

Vorlesung

Ausgewählte Kapitel der praktischen Betriebsprogrammierung I

Wintersemester 1998/99 (10306)

Ausgewählte Kapitel der praktischen Betriebsprogrammierung I (WS 1998/99)

Inhalt

A	Organisatorisches	A.1
A.1	Vorlesung	A.1
A.2	Übungen	A.5
B	Überblick	B.1

C	Betriebssystemarchitekturen	C.1
C.1	Entwicklung der Betriebssystemstrukturierung	C.1
C.2	Monolithische Betriebssystemkerne	C.2
	1 Vorteile	C.2
	2 Nachteile	C.3
	3 Struktur	C.4
C.3	Minimalkerne	C.5
	1 Vorteile	C.5
	2 Nachteile	C.6
	3 Struktur	C.7
C.4	Objektbasierte Systeme	C.8
	1 Sandboxing	C.9
	2 Vorteile	C.10
C.5	Betriebssysteme für Embedded Systems	C.11
C.6	Anforderungen an zukünftige Betriebssysteme	C.12

D	UNIX-Überblick	D.1
D.1	Historische Entwicklung.....	D.2
D.2	Grundkonzepte	D.4
	1 Dateien und Dateisystem	D.4
	2 Programme und Prozesse.....	D.7
	3 Prozesskommunikation	D.8
D.3	UNIX Systemstruktur	D.10
	1 Benutzermodus	D.11
	2 Systemkern.....	D.11
	3 Wechsel zwischen Benutzermodus und Systemkern	D.13
D.4	Struktur des Systemkerns	D.14
D.5	Dateisystem	D.15
D.6	Prozessverwaltung	D.19
	1 Prozess.....	D.19
	2 Datenstrukturen zur Ausführung eines Programms	D.21
	3 Datenstrukturen zur Abwicklung eines Prozesses	D.21
	4 Multiprogrammierung, Scheduling	D.22
	5 Prozesszustände	D.23
D.7	Treiber	D.24
	1 raw- oder character-device driver	D.24
	2 block-device driver.....	D.24

F.4	UNIX - Erzeugen eines neuen Prozesses.....	F.22
	1 Überblick	F.22
F.5	UNIX — Ablauf	F.24
	1 Überblick	F.25
	2 Ablauf (Beispiel aus XELOS).....	F.26
	3 Ablauf (2).....	F.27
	4 Wechsel zwischen zwei Prozessen	F.30
F.6	5 Erzeugen eines neuen Prozesses	F.31
F.7	UNIX — Ausführen eines Programms	F.32
	UNIX — Terminieren von Prozessen.....	F.34
	1 Überblick	F.34
	2 Ablauf von exit im Kern	F.35
	3 Systemaufruf wait().....	F.37
F.8	4 typische wait() -Anwendung	F.39
	MACH — Tasks und Threads	F.41
	1 Motivation	F.41
	2 Vergleich von Prozeß- und Thread-Konzepten	F.42
	3 Abstraktionen in MACH: Tasks und Threads	F.48
F.9	UNIX — Prozesse, LWPs & Threads.....	F.49
	1 pthread-Benutzerschnittstelle	F.50
	2 Threads & UNIX-Semantik	F.53

E	MACH	E.1
E.1	Überblick	E.1
	1 Motivation	E.1
E.2	2 wesentliche Eigenschaften	E.3
E.3	Architektur	E.4
	Basis-Abstraktionen des MACH-Kerns	E.5
	1 Task.....	E.5
	2 Thread.....	E.6
	3 Memory Object	E.7
	4 Port	E.8
E.4	5 Message	E.8
	Systemschnittstelle	E.9
F	Prozesse	F.1
F.1	Überblick	F.1
F.2	UNIX — Prozeßbild und Speicherorganisation	F.4
	1 Speicherorganisation eines Programms	F.4
	2 Speicherorganisation eines Prozesses	F.5
	3 Stackaufbau eines Prozesses	F.6
F.3	UNIX — Verwaltungsstrukturen des Systemkerns	F.12
	1 Verwaltungsdaten pro Prozeß.....	F.12
	2 Globale Prozeßverwaltungsdaten	F.14
	3 Kommando ps	F.21

