

BETRIEBSSYSTEME VO

Schriftlich, 90 Minuten, keine Unterlagen

Bsp Tests auf LVA-HP

Vorlesung Folien

Semaphoren Bsp, Deadlock Bsp, Theorie

Prozesse: Trace, Zustände (5+2), Swapping, Kontrollstrukturen
Process Image (PCB), Execution Modes
Non-Process Kernel, BS in User-Processen, Prozessbasiertes BS
Monolithische OS, Layered OS, Mikrokernel-Implementierung

Threads: Einheit für Dispatching (gemeinsamer PCB, UAS), Zustände (3)
User-Level Threads, Kernel-Level Threads

Synchronisation: Mutual Exclusion, Condition Synchronisation

Software-Lsg: Dekker Algorithmus, Petersen Algorithmus, Bakery Algorithmus

Hardware-Lsg: Interrupt Handling, Hardware Synchronisation (Test and Set, Exchange)

Spez. Konstrukte: Semaphore, Spinlocks (kein Prozess Switch)

Eventcounts, Sequenzen

Monitors (Prozeduren, Daten, Include): Condition Variablen

Message Passing

Deadlocks: Bedingungen (4), Prevention / Avoidance / Detection $\rightarrow$ Recovery
Bankers Algorithmus (Avoidance)
Claim- und Allocation Matrizen

Speicherverwaltung: Partitionierung, Relocation, Protection, Sharing, Performance

Fixed Partitioning Buddy System / $\backslash$ Page Fault
Physikalische Adressen nicht ausgelagert
Protection Key

Segmentierung, Paging $\rightarrow$ Virtual Memory, TLB, IPT, Resident Set
Demand Paging, Prepaging, Replacement Policy: LRU, FIFO, Clock
Quick Res. St. Working Set Strategy

Scheduling: Long-term, Mid-term, Short-term

Kriterien

Ein-/Auslageren

nächster Prozess

Short Term: Dispatchen: System Calls, Traps, IO Interrupt, Signal, User Interrupt
Kriterien

Selektion funktion, Decision mode (prioritätenlos)

Strategien: FCFS, Round Robin (virtual), SPN, SRT, HRRN

Feedback, Real-Time: EDF 18 Strategien /

Ein-/Ausgabe: Arten: Mensch-Interface, Maschine-Interface, Kommunikation
Programmiert IO, Interrupt-driven IO, DMA (IO Channel)
Schichten: Logical IO, Device IO, Scheduling and control
Puffer, Scheduling (minimizing Seek Time), Disk cache
Effizienz $\leftrightarrow$ Flexibilität

File-Management: File, Sequential File, Indexed Sequential File, Indexed File, Hash File
Types: Regular Files, Directory, Other Special Files, Block Special Files
Mountables, Names, Operations

File System: Disk $\rightarrow$ Partitionen, MBR

Index: Blöcke $\rightarrow$ Contiguous; Chain; Indexed Allocation, D-Nodes

Sequential Files: Fixed Blocking, Variable-length spanned / unspanned
Block

Freie Blöcke: Chain Free Portions, Bit Tables, Indexing

Performance: Disk Caching, Block Read Ahead

Security: Confidentiality, Integrity, Availability (CIA)
Concerns: Authenticity, Accountability
Passive / Active Threats

C: Exposure / Interception

I: Modification

A: DoS

Malware: Virus, Worm, Trojan Horse, Logic Bomb,
Trap Door, Root Saver, DoS

Protection: Einschränkung: Zugriff, Verwendung
Protection Domains (Prozess in Domain)
 $\rightarrow$ Access Matrix
 $\rightarrow$ Access Control List
 $\rightarrow$ Capability List
 $\rightarrow$ Lock Key System

Bell and LaPadula's Model:

Security Classifications für Objekte und Subjekte

$\rightarrow$ Operationen von Subjekten auf Objekten

read: $SC(S) \geq SC(O)$

append: $SC(S) \leq SC(O)$

read + write: $SC(S) = SC(O)$

Kryptographie: Authentifizierung, Verschlüsselung

Intrusion Detection: Threshold, Anomaly, Audit records

Prozesszustände: New $\rightarrow$ Ready $\leftrightarrow$ Running $\rightarrow$ Exit

$\rightarrow$ Ready Queue
 $\rightarrow$ Event x Queue

Blocked

Unterhalten: Supervisor Call
 Trap
 Interrupt

+ Ready Suspended, Blocked Suspended

Process Image: - Process Control Block - process identification
 - System State - process state information
 - User Program - process control information
 - User Data
 Scheduling IPC MM Resources

BS-Tabellen: Memory Table, IO Table, File Table, Process Table

Synchronisation: RT-Anforderungen: Mutual Exclusion, Progress, Bounded Waiting

Software Alg: Dekker, Peterson, Bakery-Algorithmus

Hardware Alg: Interrupt Disabling, Test and Set, exchange

Synchr-kontrolle: - Semaphore: $\text{init}(S, 1)$, $P(S)$, $V(S)$
 - Spinlocks: Busy waiting
 - Eventcounts: $\text{advance}(E)$, $\text{await}(E, val)$
 - Sequenzen: $\text{label}(S)$
 - Monitor: Prozeduren, lokale Daten,
 Condition Variable: $\text{await}(c)$, $\text{signal}(c)$
 - Messages: Ereignismeldung (Queue), Zustandsmeldung

