

GNU Debugger workshop

Jürgen Weigert

Teamlead Tools RD OPS

jw@suse.de

2008-11-06

Novell[®]

Inhalt

- Was ist ein Bug?
- Allgemeine Arbeitstechniken
- Was kann GDB?
- Arbeiten mit GDB
- Fehlersuche am Beispiel
- Ausblick

Was ist ein Bug?

GDB hilft bei

- Programmabsturz
segmentation fault, signal 11
- Endlos-Schleife
*dauerhaft 100% CPU,
zyklische Dialoge (Intelligenztest?)*
- Fehlverhalten
Logikfehler, korrupte Daten
- Blockierung
Programm wartet erfolglos

Was ist ein Bug? -2-

Andere Bugs

- Mangelnde Benutzerführung
- Mangelhafte Geschwindigkeit
- Speicherüberlauf
- Fehler zur Compilezeit
- Dokumentationsfehler
- Konfigurationsfehler
- Architektur/Designfehler

Andere Werkzeuge

- **strace**
- **ltrace**
- **valgrind**
- **printf()**
- **lint**

Allgemeine Arbeitstechniken (Vorbereitung)

- Reproduzierbarkeit sicherstellen
 - Was ist nötig um den Fehlerfall zu wiederholen?
- Reduktion
 - Was kann man weglassen, ohne das der Fehler verschwindet?
- Datensammlung (Symptome)
 - Logfiles finden, Konfigurationsdateien, screenshots
- Erwartungen kontrollieren
 - Soll-Ist Vergleich, Dokumentation lesen
- Bauumgebung schaffen
 - Source Code auspacken, Abhängigkeiten erfüllen

Allgemeine Arbeitstechniken (Arbeitsschritte -1-)

- Geschwätzigkeit erhöhen
 - Verbosity Option **-v**, (shell script '**set -x**')
 - **printf()** debugging
- Versionsvergleich
 - Gleicher Fehler in älteren Versionen? (Patches?)
 - Ander RPM-Pakete, **svn co -r**
- Eingrenzung durch Intervallhalbierung
 - In einer Datei: systematisches Auskommentieren
 - Unter Versionskontrolle: **svn co -r**, **git bisect**

Allgemeine Arbeitstechniken (Arbeitsschritte -2-)

- Testprogramme
 - Eigenes **main()** für Programmteile / Bibliotheken
 - Aufrufe per shell Script, um Abläufe zu reproduzieren
- Ablaufprotokolle erstellen
 - Systemaufrufe (**strace**), Bibliotheksaufrufe (**ltrace**)
 - Speichernutzungsprotokoll (**valgrind**)
int a[10]; a[10] = 13;
char *u; if (strlen(u) > 0) ...
 - Crashdumps, stack backtraces sammeln (**gdb**)

Allgemeine Arbeitstechniken (Arbeitsschritte -3-)

- Dokumentation lesen
 - Beschreibung von Bibliotheksfunktionen (**man 3**)
 - Systemaufrufe verstehen lernen (**man 2**)
- Hilferuf
 - Experten befragen
 - Bugzilla
 - > Was ist ein guter bugreport?
 - » <https://bugzilla.novell.com/page.cgi?id=bug-writing.html>
 - » <https://bugzilla.novell.com/docs/html/bugreports.html>
 - » http://en.opensuse.org/Bugs#Reporting_a_Bug
 - » [https://innerweb.novell.com/organizations/engineering/pqsc/Defect Management Process.pdf](https://innerweb.novell.com/organizations/engineering/pqsc/Defect%20Management%20Process.pdf)

Allgemeine Arbeitstechniken (Nachbereitung)

- Dokumentation des Eingriffes
 - Kommentare, ChangeLog
- Testen (Regression)
 - Vorhandene Tests durchführen
 - Neuen Test schreiben, der den behobenen Fehler prüft
- Abgabe
 - Versionsnummer erhöhen?
 - Patch erstellen, upstream benachrichtigen
 - **svn checkin, osc ci, submitpac**

Was kann GDB?

