
1 Übungsaufgabe #3: Fehler finden und beheben

Ziel dieser Aufgabe ist es, Fehler in den Quellen eines großen Open-Source-Projekts zu finden, diese zu beheben und die Änderungen in einem sinnvollen Format an die Entwickler weiterzugeben.

Zu diesem Zweck haben wir in eine Version des Linux-Kerns verschiedene Fehler eingebaut, die mit der in *Übungsaufgabe #1* aufgebauten Arbeitsumgebung gefunden und durch Änderungen am Quelltext behoben werden sollen.

1.1 Fehler finden

Der in

```
/proj/i4passt/kernel/linux-borked.git  
/proj/i4passt/kernel/linux-borked.config
```

mitsamt passender `.config` zu findende Quelltextbaum wurde von uns mit 11 Fehlern in verschiedenen Teilen des Systemkerns versehen.

Um die Fehler zu finden, muss der Kern aus den von uns zur Verfügung gestellten Quellen übersetzt und gestartet werden. Danach soll das System „normal“ genutzt und dabei auftretende Fehler aufgespürt werden. Dies kann einerseits mit dem in den Übungen vorgestellten Werkzeug *kgdb* wie auch z.B. durch `printk`-Debugausgaben geschehen.

Hierbei ist zu beachten, dass *kgdb* nicht in allen Fällen das optimale Werkzeug für die Behebung des Fehlers sein muss. Ihr sollt auch üben je nach Situation geeignete Debugging-Techniken selbst zu wählen und geeignet anzuwenden.

1.2 Fehler beheben

Sobald ein Fehler in den Systemkernquellen lokalisiert wurde, soll dieser durch Änderung des Quelltext behoben werden und ein *unified diff* mit den entsprechenden Änderungen erzeugt werden. Dieses *unified diff* soll anschließend als Patch per E-Mail an `linux-kernel@i4.cs.fau.de` gesendet werden. Hierbei ist darauf zu achten, dass der Patch *inline* und nicht als Anhang gesendet wird.

Bei den Fehlern kann davon ausgegangen werden, dass diese boshaft in den Quelltext eingebaut wurden, weshalb der gefundene Quelltextabschnitt nicht unbedingt eine sinnvolle Funktion erfüllen muss.

Das Beheben eines Fehlers kann dazu führen, dass sich andere Fehler im Kern auswirken können, weshalb hier ein iterativer Prozess der Fehlersuche und -behebung nötig wird.

Aufgaben:

- Systemkern aus `/proj/i4passt/kernel/linux-borked.git` übersetzen
- Fehler im Kern finden (*kgdb* / `printk` / ...)
- Gefundene Fehler durch Änderungen am Quelltext beheben
- Patches erzeugen und an `linux-kernel@i4.cs.fau.de` senden

Hinweise:

- Nicht alle Fehler treten während der Startphase des Systems auf oder ziehen einen Absturz nach sich. Um alle Fehler zu finden, muss das System „normal“ benutzt werden. Weiterhin können auch nicht-gravierende Probleme während des Systemstarts auf einen Systemkernfehler hindeuten.
- Nicht immer ist *kgdb* das optimale Werkzeug, um die eingebauten Fehler zu finden. Manchmal führen strategisch platzierte `printk`-Anweisungen schneller zum Ziel.
- Schickt gefundene Patches möglichst bald ein. Wir simulieren in dieser Aufgabe das Geschehen auf einer „richtigen“ Projektmailingliste, was bedeutet, dass es Rückmeldungen geben wird auf die Ihr reagieren müsst. Schickt also eure Patches am besten sofort, das gibt euch mehr Zeit für unsere eventuellen Änderungswünsche. Und Ihr könnt eventuell angesprochene Fehler in den nachfolgenden Patches direkt vermeiden, statt dasselbe Problem in mehreren Patches beheben zu müssen und diese erneut einzureichen.
- Wir empfehlen zur Erstellung und Einreichung der Patche sehr die Verwendung von `git` mit dem oben angegebenen Repository als Basis. Alles andere führt sehr leicht zu nicht sauber anwendbaren Patchen.

-
- Von den (je nach genauer Zählung) 11 Fehlern müsst Ihr nicht alle finden. Wir halten eine Anzahl von 5 gefundenen Fehlern für eine ausreichende Leistung. In die Bewertung geht neben der Zahl und Schwierigkeit der gefundenen Fehler auch insbesondere ein wie Ihr auf von uns angesprochene Probleme reagiert und wie sorgfältig, sauber und konform der im Linux-Kernel üblichen Kodier- und Patchrichtlinien eure Einreichungen sind.

1.3 Abgabe: am 04.06.2019