

Vorlesung Betriebssysteme WS 2014/2015

Aufgabenblatt 1 vom 13. November 2014

(Vorstellung der Lösungen bei den Tutoren bis zum 27.11.14)

Aufgabe 1.1: (15 Punkte)

Beantworten Sie folgende Fragen und erläutern Sie die Antworten Ihrem Tutor.

- Nennen und begründen Sie vier Ziele beim Entwurf eines Betriebssystems!
- Nennen Sie Beispiele für Betriebssystemfunktionalität die im Kernel- bzw. im User-Mode implementiert werden müssen!
- Was sind Vor- und Nachteile der Trennung von Kernel- und Usermode?

Aufgabe 1.2: (10 Punkte)

Erklären Sie Ihrem Tutor die Begriffe *Programm*, *Prozess*, *Thread* und *Task*! Worin bestehen die Unterschiede zwischen diesen Begriffen?

Aufgabe 1.3: (35 Punkte) -- Betriebssystem Dokumentation

Die folgenden Bilder zeigen den in der Vorlesung vorgestellten Aufbau des Windows- und des Linux-Kernels.

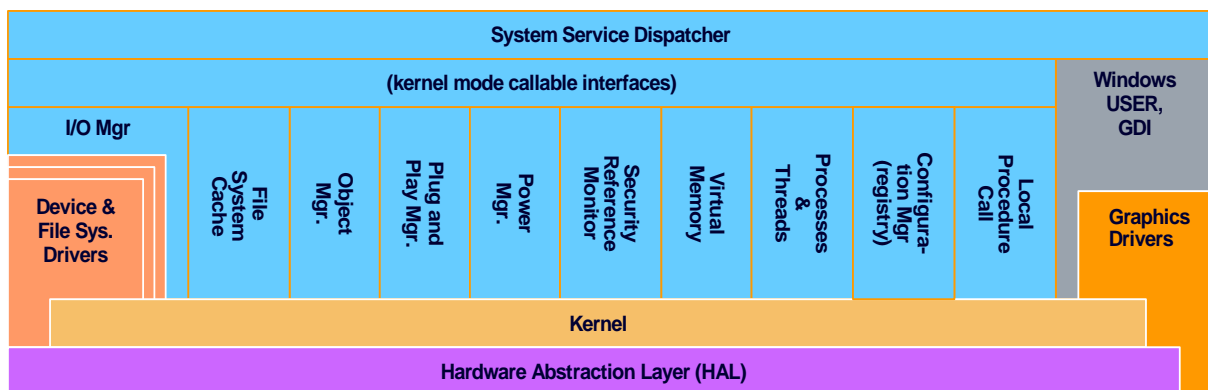


Abbildung 1: Windows-Kernel

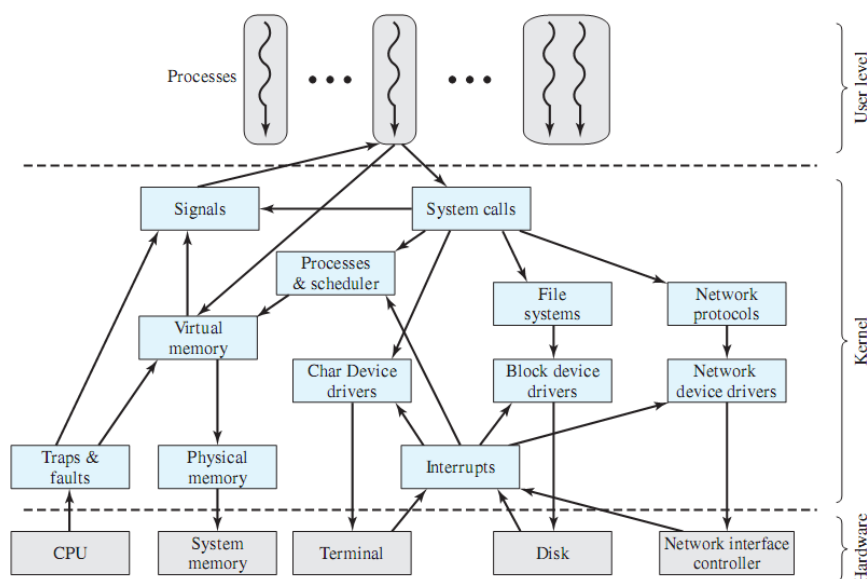


Abbildung 2: Linux-Kernel

Machen Sie sich mit der Dokumentation/Quellen des Windows Research Kernels (WRK) und des Linux-Kernels (3.2.0-amd64) vertraut und beantworten Sie die folgenden Fragen!

- Wie lässt sich in der Verzeichnisstruktur auf den Kernaufbau abbilden?
- Welche Teile des Windows-Kernels sind nicht im WRK enthalten?
- In welchen Dateien werden Scheduling Entscheidungen getroffen?
- Laden Sie sich die Quellen Linux (<https://www.kernel.org/pub/linux/kernel/v3.x/linux-3.2.tar.bz2>) / WRK <http://wrk.dcl.hpi.uni-potsdam.de/> herunter.

Hinweis: Die Dokumentation für den Windows Research Kernel ist innerhalb des HPI unter <http://wrk.dcl.hpi.uni-potsdam.de/> erreichbar. Der dokumentierte Linux Kernel ist z.B. unter <http://lxr.free-electrons.com/> oder <http://lxr.missinglinkelectronics.com/> zu finden.

Aufgabe 1.4: (10 Punkte)

Machen Sie sich mit der Betriebssystemgeschichte vertraut. Nennen Sie jeweils drei Merkmale (APIs, Kommandosprachen-Konstrukte, Systemprogramme) die Windows und Linux aus Vorgängersystemen übernommen haben.

Aufgabe 1.5: (30 Punkte)

Schreiben Sie ein C-Programm unter Windows und Linux auf der [InstantLab Platform](#), welches einen Systemruf ausführt um die Prozess-ID des Rufers zu ermitteln und prüfen Sie das Ergebnis mit der entsprechenden Usermode-API. Die Ausgabe soll folgendermaßen formatiert sein:

```
> blatt15X
syscall: 15679 usermode-API: 15679
```

X meint die Teilaufgabe a oder b.

- Unter Linux nutzen Sie den Systemruf `_NR_getpid` (`arch/x86/include/asm/unistd.h`) des für die Übung festgelegten Kernels (3.2.0-`amd64`) auf 86_64 Architektur.
- Unter Windows 2003 Server SP1 nutzen Sie `NtQueryInformationProcess` auf x86 Architektur. Die notwendigen Informationen können der WRK Dokumentation entnommen werden, sowie unter <http://j00ru.vexillum.org/ntapi/>. Einen Programmrahmen finden Sie auf der Vorlesungsseite.

Beantworten Sie folgende Fragen!

- Welche strukturellen Unterschiede gibt es in den Systemruffschnittstellen von Windows und Linux?
- Wo lassen sich die Informationen die über den Windows-Systemruf verfügbar sind unter Linux wiederfinden?

Demonstrieren Sie das Programm Ihrem Tutor auf der InstantLab-Plattform bs1.instantlab.org!

Verpacken Sie die Source-Code-Dateien (inklusive Makefile) in eine ZIP-Datei (blatt15a.zip, blatt15b.zip). Diese ZIP-Datei muss mindestens 24 Stunden vor dem Tutoriumstermin in das Abgabesystem eingestellt werden.