\$ ***gdb programm core***

- Nur Code auflisten, Stack Backtrace, Variablen ausgeben (*Post Mortem Analyse*)

\$ ***gdb programm processID***

\$ ***gdb --args programm parameter ...***

- Starten, unterbrechen, Code auflisten, Variablen ausgeben/ändern, Stack Backtrace, Funktionen aufrufen, weiterlaufen lassen.

Arbeiten mit GDB

(wichtige Kommandos im GDB)

run [parameter ...]	where	break
print [/x] expression	up	disable bpt
ptype variable	down	enable bpt
list [file.c:linenumber]	next	info ...
set variable=expression	step	help
CTRL-C	continue	quit

Syntax für Ausdrücke (expressions):

- Alles was in C erlaubt ist, z.B. **p (123&0xf0)>>4**
- **@** für array-Ausgabe, z.B. **p Prime[0]@50**

Arbeiten mit GDB (Arbeitsumgebung)

\$ ulimit -c unlimited

- erlaube coredumps

\$ gcc -g -Wall -O0

- Makefile, CFLAGS=-g
- Mit Debuginfo compilieren, ohne Optimierung
-

- Evtl. Debuginfo Pakete installieren für Bibliotheken
- Zwei/drei grosse Shells nebeneinander (für Editor, Compiler, Debugger)

Fehlersuche am Beispiel

```
$ wget novell/gdb/prime-0.3.tar.bz2
```

```
$ tar xvf prime-0.3.tar.bz2
```

```
$ cd prime-0.3
```

```
$ cc -o prime main.c prime.c
```

```
$ ./prime
```

Bitte obere Schranke eingeben: **10**

2 ist Primzahl

3 ist Primzahl

5 ist Primzahl

7 ist Primzahl

... so sollte es sein

Viel Erfolg!

Ausblick (was jetzt noch fehlt)

- Fehlervermeidungsstrategien
 - > *Test Driven Development*, **assert()**
 - > Compiler Warnungen, **lint**
- C++ demangling
 - > Namen und Signaturen
 - > QT4 debugging
- Netzwerk Fehlersuche
 - > Warum geht das nur manchmal?
\$ **netcat -l -p 8888 localhost**
\$ **telnet localhost 8888**
- Graphische Oberflächen
 - > **ddd, eclipse**

Ausblick (was GDB nicht kann)

- GDB kann keine Syntax-Fehler finden
 - Hier helfen **lint** und **gcc -Wall -O2**
- GDB kann den Programm Code nicht verändern
- GDB zeigt optimierten Code 'nicht richtig' an
- GDB kennt keine CPP-Macros
- GDB kann nicht rückwärts laufen
- GDB findet oft nur ein Symptom, statt der Ursache
- GDB kann keine Scriptsprachen debuggen
 - Dazu gibt es eigene (meist ganz ähnliche) Debugger

Ausblick (was GDB sonst noch alles kann)

\$ info gdb

<http://www.gnu.org/software/gdb/documentation/>

Novell®

Unpublished Work of Novell, Inc. All Rights Reserved.

This work is an unpublished work and contains confidential, proprietary, and trade secret information of Novell, Inc. Access to this work is restricted to Novell employees who have a need to know to perform tasks within the scope of their assignments. No part of this work may be practiced, performed, copied, distributed, revised, modified, translated, abridged, condensed, expanded, collected, or adapted without the prior written consent of Novell, Inc. Any use or exploitation of this work without authorization could subject the perpetrator to criminal and civil liability.

General Disclaimer

This document is not to be construed as a promise by any participating company to develop, deliver, or market a product. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. Novell, Inc. makes no representations or warranties with respect to the contents of this document, and specifically disclaims any express or implied warranties of merchantability or fitness for any particular purpose. The development, release, and timing of features or functionality described for Novell products remains at the sole discretion of Novell. Further, Novell, Inc. reserves the right to revise this document and to make changes to its content, at any time, without obligation to notify any person or entity of such revisions or changes. All Novell marks referenced in this presentation are trademarks or registered trademarks of Novell, Inc. in the United States and other countries. All third-party trademarks are the property of their respective owners.

