

1. Wie wird ein Betriebssystem definiert? (siehe Satz: Theorie-1, Folie-8)

Programme, welche zusammen mit der Hardware die Basis des Rechensystems bilden und die Abwicklung von Prozessen steuern und überwachen, sie verwalten auch die Rechnerressourcen.

2 Punkte

2. Was versteht man unter Stapelverarbeitung?

Eine als Eingabe bereitgestellte Menge an Aufgaben wird vollständig, automatisch und (meist) sequentiell verarbeitet.

2 Punkte

3. Was versteht man unter einem Prozess? (siehe Satz: Theorie-1, Folie-31)

- Programm das gerade ausgeführt wird (Instanz eines Programm)
- hat einen Eintrag in der Prozesstabelle
- ihm werden Ressourcen zugeordnet (Hauptspeicher etc)

2 Punkte

4. Wozu dient die Prozesstabelle? (siehe Satz: Praxis-1, Folie-28)

- dient zur Implementierung des Prozessmodells und besitzt pro Prozess einen Eintrag mit der Registerbelegung und weiteren Elementen, die zur Wiederaufnahme der Prozesse benötigt werden.

2 Punkte

5. Nennen Sie zwei Ereignisse, welche zur Erzeugung von Prozessen führen. (siehe Satz: Theorie-2, Folie-5)

Folien:

1. Initialisierung des Systems
2. Ausführung eines Systemaufrufs zur Prozesserzeugung durch einen laufenden Prozess
fork()
3. Nutzeranforderung zur Erzeugung eines neuen Prozesses
4. Einleitung eines Batch Jobs

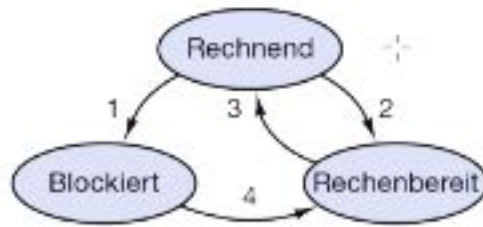
Durch ein fork() wird ein neuer Prozess aus einem alten erstellt (hat noch die selben Eigenschaften).

Bei Systemstart von Linux wird ein Startprozess init erstellt, der wiederum alle anderen Prozesse erzeugt.

2 Punkte

6. In welchen drei Zuständen kann sich ein Prozess befinden? (siehe Satz: Theorie-2, Folie 8)

- laufend/rechnend (running),
- angehalten/rechenbereit (ready),
- blockiert (blocked),
- *[gestoppt (terminated)]*



1. Prozess blockiert, weil er auf Eingabe wartet
2. Scheduler wählt einen anderen Prozess aus
3. Scheduler wählt diesen Prozess aus
4. Eingabe vorhanden

3 Punkte

7. Was versteht man unter einem Thread?

- Einen Ausführungsfaden eines Prozesses, welcher mit anderen Fäden den gleichen Adressraum benutzt
- Leichtgewichtiger und schneller zu erzeugen als ein Prozess
- Eigener Threadkontext mit Instruction Pointer und Stack
- Eigene Threadtabelle (bei Kern oder Benutzer)

2 Punkte

8. Threads können im Benutzeradressraum oder im Kern des Betriebssystems implementiert werden. Beschreiben Sie die wesentlichen Unterschiede beider Implementierungen!

im Benutzeradressraum:

- laufen auf der Basis eines Laufzeitsystems, welches aus einer Sammlung von Prozeduren besteht, die die Threads verwalten
- jeder Prozess braucht seine eigene private Thread-Tabelle
- → Threads im Benutzeradressraum sind besser skalierbar
- Betriebssystem muss keine Threads unterstützen, daher aber auch kein Einfluss

im Kern des Betriebssystems:

- der Kern hat eine Thread-Tabelle, die alle Threads im System verwaltet
- Kern übernimmt die Erzeugung oder Zerstörung von Threads
- → Threads im Kern erzeugen verhältnismäßig hohe Kosten, welche durch das Recyclen von Threads gelindert werden können

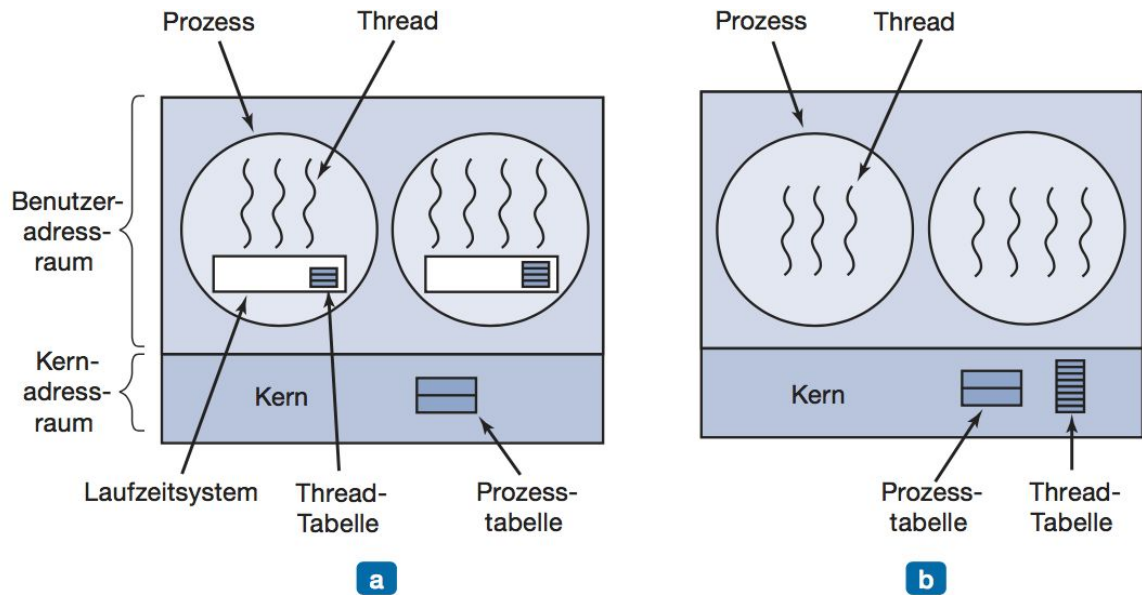


Abbildung 2.16: (a) Ein Thread-Paket auf Benutzerebene. (b) Ein Thread-Paket, das vom Kern verwaltet wird.

4 Punkte

9. Nennen Sie vier Scheduling-Algorithmen für interaktive Systeme! (siehe Satz: Theorie-2, Folie-58)

Priority Scheduling, Multiple Queues, Lottery Scheduling, Shortest Process next, Guaranteed Scheduling, Fair-Share Scheduling, Round Robin

4 Punkte

10. Erläutern Sie kurz die beiden Formen: Preemptiv / Non Preemptiv Scheduling!

Preemptives (unterbrechendes) Scheduling: Das Betriebssystem kann einem Prozess zu nahezu jeder Zeit die CPU entziehen und einen anderen Prozess bearbeiten.

Nicht-Preemptives (nicht unterbrechendes) Scheduling: Jeder Prozess im Kernspace wird solange ausgeführt bis er blockiert oder die CPU wieder freigibt.