F.10	Koordinierungsmechanismen	F.57
	1 UNIX — Semaphore	F.57
	2 Pthreads-Koordinierung	F.64
	3 Thread-Cancellation	F.72
F.11	4 Koordinierung im UNIX-Kern (sleep/wakeup).....	F.73
	UNIX — Signale.....	F.79
	1 Ablauf der Signalverarbeitung im Systemkern	F.80
F.12	2 System V ↔ BSD/POSIX-Semantik	F.82
	UNIX — Scheduling.....	F.83
	1 Scheduling in UNIX System V	F.85
	2 Scheduling in BSD UNIX.....	F.87
F.13	3 Vergleich: System V · BSD.....	F.90
	MACH — Scheduling	F.96
	1 Architektur	F.96
	2 Schedulingstrategie & Prioritäten	F.97
	3 Scheduling-Unterstützung für nebenläufige Anwendungen	F.99

G	Interprozesskommunikation	G.1
G.1	Überblick	G.1
G.2	Sockets	G.2
	1 Communication Domain und Protokoll	G.3
	2 Socket Typen	G.3
	3 Client-Server Modell	G.5
	4 Generieren eines Sockets	G.7
	5 Namensgebung	G.8
	6 Verbindungsanforderungen annehmen	G.8
	7 Verbindungsanbau und Kommunikation (Stream-Sockets)	G.9
	8 Stream-Sockets — Systemschnittstelle	G.11
	9 Verbindungslose Sockets	G.12
	10 weitere Systemaufrufe der Socketschnittstelle	G.12

I	Sicherheit in heutigen Betriebssystemen	I.1
I.1	Überblick	I.1
I.2	UNIX-Sicherheitsmechanismen	I.2
I.3	Angriffe auf UNIX-Systeme	I.4
	1 Angriffspunkte	I.4
	2 Vorgehensweise	I.5
	3 Gegenmaßnahmen	I.6
	4 Risikoabschätzung	I.7
I.4	Netzdienste unter UNIX	I.8
I.5	Verschlüsselungsverfahren	I.9
	1 Anwendung von Verschlüsselung	I.10
	2 IP Security	I.11
I.6	Firewalls	I.12
J	UNIX-Dateisystem	J.1
J.1	Funktionalität	J.1
J.2	Directories (Kataloge)	J.2
J.3	Inodes	J.2
	1 Umwandlung: Pfad : Inode	J.3
	2 Datei öffnen	J.5
	3 Kontrollstrukturen und Systemaufruf dup()	J.10
	4 Kontrollstrukturen und Systemaufruf fork()	J.11
J.4	Vnodes	J.12
	1 Motivation	J.12
	2 Kontrollstrukturen (SystemV.4)	J.14

H	Rechnerkommunikation (TCP/IP)	H.1
H.1	Überblick	H.1
H.2	Einordnung in das ISO/OSI Referenzmodell	H.1
H.3	Internet Protocol - IP	H.4
	1 Adressierung	H.5
	2 Routing	H.6
H.4	User Datagram Protocol - UDP	H.9
	1 Adressierung	H.9
	2 Motivation	H.16
H.5	Transmission Control Protocol - TCP	H.17
H.6	Systemschnittstelle	H.19
	1 Adressierung	H.20
	2 Rechnernamen / IP-Adressen	H.20
	3 weitere Verwaltungsdaten	H.21
H.7	spezielle Funktionalität	H.23
		H.24

J.5	Dateisysteme	J.15
	1 SystemV-Dateisystem	J.15
	2 Berkeley Fast File System (UNIX File System — UFS)	J.17
J.6	Virtuelles Dateisystem	J.19
	1 Überblick	J.19
J.7	Mount-Mechanismus	J.20
	Network File System (NFS)	J.24
	1 Überblick	J.24
	2 Beispiel	J.27
	3 Sicherheitsaspekte	J.28
J.8	Block-Puffer Cache	J.29
	1 Implementierung	J.31
	2 Verwaltungsstrategien	J.31
	3 Integration von Buffer Cache und Speicherverwaltung	J.32
J.9	Kontrollstrukturen	J.33
	Dateisystem - Buffer Cache	J.35
J.10	Treiber	J.35
	1 Kontrollfluß	J.35
	Treiber	J.36
	1 Überblick	J.36
	2 Treiber-Funktionsvektoren	J.38
J.11	Hintergrundspeicher	J.42
	1 Organisation einer Disk - Laufwerk	J.42
	2 Organisation einer Disk (SCSI)	J.45
	3 Organisation einer Disk — Betriebssystem	J.45
J.12	Disk-Zugriffsstrategien	J.48
	4 Disk-Treiber: Datenstrukturen und Kontrollfluß	J.48