

Do any of the following bugs describe the bug you're trying to report?

- ▷ **#243222 Evince crashes with Segfault when opening the same PDF repeatedly**  
Invalid (1 comment) last updated 2008-06-26 [view this bug](#)
- ▷ **#556511 evince crashed after opening an encrypted pdf**  
Fix Released (10 comments) last updated 2010-05-29 [view this bug](#)
- ▷ **#251202 OpenOffice crashes while opening large pdf file with evince**  
Invalid (3 comments) last updated 2008-07-30 [view this bug](#)
- ▷ **#146266 evince crashed with SIGSEGV on opening PDF files not created by OpenOffice**  
Invalid (5 comments) last updated 2007-10-19 [view this bug](#)
- ▷ **#277294 evince crashed with SIGFPE, trying to seek in KXTGA930.PDF**  
Open (27 comments) last updated 2010-05-08 [view this bug](#)
- ▷ **#436197 evince crashed with SIGSEGV in TextWord::TextWord()**  
Open (9 comments) last updated 2010-01-31 [view this bug](#)
- ▷ **#381067 evince crashes with segfault after opening pdf file**  
Fix Released (5 comments) last updated 2009-05-30 [view this bug](#)
- ▷ **#459735 Evince crash opening certain PDF files**  
Fix Released (7 comments) last updated 2010-05-12 [view this bug](#)
- ▷ **#580965 opening password secured pdf causes segfault and evince crash**  
Invalid (4 comments) last updated 2010-05-19 [view this bug](#)
- ▷ **#348565 Evince crashes when opening JSTOR PDF files**  
Fix Released (2 comments) last updated 2009-03-25 [view this bug](#)

Falls einer der schon eingetragenen Fehler schon den Fehler beschreibt, den ein Nutzer melden will, kann ein Nutzer mit einem Knopfdruck im System eintragen, dass er auch vom selben Fehler betroffen ist. Im Unterschied zu Debian, muss hier ein Nutzer also nicht manuell das System navigieren, um nach den Fehler zu suchen.



## 5 Fazit

Es wurde gezeigt, welche Werkzeuge die Entwicklerteams von Debian und Ubuntu entwickelt haben, um Fehler zu melden und zu beheben. Debian hat mit BTS und reportbug ein recht primitives System, das von Nutzern und Entwicklern verlangt, viel Zeit zu investieren, um sich damit vertraut zu machen, bevor sie es nutzen können. Launchpad und apport machen es mit einer gut ausgearbeiteten Webschnittstelle für Ubuntu-Nutzer sehr leicht, Fehler zu melden. Auch für Ubuntu-Entwickler und Maintainer bietet Ubuntu System erhebliche Vorteile, da sie damit einen guten Überblick über die Fehler haben und sich leicht mit Nutzern und Mitentwicklern bei der Fehlerbehebung koordinieren können. Allerdings benötigt ein solches System auch große finanzielle Investitionen, um realisiert zu werden. Da Debian, im Gegensatz zu Ubuntu, nicht von einer Firma mit großen Einnahmen entwickelt wird, ist es für das Debian Projekt schwieriger, die finanziellen Mittel für ein System wie Launchpad zu kriegen.