2 Punkte

11. Welche Bedingung muss das Scheduling in Echtzeitsystemen erfüllen? (siehe Satz: Theorie-2, Folie-61)

1. Zeit spielt eine wesentliche Rolle

2. Kategorien

- (Harte) Echtzeit
- „Soft real time“ – ist laut Def. keine Echtzeit!
- Periodisch oder Aperiodisch

3. Scheduling muss erfüllen: $\sum_{i=1}^m \frac{C_i}{P_i} \leq 1$

P_i : Intervall des Auftretens

C_i : Benötigte CPU Zeit

i : Ereignis/Task

* ... (bezieht sich auf 2. zweiter Anstrich) Echtzeitsysteme sind „Systeme zur unmittelbaren Steuerung und Abwicklung von Prozessen“, die quantitative Echtzeitanforderungen erfüllen müssen. Eine häufige Anforderung ist, dass ein Ergebnis innerhalb eines vorher fest definierten Zeitintervalls garantiert berechnet ist. Die Größe des Zeitintervalls spielt dabei keine Rolle.

Wenn es m periodische Ereignisse gibt und das Ereignis i tritt mit der Periode P_i ein und benötigt C_i Sekunden an CPU-Zeit, um jedes Ereignis zu behandeln, dann kann die gesamte Belastung nur bearbeitet werden, wenn gilt:

$$\sum_{i=1}^m \frac{C_i}{P_i} \leq 1$$

4 Punkte

12. Wozu dienen Basis- und Limitregister bei der Speicherverwaltung? (siehe Satz: Theorie-3, Folie-7)

Folien:

Basis- und Limitregister können benutzt werden, um jedem Prozess einen eigenen Adressraum zu geben

Nicht in den Folien:

Ein Basisregister zeigt auf den Beginn eines Adressraums eines Prozesses, im Limitregister steht wie lang dieser Raum ist. Dies dient dazu Adressräume unterschiedlicher Prozesse zu trennen, sonst könnten alle Prozesse überall hin schreiben, was im schlimmsten Fall das BS abstürzen lässt.

Andere Formulierung aus dem Buch:

Sobald ein Prozess läuft, wird das Basisregister mit der physischen Adresse geladen, an der das Programm im Speicher anfängt, und das Limitregister wird mit der Länge des Programms geladen.

2 Punkte

13. Was versteht man unter virtuellem Speicher? (siehe Satz: Theorie-3, Folie-14)

- Jedes Programm hat einen eigenen Adressraum, welcher in Seiten aufgeteilt ist.
- möchte man Programme ausführen, welche nicht in den Hauptspeicher passen, werden Teile der Informationen auf der Festplatte gespeichert

Andere Formulierung aus dem Buch:

Grundidee dahinter ist, dass der Adressraum eines Programms in Einheiten aufgebrochen wird, die sogenannten Seiten (page). Jede Seite ist ein aneinander angrenzender Bereich von Adressen. Die Seiten werden dem physischen Speicherbereich zugeordnet, wobei nicht alle Seiten zur selben Zeit im physischen Speicher vorhanden sein müssen, damit das Programm läuft. Wenn das Programm auf einen Teil des Adressraums zugreift, der sich aktuell im physischen Speicher befindet, dann kann die Hardware die notwendige Zuordnung sehr schnell durchführen. Wenn das Programm dagegen auf einen Teil des Adressraums zugreifen will, der nicht im physischen Speicher ist, so wird das Betriebssystem alarmiert, das fehlende Stück zu beschaffen und den fehlgeschlagenen Befehl noch einmal auszuführen.

2 Punkte

14. In den heutigen Betriebssystemen wird hauptsächlich mit virtuellem Speicher gearbeitet.

14.a Was ist ein Seitenfehler?

- wenn ein Programm auf einen Speicherbereich zugreifen möchte, welcher sich weder im Hauptspeicher befindet noch einen Eintrag in der MMU hat.

14.b Welche Schritte muss das Betriebssystem bei der Behebung eines Seitenfehlers durchführen? (siehe Satz: Theorie-3, Folien-44-46)

- Schauen ob genügend freie Seiten vorhanden sind, wenn nicht müssen welche ausgelagert werden.
- Sicherung der Register und wichtigen Informationen -> Sprung in den Kernel
- Suche nach virtueller Seite
- Laden der benötigten Seite von der Festplatte
- Seitentabelle aktualisieren
- Zurückladen der Register

Aus dem Skript:

1. Die Hardware speichert den Programmzähler auf dem Stack und löst einen Sprung in den Kernel aus.
2. Assembler-Routine speichert allgemeine Register und andere flüchtige Informationen
3. System entdeckt Seitenfehler und versucht herauszufinden, welche virtuelle Seite benötigt wird
4. Sobald der durch die virtuelle Adresse verursachte Fehler bekannt ist, prüft das System, ob die Adresse gültig und der Zugriffsschutz konsistent ist.
5. Wenn der Frame als "dirty" markiert ist, wird die Seite für die Übertragung auf die Festplatte geplant, der Kontextwechsel findet statt und der fehlerhafte Prozess wird ausgesetzt.
6. Sobald der Frame sauber ist, sucht das Betriebssystem die Festplattenadresse, wo die benötigte Seite ist, und plant die Festplattenoperation, um sie zu laden.
7. Wenn der Festplatten-Interrupt anzeigt, dass die Seite angekommen ist, werden die Tabellen aktualisiert, um die Position zu reflektieren, und der Frame wird als normal markiert.
8. Sicherung der Fehlinstruktion in dem Zustand, wo es begann, und zurücksetzen des Programmzählers
9. Der Fehlerprozess ist geplant, das Betriebssystem kehrt zur Routine zurück, die es aufgerufen hat.
10. Routine lädt Register und andere Zustandsinformationen neu und kehrt zum Benutzerraum zurück, um die Ausführung fortzusetzen

5 Punkte

15. Nennen Sie vier Algorithmen für die Seitenersetzung! (siehe Satz: Theorie-3, Folie-23)

- FIFO (First in First Out)
- NRU (Not recently used)
- LRU (Least recently used)
- Clock
- Second-Chance
- Optimal
- Working set
- WSCLOCK

4 Punkte

16. Nennen Sie 6 Dateioperationen! (siehe Satz: Theorie-4, Folie-12)

- copy
- move
- rename
- delete
- create
- close

- *read*
- *write*
- *append*
- *seek*
- *get attributes*
- *set attributes*

3 Punkte

17. Skizzieren Sie ein hierarchische Verzeichnissystem mit Verzeichnissen und Dateien (mindestens 2 Ebenen) (siehe Satz: Theorie-4, Folie-17)

