
1 Übungsaufgabe #1: Arbeitsumgebung einrichten

Ziel dieser Übungsaufgabe ist die Einrichtung einer Entwicklungsumgebung für das Praktikum. Die Umgebung soll Euch erlauben Eure Änderungen am Linuxkern in einer virtuellen Maschine zu testen. Im Verlauf der Aufgabe lernt Ihr die Werkzeuge kennen die Ihr im weiteren Praktikumsverlauf benötigen werdet.

1.1 Erstellen eines lauffähigen Linux-Systems

Um das Arbeiten am Linuxkern möglichst einfach und effizient zu gestalten, werden wir das Linux in einer virtuellen Maschine (VM) ausführen. Wir empfehlen dazu die Verwendung von `kvm/qemu`. `kvm/qemu` sind bereits auf den CIP Rechnern vorinstalliert.

Da ein Linuxkern alleine nicht gerade hilfreich ist, wird im ersten Schritt ein Debian Linux als Laufzeitumgebung, bestehend aus Dateisystem, Gerätedateien, Systemprogramme wie `init`, `sh`, `ls` usw. in der VM installiert.

Die Distribution kann unter <http://ftp.fau.de/debian> (auf aktuelle Distroversion & 64Bit Architektur achten) bezogen werden. Benötigt werden die `linux` und die `initrd.gz` Dateien.

1.2 Kernelquellen beschaffen, konfigurieren und übersetzen

Bezieht mit `git` die unter `/proj/i4passt/kernel/linux-stable.git` bereitgestellten Linuxquellen. Bevor Ihr den Kern übersetzen könnt, müsst Ihr eine Kernel-Konfiguration erstellen. Die Konfiguration muss mindestens die passenden Treiber für die „Hardware“ enthalten. Außerdem sollten die Debugging-Features (`kgdb`) aktiviert werden.

Den konfigurierten Kern müsst Ihr nun noch übersetzen. Um weitere Übersetzungsläufe zu beschleunigen, bietet sich die Verwendung von Software wie `ccache` an.

1.3 Booten der VM mit eigenem Kernel

Den übersetzten Kern könnt Ihr mit der Umgebung, die Ihr in 1.1 erstellt habt, starten. Verwendet dazu die in der Tafelübung vorgestellten Befehle. Es bietet sich an ein Skript mit den notwendigen Befehlen zu schreiben. Macht Euch dazu auch mit der Dokumentation von `kvm/qemu` vertraut. Ihr solltet den Befehl und die Optionen erklären können.

1.4 Debugging mit kgdb

Abschließend soll das Debugging des Kerns getestet werden. Setzt dazu mit `kgdb` einen Breakpoint innerhalb Eures Kerns, um ihn dann „im Betrieb“ anzuhalten.

Aufgaben:

- Einrichten einer Umgebung in einer VM.
- Klonen, Konfigurieren und Übersetzen des Kerns.
- Lesen der Dokumentation zu `kvm/qemu`.
- Booten und Anhängen an einen laufenden Linuxkern mit `gdb`.
- Setzen und Auslösen eines Breakpoints, schrittweise Ausführung und Verändern von Werten mit `gdb`.
- Vorführen und Erklären dieser Schritte bzw. deren Ergebnis in der Rechnerübung.
- Optional: Finden einer minimalen lauffähige Konfiguration.

Hinweise:

- Mithilfe der `make`-Option `'O=<directory>'` könnt Ihr ein extra Ausgabeverzeichnis für den Build-Prozess angeben. Somit kann man mehrere Konfigurationen „nebeneinander“ bauen, wenn man das `'O=<directory>'`-Flag auch bei `make menuconfig` mit angibt.
- Bitte baut einen 64-Bit Kernel (Architektur: `amd64`).
- Bei der Bewertung achten wir insbesondere auf selbständige Arbeitsweise und Problemlösung sowie die sichere Beherrschung der grundlegenden Tätigkeiten durch beide Übungspartner.

1.5 Abgabe: am 28.10.2019