

## Übungsaufgabe #4: Statische Analyse mit splint

10.12.2008

In dieser Aufgabe soll die Lösung der Aufgabe #2, `filecrypt.c`, verbessert werden. Dazu wird die Lösung mit einem Werkzeug zur statischen Analyse untersucht.

Teilaufgaben:

- a) Sichern Sie sich ihre alte Lösung. Vorzugsweise benutzen Sie ein Versionskontrollsystem (svn, git, bzd, hg, etc.) um ihre Zwischenstände der folgenden Teilaufgaben einfach vergleichen zu können.
- b) Falls noch nicht geschehen, schreiben Sie ein Makefile und registrieren Sie es mit ihrem Versionskontrollsystem. Notieren Sie Compilerflags in der Variable `CFLAGS`.
- c) Übersetzen Sie ihr Programm mit `-std=c89 -Wall -Werror`. Untersuchen Sie das GCC Manual in Kapitel 3.8 nach zusätzlichen Schaltern für zusätzliche Warnungen. Sichern Sie diese Version in ihrem Versionskontrollsystem.
- d) Benutzen Sie `splint` mit den Schaltern `-weak +posixlibs`. Erweitern Sie dazu ihr Makefile um ein Ziel `splint` und eine Variable `SFLAGS`. Untersuchen und dokumentieren Sie die aufgetretenen Meldungen von `splint`. Fügen Sie weitere `splint`-Schalter in die Variable `SFLAGS` hinzu und begründen Sie deren Verwendung in Form von Kommentaren.
- e) Verbessern Sie ihren Programmtext unter Zuhilfenahme der von `splint` gemeldeten Fehlern. Prüfen Sie dabei bei jeder Änderung, ob der von `splint` gemeldete Fehler gerechtfertigt ist.
- f) Benutzen Sie `splint` ohne den Schalter `-weak`. Dies aktiviert weitere Analysen, darunter auch eine Analyse zum Speichermanagement. Annotieren Sie ihren Programmtext, und prüfen Sie ob die von `splint` erzeugten Warnungen gerechtfertigt sind.

Die Abgabe der bearbeiteten Aufgabe erfolgt in den Übungen am 17. oder 19.12. Die Bearbeitung kann in Gruppen zu zwei oder drei Personen erfolgen.

## Übungen zu Systemsicherheit