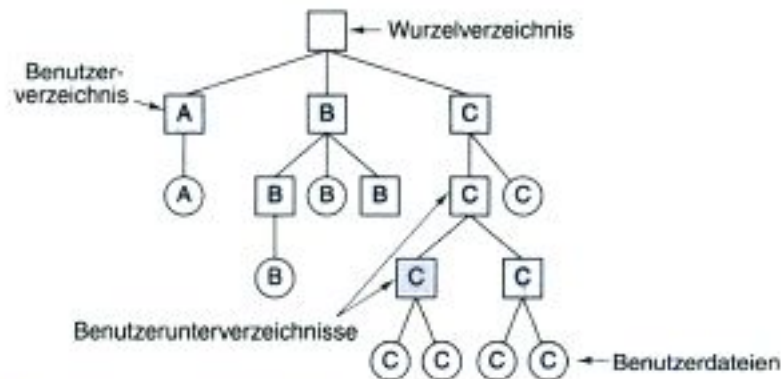


Abbildung 4.7: Ein hierarchisches Verzeichnissystem

4 Punkte

18. Beschreiben Sie die Funktionsweise eines Dateisystems mit Datei-Allokationsliste (FAT).

Die FAT ist eine Tabelle in welcher vermerkt ist, welche Blöcke belegt und frei sind. Außerdem steht der erste Block einer Datei dort. Diese bildet eine verkettete Liste, da in jedem neuen Block steht, wo der nächste Block zu finden ist. Dies führt zu einer starken externen Fragmentierung, da es keine komplette Liste der Blöcke der Datei gibt und diese über die Festplatte verteilt sein können. Weiterhin ist das Lesen mit einer Festplatte schwierig, denn wenn die Blöcke hintereinander liegen, und die Festplatte noch den ersten Block auswertet, muss man eine Umdrehung warten um den nächsten Block der Datei lesen zu können.

Im Buch steht noch:

Der Hauptnachteil dieser Methode ist, dass sich die gesamte Tabelle die ganze Zeit über im Speicher befinden muss, um zu funktionieren.

5 Punkte

19. Was versteht man unter einem Journaling-Dateisystem? Welche besondere Eigenschaft müssen alle Aktionen im Journal erfüllen?

- Grundidee: es wird ein Log darüber geführt, welche Aktionen das Dateisystem plant. Falls es dann zu einem Systemabsturz kommt, bevor diese geplante Arbeit erledigt werden konnte, kann das System bei einem Neustart im Log nachschauen, was zum Zeitpunkt des Absturzes vorging, sodass das Dateisystem seinen Auftrag beenden kann
- Änderungen werden also vor dem eigentlichen Schreiben in einem dafür reservierten Speicherbereich, dem Journal, aufgezeichnet
- Damit ist es möglich, einen konsistenten Zustand der Daten zu rekonstruieren, auch wenn ein Schreibvorgang an beliebiger Stelle abgebrochen (z.B. Abstürze, Stromausfall) wurde
- Änderungen werden ins Journal geschrieben -> Änderungen werden auf die Festplatte geschrieben -> Eintrag im Journal wird gelöscht (damit es nicht wie bei einem Log die Festplatte zumüllt)
- Besondere Eigenschaft:
 - Damit Journaling korrekt funktioniert, müssen die Operationen im Log idempotent sein, sodass sie so oft wie nötig wiederholt werden können, ohne Schaden anzurichten

5 Punkte

20. Nennen Sie die beiden Arten von Ein-/Ausgabegeräten! Worin unterscheiden sich beide voneinander? (siehe Satz: Theorie-5, Folie-4)

Blockorientierte Geräte (Festplatte, Diskette, CD, ...):

Speichert Informationen in Blöcken fester Größe, Transfers erfolgen in Einheiten von ganzen Blöcken

Zeichenorientierte Geräte (Serielle/Parallele Schnittstelle, Pseudoterminals[ttyXX]):

Liefert oder akzeptiert einen Zeichenstrom, ohne Berücksichtigung der Blockstruktur, nicht adressierbar

4 Punkte

21. DMA

a. Wofür steht die Abkürzung DMA?

Direct Memory Access (Speicherdirektzugriff)

b. Erläutern Sie eine Ein-/Ausgabe mit DMA!

Peripheriegeräte können ohne Umweg über die CPU mit dem Hauptspeicher arbeiten.

Wenn man für jedes Zeichen bei der Ein-/Ausgabe von z. B. einem Drucker ein Interrupt ausführen würde, würde dies ein sehr zeitintensiver Prozess sein. Stattdessen schreibt der DMA-Controller (man braucht spezielle Hardware dafür) die Zeichen nacheinander in ein

Register, wenn dieses voll ist, wird ein Interrupt ausgelöst und die Daten gesendet/empfangen. So kann man die Anzahl der Interrupts von einmal pro Zeichen zu einmal pro Puffer reduzieren.

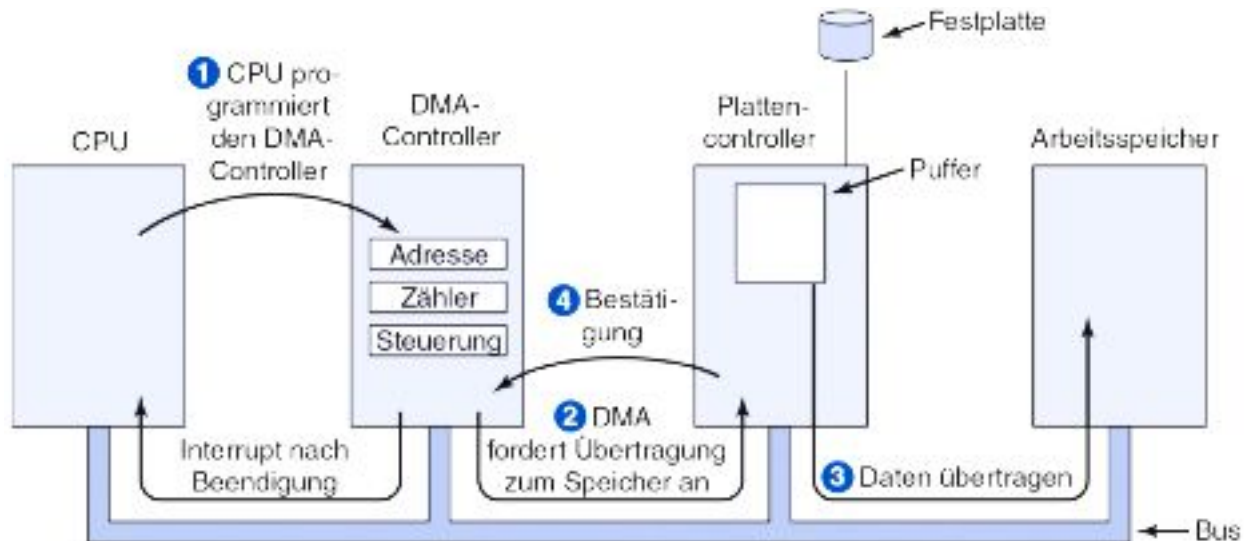


Abbildung 5.4: Ablauf eines DMA-Transfers.