Deadlock: Voraussetzungen: Hold and wait, No preemption, Mutual Exclusion
 + Circular wait

Preemption: Indirect: Hold and Wait, Preemption
 Direct: Circular wait: Ordnung für Ressourcen

Abstraktion: Initiation Demand
 Allocation Demand: Bankers Algorithmus

Vektoren: Resource: R
 Available: V

Matrizen: Claim: C
 Allocation: A
 Request: N
 Nächste Anforderung: Q

Speicherverwaltung: Externe / Interne Fragmentierung
→ Free / Dynamische Partitionierung → Buddy System

Relocation: Von Prozessen belegter Speicher ist nicht fixiert
↳ logische Adressen (überwinkelter Code)

Segmentierung: Segmente variabler Länge

Paging: Pages gleicher Größe, Frames in HS

Virtual Memory: Dynamische Adressübersetzung
Aufspaltung Prozesse in Pages
TLB, Multilevel Page Table (2 Page Index)
Shared Page Table

Replacement Policy: Least Recently used
First In First Out
Clock Policy

Resident Set: Fixed Allocation, Variable Allocation
↳ Working Set Strategy

Protection: Zugriff nur auf Adressen in Page Table
Prozess hat Key, Vergleich mit Frame Key
Jeder TLB Eintrag hat Key, Vergleich mit TLB Key

Scheduling: User Oriented: Performance Response Time, Deadlines
Other Predictability
System Oriented: Performance Throughput
Other Fairness, Resource Balance

Scheduling Function:

First Come First Serve	:	NP
Round Robin		P
Virtual Round Robin	(10 Queue)	P
Shortest Process Next		NP S
Shortest Remaining Time		P S
Highest Response Ratio		NP
→ max $RR = (w + s) / s$		
Feedback		P S
→ Verringerte Ausführungszeit		
Earliest Deadline First		P
→ minimiert Deadline misses		

Input/Output: Schichtenmodell:

- Logical IO: open, close, read
- Device IO: Übersetzung in Gerätekommandos + Pufferung
- Scheduling + Control: Verwaltung von IO Operationen, Queues

Blocking: Prozess in Blocked Queue < wartet auf IO

Non-Blocking: Return Status liefert Feedback

Synchron: Prozess wartet auf IO

Asynchron: Prozess wird über Beendigung informiert

Puffer: Zwischenspeicher für IO Daten

→ Entkopplung von Prozess IO und OS IO

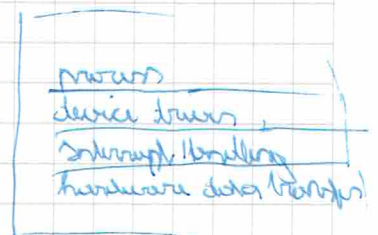
→ Swapping während IO möglich

Disk IO: $T_S + T_{RD} + T_{TF} = T_A$

totale Zeit $\frac{1}{10}$ pro Sek

Scheduling: Ordnung von IO Requests

→ Priority, FIFO, C-Scan



Disk Cache: Im Hauptspeicher

→ Nachschreibstrategie: Least Recently Used, Least Frequently used

Security: Ziele: Confidentiality, Integrity, Availability

Threats: Exposure, Modification, DoS

Prinzipien: Open Design

Least Privilege

Complete Mediation

Default Einstellungen: Keine Zugriffe

Einfache Implementierungen auf niedriger Ebene

Access Matrix: Domain (Subjekte), Objekte, Rechte

→ ACL: Zugriffsrechte bei Objekten

→ Capability list: Zugriffsrechte bei Subjekten (Tickets)

Bell and LaPadula's Modell

→ SC (security classifications) für Subjekt und Objekt

→ read $SC(S) \geq SC(O)$

→ append $SC(S) \leq SC(O)$

→ read+write $SC(S) = SC(O)$

Intrusion Detection: Anomalien in Logs

File Management: Datei Organisation: unstructured file
sequential file
indexed sequential file
indexed file
direct (hashed) file

Naming: Anzahl Zeichen
Groß- & Kleinschreibung
File Name Extensions

File System: Master Boot Record, Partition Table, Partitions
→ Records

Datei Implementierung: Datei → Sammlung von Blöcken

Block Allokierung: Contiguous Allocation
Chained Allocation: Zeiger auf nächsten Block
Indexed Allocation: Zeiger in FAT
File Allocation Table
I-Node: Datenstruktur für jedes File

I-Node: Enthält File-Metadaten und Referenzen auf Blöcke des Files.
Größe Files → Indirekte Nodes (bezeichnete Block-Referenzen).

Sequential Files:

Block Verwaltung: Fixed Blocking: Records fester Länge
Variable Length Spaced Blocks: Records variabel und können auf Blöcke verteilt sein
Variable Length Unspaced Blocking

Disk Allocation Table: Markierung freier Blöcke
→ Chained Free Portions
→ Bit Table
→ Indexing

Strukturierung

Performance: Disk Caching, Block Read Ahead, Read Strategie

Unix partitionen: boot block, super block, free map, i-nodes, root dir, files