

Beispiellösung - Übungsblatt 10

Erstellt von Daniel Bub

Aufgabe 1.

Angenommen, Sie führen zwei Mal hintereinander das selbe Programm aus. Erstellen Sie somit zwei identische Prozesse? Erläutern Sie Ihre Antwort.

Lösung 1.

Zwei Prozesse können niemals identisch sein, selbst wenn ein Programm in kurzer Zeit zwei Mal hintereinander gestartet wird. Prozesse können sich in folgenden Punkten unterscheiden:

- Prozess-ID (PID)
- Programmparameter / Variablenwerte
- Befehlszeiger stehen auf verschiedene Abschnitte im Programmcode
- Ausführungsrechte
- verwenden nicht den exakt gleichen Speicherbereich

Aufgabe 2.

Erläutern Sie den Begriff *Userland-Prozess*?

Lösung 2.

Wie es der Name schon vermuten lässt, handelt es sich bei *Userland-Prozessen* um Prozesse, welche nicht im Kernel-Space, sondern im User-Space-Kontext ausgeführt werden. Nach Abbildung ?? zufolge

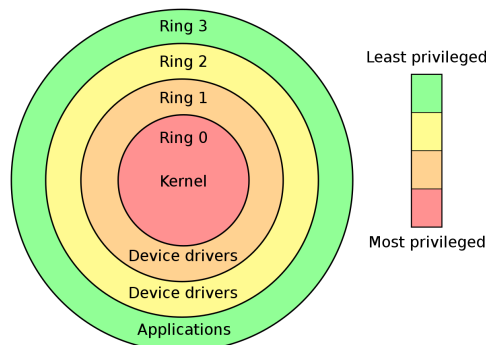


Abbildung 1: *Userland*: Schematische Darstellung als Ring-Modell.

laufen Userland-Prozesse in den Ringen 1-3. Eine derartige Trennung bringt den Vorteil, die Hardware vor unerlaubten Zugriffen von nicht-privilegierten Prozessen zu schützen.

Aufgabe 3.

Wie werden *Userland-Prozesse* unter **Linux**, **Windows** und **RIOT OS** erzeugt, bzw. beendet?

Lösung 3.

Linux: Unter Linux werden die Syscalls **fork()** / **exec*()** zum Erzeugen und die Befehle **exit()** / **kill** zum Beenden vom Prozessen verwendet.

Windows: Zur Prozess-Erzeugung unter Windows wird der Systembefehl **CreateProcess()**, zum Beenden die Funktion **ExitProcess()** verwendet. Prozesse können, analog zum **kill**-Befehl unter Linux, ebenfalls von außerhalb mittels **TerminateProcess()** beendet werden.

RIOT OS: Da RIOT OS-Betriebssysteme größtenteils auf dem POSIX-Standard beruht, sind auch hier die Linux-Systemaufrufe **fork()** zum Erzeugen und **exit()** zum Beenden von Prozessen implementiert.

Aufgabe 4.

Recherchieren Sie Informationen über das Betriebssystem **ULIX**. Nennen Sie mindestens zwei Unterschiede, hinsichtlich Prozesslimits, zwischen Linux und Ulix OS.

Lösung 4.

ULIX OS wurde an der Universität Erlangen-Nürnberg entwickelt. Ziel war es, ein komplett funktionierendes System zu erstellen, welches Schülern und Studenten ermöglichen soll, die Grundkonzepte heutiger Betriebssysteme (bspw. Scheduling & Paging) zu verstehen und zu veranschaulichen.

Prozesslimits - Unterschiede:

max. Anzahl paralleler Prozesse: Linux: *32.768*, Ulix: *1024*

max. Anzahl offener Dateien: Linux: ≥ 300.000 , Ulix: *16*

max. Pfadlänge: Linux: *4096 Zeichen*, Ulix: *256 Zeichen*

max. Anzahl Signalhandler: Linux: *beliebig viele*, Ulix: *32*

max. Anzahl Locks: Linux: *beliebig viele*, Ulix: *1024* = Anzahl an max. Prozessen

...