- **1. Schritt:**
 - Zuerst programmiert der Prozessor die Register des DMA-Controllers, damit er weiß, was er wohin transportieren soll
 - Zusätzlich wird ein Kommando an den Plattencontroller ausgegeben, damit dieser die Daten von der Platte in seinen internen Speicher einliest und die Prüfsumme testet. Sobald gültige Daten im Speicher des Plattencontrollers vorliegen, kann die Übertragung per DMA beginnen.
- **2. Schritt:**
 - Der DMA-Controller leitet den Datentransfer mit einem Lesebefehl über den Bus zum Plattencontroller ein
- **3. Schritt:**
 - Das Schreiben in den Speicher ist wie ein Standard-Buszyklus
- **4. Schritt:**
 - Sobald das Schreiben beendet ist, schickt der Plattencontroller ein Signal zur Bestätigung über den Bus zum DMA-Controller
 - Der DMA-Controller erhöht daraufhin die Speicheradresse und verringert die Anzahl der noch zu übertragenden Zeichen. Solange diese Anzahl größer als null ist, werden die Schritte 2 bis 4 wiederholt. Ist die Anzahl null, dann erzeugt der DMA-Controller einen Interrupt und teilt der CPU mit, dass die Übertragung abgeschlossen ist

(aus dem Tanenbaum - zur Erläuterung der Skizze)

6 Punkte

22. Was versteht man unter einem Gerätetreiber? (siehe Satz: Theorie-5, Folie-22; Praxis-1, Folie-23)

Meist vom Gerätehersteller für ein Betriebssystem bereitgestellte geräteabhängige Steuersoftware für angeschlossene Ein-/Ausgabegeräte.

2 Punkte

23. Warum haben moderne Festplatten Zonen mit verschiedener Anzahl von Sektoren?

Die Sektoren nehmen vom Mittelpunkt zum Rand an Länge hinzu, man kann also, indem man mehr Sektoren in den äußeren Ringen hat als in den inneren Ringen, die Speicherkapazität erhöhen. Versetzte Sektorenübergänge von Zone zu Zone haben den Vorteil, dass der Kopf der Festplatte mehr Zeit zum Spurwechsel hat, falls die Datei so verteilt wurde.

2 Punkte

24. Bei magnetischen Festplatten wird u.a. das Scheduling nach Shortest Seek First angewendet.

a. Erläutern Sie die Funktionsweise von Shortest Seek First!

- Eingehende Schreib- und Leseanfragen werden zusammen mit der Information, wo die zu verwendenden Blöcke liegen, gepuffert
- Der Algorithmus wählt nun die Anfrage aus, deren Blöcke am nächsten an der aktuellen Kopfposition sind.

b. Ist Shortest Seek First auch bei SSDs (Solid State Disks) sinnvoll? Begründen Sie Ihre Antwort.

Nein, da SSD keine Schreib- und Leseköpfe haben, müssen sie nicht aus Performancegründen darauf achten. Die Zugriffszeiten einer SSD sind so schnell, dass es keinen Unterschied macht, wo die Daten hingeschrieben werden. Puffern muss eine SSD trotzdem um neu ankommende Daten zurückzuhalten oder aufzuräumen (trim).

6 Punkte

25. Nennen Sie vier typische Aufgaben eines Uhrentreibers! (siehe Satz: Theorie-5, Folie-44)

Typische Aufgaben eines Uhrentreibers:

1. Pflege der Tageszeit.
2. Verhindern, dass Prozesse länger als zulässig ausgeführt werden.
3. Berücksichtigung der CPU-Auslastung.

4. Verarbeitung von Alarmsystemaufrufen von Benutzerprozessen.
5. Bereitstellung von Watchdog-Timern für Teile des Systems.
6. Profilerstellung, Überwachung, Statistikerfassung.

2 Punkte

26. Was versteht man unter einem Thin Client? (siehe Satz: Theorie-5, Folie-61)

- ein Client, der zur Erfüllung seiner Aufgaben auf die Hilfe eines Servers angewiesen ist.
- (Fat Client: Hard- und Software so gebaut, dass er seine eigenen Aufgaben selbst erledigen kann.)

2 Punkte

27. Was versteht man unter:

a) Unterbrechbaren Ressourcen

- kann dem Prozess, der sie besitzt, ohne unerfreuliche Nebenwirkungen entzogen werden
- mögliche Deadlock-Situationen können so vermieden werden

b) Nicht unterbrechbaren Ressourcen

- kann ihrem aktuellen Besitzer nicht entzogen werden, ohne Gefahr zu laufen, dass dessen Ausführung fehlschlägt

Generell:

- Deadlocks haben immer mit nicht unterbrechbaren Ressourcen zu tun
- Mögliche Deadlocks können normalerweise aufgelöst werden (falls unterbrechbare Ressourcen beteiligt sind), wenn man unterbrechbare Ressourcen neu zuteilt

4 Punkte

28. Wie ist ein Deadlock definiert? (siehe Satz: Theorie-6, Folie-8)

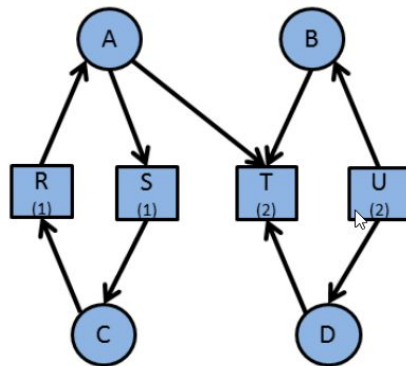
Eine Menge von Prozessen ist blockiert, wenn:

- Jeder Prozess in der Menge auf ein Ereignis wartet
- Dieses kann nur durch einen anderen Prozess aus der Menge verursacht werden

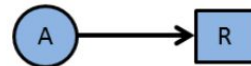
Definition aus dem Buch:

- Eine Menge von Prozessen befindet sich in einem Deadlock-Zustand, wenn jeder Prozess aus der Menge auf ein Ergebnis wartet, das nur ein anderer Prozess aus der Menge auslösen kann

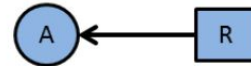
29. Gegeben ist der folgende Ressourcen-Belegungsgraph:



Bedeutung:



A hat Ressource R angefordert



Ressource R von A belegt

R und S können jeweils von einem Prozess genutzt werden, T und U können gleichzeitig von zwei Prozessen genutzt werden.

a) Beschreiben Sie die Zustände der einzelnen Prozesse! Welche Schlussfolgerung ziehen Sie bezüglich des Zustandes des Gesamtsystems?

- Zustand Prozesse:
 - A: belegt R, fordert S und T
 - C: belegt S, fordert R
 - B: belegt U, fordert T
 - D: belegt U, fordert T
- Zustand des Systems:
 - der linke Kreis ist ein Deadlock (A & C bedingen sich gegenseitig)
 - dies könnte vermieden werden, wenn A Ressource R freigibt, falls A die angeforderte Ressource T erhält
 - T wird momentan von keinem Prozess belegt
 - T kann zwar von 2 Prozessen genutzt werden, wird jedoch von 3 Prozessen (A, B, D) angefragt
 - Folgerung: Zustand des Gesamtsystems könnte in deadlock geraten, falls B und D beim "Kampf" um Ressource T den Zuschlag erhalten.

b) Unter welchen notwendigen Bedingungen können Deadlocks auftreten? (siehe Satz: Theorie-6, Folie-9)

Vier Bedingungen, die erfüllt sein müssen:

1. Bedingung des wechselseitigen Ausschlusses
2. Hold-and-Wait Bedingung
3. Bedingung der Ununterbrechbarkeit
4. Zyklischen Wartebedingung

c) Erläutern Sie die verschiedenen Strategien, Deadlocks entgegenzuwirken!

1. Behebung durch Unterbrechung □ (ob Unterbrechung möglich ist, hängt von Art der Ressource ab. Z.B. Laserdrucker anhalten, anderer Druckjob und manuell fortfahren)
2. Behebung durch Rollback □ (Zustand Prozess regelmäßig an Checkpoints in Datei speichern, bei Unterbrechung zurück zum letzten Checkpoint)
3. Behebung durch Prozessabbruch □ (Prozess abbrechen, vielleicht laufen die anderen dann weiter)

(aus Theorie-6, Folie-23)

12 Punkte

30. Nennen Sie die vier Strategien für die Behandlung von Deadlocks! (siehe Satz: Theorie-6, Folie-13)

Strategien für die Behandlung von Deadlocks:

1. Ignoriere das Problem, vielleicht geht es weg. (Vogel-Strauß-Algorithmus)
2. Erkennen und Beheben. Deadlocks zulassen, erkennen und etwas dagegen unternehmen.
3. Dynamische Verhinderung durch sorgfältige Ressourcenzuteilung.
4. Vermeidung von Deadlocks. Eine der vier notwendigen Bedingungen muss prinzipiell unerfüllbar werden.

2 Punkte

31. Was versteht man unter Typ-1- und Typ-2-Hypervisoren?

- VMM (Virtual Machine Monitor) erzeugt die Illusion von mehreren (virtuellen) Maschinen auf der gemeinsamen physischen Hardware
- Ein VMM ist auch bekannt als Hypervisor
- Typ-1, läuft direkt auf der Hardware
 - entspricht in technischer Hinsicht einem Betriebssystem, da es das einzige Programm ist, das im Modus mit den höchsten Privilegien läuft
 - seine Aufgabe ist es mehrere Kopien der tatsächlichen Hardware, sog. virtuelle Maschinen, zu unterstützen (ähnlich den Prozessen, die auf einem normalen Betriebssystem laufen)
- Typ-2 verwendet alle Dienste und Abstraktionen, die von dem zugrunde liegenden Betriebssystem angeboten werden.
 - ist ein Benutzerprogramm, das ähnlich einem normalen Prozess Ressourcen belegt und zeitlich zuordnet
 - gibt immer noch vor, vollständiger Rechner mit einer CPU und verschiedenen Geräten zu sein

Beide Typen müssen den Befehlssatz der Maschine auf sichere Art ausführen.

4 Punkte

32. Was muss der Virtual Machine Monitor mit jedem Befehl machen bei (siehe Satz: Theorie-7, Folie-10)

a. Emulation der Hardware

- der Virtual Machine Monitor muss die Befehle vor Ausführung auf der nativen Hardware modifizieren

b. Prozessorvirtualisierung (d.h. Hardwareunterstützung für Virtualisierung)

- Befehle können ohne Änderung unmittelbar an den Prozessor geschickt werden
- Prozessor kümmert sich selbständig um die Abgrenzung zwischen Gastsystem- und VMM- Prozess

4 Punkte

33. Was unterscheidet echte Virtualisierung von Paravirtualisierung?

Paravirtualisierung unterscheidet sich deutlich von der vollen Virtualisierung, weil sie noch nicht einmal darauf abzielt, eine virtuelle Maschine zu präsentieren, die genauso wie die zugrunde liegende Hardware aussieht. Stattdessen wird bei Paravirtualisierung eine maschinenähnliche Softwareschnittstelle präsentiert und somit die Tatsache ausdrücklich herausstellt, dass es eine virtualisierte Umgebung ist.

Also:

echte Virtualisierung => virtuelle Maschine

Paravirtualisierung => virtualisierte Umgebung

2 Punkte

34. Was versteht man unter einer Virtual Appliance? (siehe Satz: Theorie-7, Folie-15)

- ist eine virtuelle Maschine, die mit dem geforderten Betriebssystem, Compilern, Bibliotheken und Anwendungscode beladen und ausführbereit eingefroren wird
- ist also ein vollständiges, funktionierendes Paket, das unabhängig ist von Betriebssystem, Software, usw.

2 Punkte

35. Erläutern/skizzieren Sie folgende Begriffe:

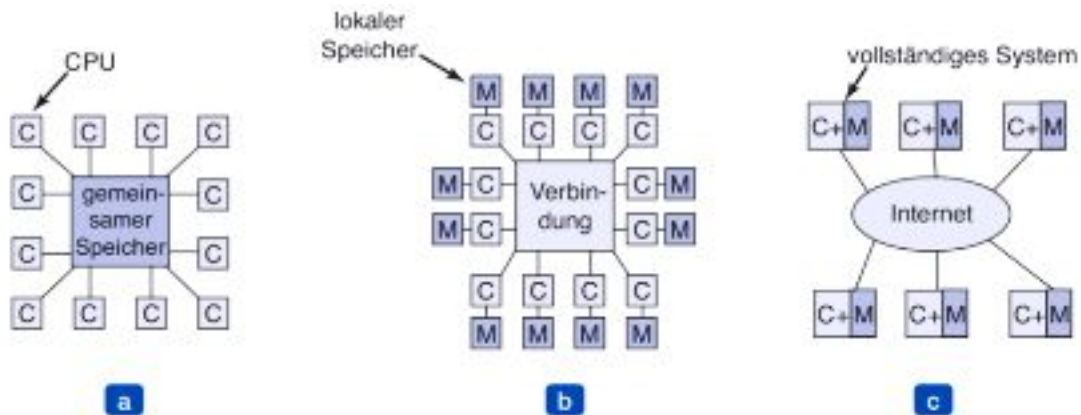


Abbildung 8.1: (a) Multiprozessorsystem mit gemeinsamem Speicher. (b) Multicomputer mit Nachrichtenaustausch. (c) Großräumig verteiltes System.

a. Multiprozessorsystem

- ist ein Computersystem, in dem sich zwei oder mehr CPUs unbeschränkten Zugriff auf ein gemeinsames RAM teilen

b. Multicomputer (Cluster)

- Multicomputer sind eng verbundene CPUs ohne gemeinsamen Speicher. Jede CPU hat wie in *Abbildung 8.1b* ihren eigenen Speicher.

c. (Großräumig) verteiltes System (Grid)

- ähneln den Multicomputern dahingehend, dass jeder Knoten seinen eigenen privaten Speicher besitzt und es keinen gemeinsamen physischen Speicher im System gibt
- verteilte Systeme sind noch loser verbunden als Multicomputer.

6 Punkte

36. In welchen drei Schichten kann man einen Distributed Shared Memory implementieren?

- Hardware
- Betriebssystem
- Software auf Benutzerebene

3 Punkte

37. IT-Sicherheit: Welche Schutzziele gibt es und wodurch werden diese bedroht? (siehe Satz: Theorie-9, Folie-4)

Vertraulichkeit - Enthüllung der Daten

Integrität - Manipulation der Daten

Verfügbarkeit - Dienstverweigerung (Denial of Service)

6 Punkte

38. Ist es möglich, ein sicheres Computersystem zu bauen? Begründen Sie Ihre Antwort!

Gibt es nicht. -> Hardwarefehler, komplexe Software, Software im PC (OS, Anwendungen, BIOS/UEFI, Management Unit, Netzwerkboot, Firmware)

Was man an Sicherheit erreichen will, müsste man mit Performance ;) und Benutzerfreundlichkeit einbüßen.

Im Prinzip nein, ist aber auch von der Komplexität des Systems abhängig.

3 Punkte

39. Womit kann man den Zugriff auf Ressourcen aus der Sicht der IT-Sicherheit steuern? (siehe Satz: Theorie-9, Folie-9)

Trusted Computing base - Ein Referenzmonitor prüft alle Systemaufrufe

Schutzdomänen (Unix-Rechte)- Man kann eine Datei einer Domäne zuordnen.

Owner, group, other -> 3 Schutzdomänen

Zugriffskontrolllisten (ACL) Für jede Datei gezielte Freigabeliste Schutz aus der Sicht des Nutzers

Capabilities Was darf ich mit dem Prozess machen?

2 Punkte

40. Erläutern Sie die Funktionsweise einer Zugriffskontrollliste (Access Control List, ACL).

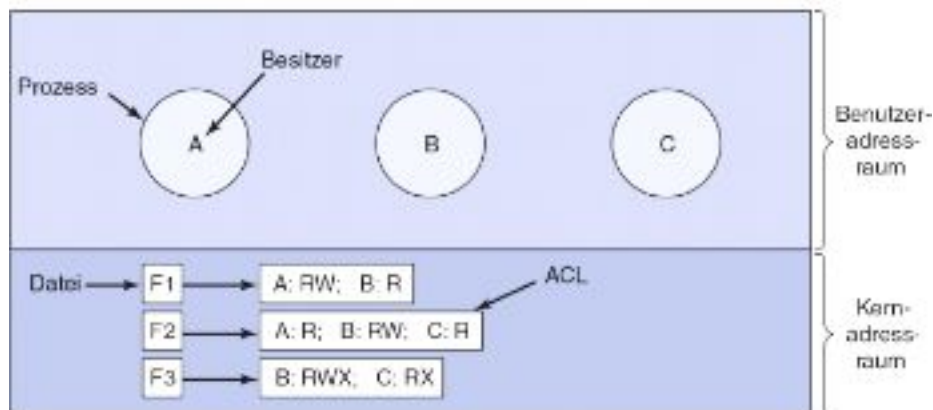


Abbildung 9.6: Verwendung von Zugriffskontrolllisten zur Verwaltung des Dateizugriffs.

nach Tanenbaum:

- zu jedem Objekt wird eine (geordnete) Liste geführt, die alle Domänen enthält, die auf das Objekt zugreifen können
- für jede Domäne wird angegeben, wie der Zugriff erfolgen darf
- zu vergebende Zugriffsrechte sind: Lesen, Schreiben, Ausführen

Definition von ubuntuusers:

Mit Hilfe von Access Control Lists (kurz ACL) ist es möglich, einzelnen Nutzern (oder auch Gruppen) gezielt Rechte an einzelnen Dateien zu geben oder zu entziehen. ACLs ergänzen damit die normale Rechteverwaltung von Linux. Außerdem kann man mit ACLs die einheitliche Vergabe von Rechten für neu angelegte Dateien innerhalb eines Verzeichnisbaumes erzwingen.

4 Punkte

41. Was versteht man unter einem verdeckten Kanal?

Ein verdeckter Kanal ist ein nicht autorisierter Kommunikationskanal, er wird in einem System durch Beobachtung und Manipulation messbarer Leistungsmerkmale des Systems erzeugt. Die wichtigste Voraussetzung sind gemeinsam genutzte Systemressourcen (bspw. CPU, Festplatte oder Netzwerk), zum Senden der geheimen Signalen.

4 Punkte

42. Wofür werden in der IT-Sicherheit Zufallszahlen mit hoher Qualität benötigt?

Kryptographische Algorithmen brauchen Zufallszahlen hoher Qualität zur Generierung von Schlüsseln, die möglichst schwer zu erraten/ermitteln sind. Diese Schlüssel werden für wichtige Anwendungen, wie Verschlüsselung von Daten oder Signaturen, eingesetzt.

2 Punkte

43. Biometrische Authentifizierung:

a. Was versteht man darunter?

Die Anmeldung an Computersystemen mithilfe physischer Merkmale eines Benutzers, die schwer zu fälschen sind.

b. Nennen Sie 3 Beispiele!

Fingerabdrucksensor, Stimmerkennungssensor, Iris-Erkennung

5 Punkte

44. Erläutern/skizzieren Sie einen Pufferüberlaufangriff?

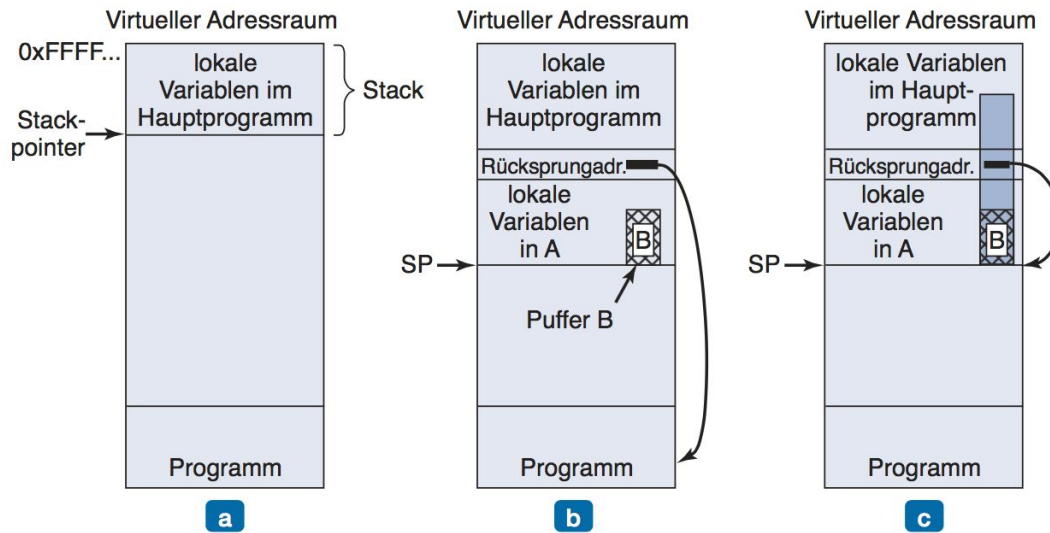


Abbildung 9.21: (a) Zustand bei laufendem Hauptprogramm. (b) Nachdem die Prozedur A aufgerufen wurde. (c) Pufferüberlauf wird dunkler dargestellt.

- von einem Programm werden zu große Datenmengen in den dafür reservierten Speicherbereich geschrieben
- ist das System nicht entsprechend abgesichert, werden die Daten außerhalb des reservierten Bereiches in den Zielbereich für diesen Angriff geschrieben
- diese Daten können so gezielt platziert werden, um dann bspw. als Programmcode eines anderen Prozesses ausgeführt zu werden

4 Punkte

45. Nennen Sie 4 Arten von Malware! (siehe Satz: Theorie-9, Folie-48)

Trojaner, Viren, Würmer, Spyware, Rootkits

2 Punkte

46. Was versteht man unter einem polymorphen Virus?

- polymorphe Viren ändern ständig ihren eigenen Programmcode, indem sie Kopien von sich selbst erstellen, die durch Permutation geändert wurden

2 Punkte

47. Nennen Sie die drei Leitlinien für den Entwurf von Betriebssystemen! (siehe Satz: Theorie-12, Folie-6)

Leistungsfähigkeit, Einfachheit, Vollständigkeit

2 Punkte

BS-Praxis

1. Nennen Sie 3 Arten von Betriebssystemen!(siehe Satz: Praxis-1, Folie-10)

Betriebssysteme für Großrechner, Server, Multiprozessorsysteme, PCs, Handheld-Computer, eingebettete Systeme, Sensorknoten, Smartcards, Echtzeitbetriebssysteme

3 Punkte

2. Was ist der Unterschied zwischen einem „harten“ und einem „weichen“ Echtzeitsystem?

Harte Echtzeitsysteme sind zB für industrielle Fertigungssteuerung, wo eine absolute Garantie vorliegen muss, dass eine Aktion zu einer bestimmten Zeit ausgeführt wird. Bei weichen Echtzeitsystemen können verpasste Deadlines auch toleriert werden, Einsatzmöglichkeit zB in Multimediasystemen.

2 Punkte

3. Wie viele Bytes enthält 1 TByte Speicher?

Wenn Terabyte (TB) gemeint sind, dann 10^{12} Byte,
wenn Tebibyte (TiB) gemeint sind, dann 2^{40} Byte.

1 Punkte

4. Mit welchem UNIX Befehl kann man sich die aktiven Prozesse anzeigen lassen?

ps [-e]

top

1 Punkte

5. Am primären IDE Controller ist eine Festplatte als Master angeschlossen. Sie habe die folgende Struktur:

<i>Partitionen mit Partitionsnummer</i>	<i>Logische Laufwerke</i>	<i>Format</i>	<i>Windows</i>	<i>Linux</i>
1. Primäre		NTFS		
2. Primäre		Linux Swap		
3. Erweiterte				
	1. logisches Laufwerk	NTFS		
	2. logisches Laufwerk	EXT4		

Geben Sie die Laufwerksbuchstaben unter Windows (typische Standardvergabe) und die Bezeichnungen unter Linux für die einzelnen Partitionen an!

Vergleiche zweite Klausur, Aufgabe 5.

Lösung aus Übungsklausur

Partition mit Partitionsnummer	Log. Laufwerk	Format	Windows	Linux
Primär – 1		NTFS	C:	/dev/sda1 (a weil 1. Hdd im system)
Primär - 2		Linux Swap		/dev/sda2
Erweitert - 4				/dev/sda4
	1. log. Laufwerk	FAT	D:	/dev/sda5 (log. Lw starten immer bei 5)
	2- log. Laufwerk	Ext 3		/dev/sda6

7 Punkte

6. In welchem UNIX-Verzeichnis befinden sich üblicherweise Konfigurationsdateien?

/etc/

1 Punkt

7. In welchem Runlevel sollte man einen Server standardmäßig starten? Begründen Sie Ihre Antwort! (siehe Satz: Praxis-2, Folie-3)

Runlevel 3 (Netzwerkbetrieb, über das Netzwerk erreichbare Ressourcen sind nutzbar, eine grafische Oberfläche steht nicht zur Verfügung)

→ alles was für Serverbetrieb nötig ist, ist vorhanden. Grafische Oberflächen bieten Angriffspunkte und kosten Systemressourcen.

Runlevel	Beschreibung
0	Shutdown. Alle Netzverbindungen werden geschlossen, Dateipuffer werden geschrieben, Mounts auf Partitionen werden entfernt (d. h. die im Verzeichnisbaum eingebundenen Datenträger werden ausgehängt).
S	Single-User-Runlevel; niedrigster Systemzustand für Wartungsarbeiten, in dem ausschließlich Systemressourcen wie Festplatten oder Dateisysteme aktiv sind.
1	Einzelnutzerbetrieb ohne Netzwerk mit ausschließlich lokalen Ressourcen. In vielen Implementierungen identisch mit 'S'.
2	Lokaler Mehrnutzerbetrieb ohne Netzwerk mit ausschließlich lokalen Ressourcen. Unter einigen Linuxdistributionen (z. B. Debian) wird in Runlevel 2 auch das Netzwerk konfiguriert.
3	Netzwerkbetrieb, über das Netzwerk erreichbare Ressourcen sind nutzbar, eine grafische Oberfläche steht nicht zur Verfügung. Firewalls sollten aktiviert werden.
4	Ist normalerweise nicht definiert. Kann aber für diverse Dienste genutzt werden.
5	Wie 3, zusätzlich wird die grafische Oberfläche bereitgestellt.
6	Reboot. Alle Netzverbindungen werden geschlossen, Dateipuffer werden geschrieben, Mounts auf Partitionen werden entfernt.

2 Punkte

8. Mit welchem Befehl kann man die Liste der Dateien im Verzeichnis /tmp in eine Datei liste.txt schreiben? Hängen Sie die Liste der Dateien im Verzeichnis /var an diese Liste an! (siehe Satz: Praxis-2, Folie-11/ 30)

ls /tmp > liste.txt

ls /var >> liste.txt

3 Punkte

9. Geben Sie die Befehlsfolge zum Ausschneiden der 5. Spalte aus der Datei /etc/passwd an! Trennzeichen der einzelnen Spalten ist der Doppelpunkt („:“).

```
cat /etc/passwd | cut -d: -f5
```

(-d [TRENNZEICHEN]
-f [SPALTE])

In der Fragestellung fehlt, “und geben Sie sie am Bildschirm aus”. Die Antwort auf seine wortwörtliche Frage (die Spalte aus der Datei auszuschneiden) wäre in etwa so:

(!Achtung, nicht wirklich ausführen als root :D :D)

```
cut -d: -f1-4,6,7 /etc/passwd > neu.txt ; cat neu.txt > /etc/passwd
```

2 Punkte

10. Erzeugen Sie im Homeverzeichnis eine Datei „dateirechte.zufall“, welche 768 Byte Nullen enthält. Geben Sie dieser Datei folgende Berechtigungen: (siehe Satz: Praxis-2, Folie-46 / 64)

- Eigentümer: Lesen, Schreiben - Gruppe: Lesen - Andere: Keine

```
dd if=/dev/zero of=dateirechte.zufall bs=768 count=1
```

```
chmod 640 dateirechte.zufall
```

(640 → Eigentümer bekommt 6 (r+w), Gruppe bekommt 4 (r-w),
Other bekommt 0)

7 → read+write+ausführbar (x)

6 → read + write

5 → read only + ausführbar (x)

4 → read only

3 → write only + ausführbar (x)

2 → write only

1 → ausführbar only

0 → keine Rechte

	R	W	X	
Owner	1	1		$4 + 2 = 6$
Group	1			4
Other				0

3 Punkte

11. Schreiben Sie ein Shellsript, welches die Zahlen von a bis b addiert und das Ergebnis auf dem Bildschirm ausgibt: $x = a + (a+1) + (a+2) + \dots + b$. Die Zahlen a und b sind dem Shellsript als Kommandozeilenparameter zu übergeben! Speichern Sie das Shellsript im Homeverzeichnis und machen Sie es ausführbar.

```
#!/bin/bash
a=$1
b=$2
result=$a

for ((i=$a+1; i<=$b; i++))
do
    result=$((result + $i))
done

echo "Das Ergebnis von $a + (($a+1)) + ... + $b ist: $result"
exit 0
```

Speichern unter dateiname, dann:

```
chmod 755 dateiname
```

10 Punkte

12. Mit welcher Befehlskombination können die den Inhalt der Festplattenpartition /dev/sda3 auslesen, komprimieren (möglichst schnell, also niedrige Kompressionsrate) und in eine Datei backup.dd schreiben? Darf dabei die Partition /dev/sda3 von einem Betriebssystem verwendet werden? Begründen Sie Ihre Antwort! (siehe Satz: Praxis-2, Folie-66)

1 : 1 Kopie ohne Komprimierung

```
dd if=/dev/sda3 of=/backup.dd
```

Mit Komprimierung (wie verlangt) Kompr. Rate: 1-9

```
dd if=/dev/sda3 | gzip -1 > of=/backup.dd
```

(Der Input "if" wird mit "|" an "gzip" gepiped und dessen Output zum Output von dd, "of" (steht für output file), geschickt.)

oder:

```
dd ibs=64k if=/dev/sda3 | gzip --fast | dd obs=64k of=/backup.dd
```

(so hatte es Brunner gemacht - das "--fast" erfüllt die Anforderung, es möglichst schnell zu komprimieren)

- sie darf nicht verwendet werden, weil sonst die Dateien verändert werden, und damit irgendwelche Tables hinüber sind
- sie darf nur lesend verwendet werden (read-only)

5 Punkte

1) Nennen Sie die Hauptaufgaben des BIOS! (siehe Satz: Praxis-1, Folie-11)

bzw. UEFI

- Konfiguration von Komponenten
- **Power On Self Test**: überprüft das Vorhandensein aller relevanten Komponenten
- Start des Betriebssystems

2. Wozu sind Gerätetreiber notwendig? (siehe Satz: Praxis-1, Folie-23)

erzeugen einheitliche schnittstelle zwischen betriebssystem und gerät

3. Wie viele Bytes enthält 1 Gibibyte (GiB) Speicher? Bitte binär nach IEC angeben!

2^{30}

4. Wozu dient bei einer Partitionierung mittels GUID der Protective MBR? (siehe Satz: Praxis-2, Folie-20)

damit ein Betriebssystem, das nicht mit GUID umgehen kann, den Datenträger nicht formatiert (Schutz gegen versehentliches Formatieren)

5.

<i>Partitionen mit Partitionsnummer</i>	<i>Logische Laufwerke</i>	<i>Format</i>	<i>Windows</i>	<i>Linux</i>
Primär – 1		NTFS		
Primär – 2		Linux Swap		
Erweitert – 4				
	1. logisches Laufwerk	FAT		
	2. logisches Laufwerk	ext3		

Partionen mit

<i>Partitionsnummer</i>	<i>Logische Laufwerke</i>	<i>Format</i>	<i>Windows</i>	<i>Linux</i>
primär 1		NTFS	C:	/dev/sda1
primär 2		Linux Swap		/dev/sda2
erweitert 4				/dev/sda4
	1. logisches Lw	FAT	D:	/dev/sda5
	2. logisches Lw	ext3		/dev/sda6

sda weil erste Platte im System, siehe Aufgabe

bei zweiter Platte sdb

erweitert - erweiterte partitionstabelle liegt vor, geht immer weiter mit 4

primär ist immer 1-4, wenn nichts angegeben leer lassen

6. In welchem UNIX-Verzeichnis befinden sich üblicherweise Konfigurationsdateien?
/etc

7. Sie sollen einem Mitarbeiter eine Workstation bereitstellen. Welches Standard-Runlevel wählen Sie dafür? Begründen Sie Ihre Antwort!
Level 5, wegen grafischer Oberfläche

grafisches Oberfläche+Level 3
Level 3: multiuser+Netzwerk
Für zB Server Level 3
(grafische Oberfläche mit startx starten, aber: Sicherheitslücke)

8. Sollte man auf einem Server nicht zwingend benötigte Software wie Compiler, Quellcodes für Kernel und Programme usw. installieren? Begründen Sie Ihre Antwort! Hinweis: Beachten Sie dabei Sicherheitsaspekte, insbesondere auch den Fall eines erfolgreichen Einbruchs in den Server!

Man sollte nicht benötigte Software nicht installieren, alles was läuft könnte eine Sicherheitslücke mitbringen, wenn Kompilerquellcode vorliegt kann dort eine Sicherheitslücke eingebaut werden und in den Kernel eingepatcht werden.

Nicht installieren
jede software kann eine Sicherheitslücke haben
wenn quellcode vorliegt, ist hintertür einkompilieren möglich

9. Geben Sie die Befehlsfolge zum Ausschneiden der 1. Spalte aus der Datei /etc/passwd an! Trennzeichen der einzelnen Spalten ist der Doppelpunkt („:“).
cat /etc/passwd | cut -d ":" -f 1
oder
cut -d ':' -f 1 /etc/passwd

10. Erzeugen Sie im Homeverzeichnis ihres Klausur-Users eine Datei „dateirechte.zufall“, welche 512 Byte binäre Zufallszahlen enthält. Geben Sie dieser Datei folgende Berechtigungen:

- Eigentümer:
Lesen, Schreiben, Ausführen
- Gruppe:
Lesen, Ausführen
- Andere:
Lesen, Schreiben

zu home mit "cd"
dd if=/dev/urandom of=dateirechte.zufall bs=1 count=512

gleiches wäre

```
dd if =/dev/urandom of= dateirechte.zufall bs=2 count=256
```

```
dd if =/dev/urandom of= dateirechte.zufall bs=516 count=1
```

(if = input file, of = output file, bs = block size, count = größe)

Im besten Fall blocksize zu standard der festplatte setzen

Größe ist count*bs

dev/null zum löschen

dev/zero mit null überschreiben

1. Eigentümer (lesen 4 + schreiben 2 + ausführen 1) 7

2. Gruppe (lesen 4 + ausführen 1) 5

3. Andere (lesen 4 + schreiben 2) 6

```
chmod 756 dateirechte.zufall
```

11. Schreiben Sie ein Shellsript, welches die Zahlen von a bis b wahlweise multipliziert oder

addiert und das Ergebnis auf dem Bildschirm ausgibt:

$x = a * (a+1) * (a+2) * \dots * b$ bzw.

$x = a + (a+1) + (a+2) + \dots + b$

Die Zahlen a und b sowie die gewünschte Operation (Addition oder Multiplikation) sind dem Shellsript als Kommandozeilenparameter zu übergeben! Speichern Sie das Shellsript im Homeverzeichnis ihres Klausur-Users und machen Sie es ausführbar.

```
vi add_multi
```

```
#!/bin/bash
```

```
a=$1
```

```
OP=$2
```

```
b=$3
```

```
#For-schleife
```

```
result=$a
```

```
for((i=$a+1;i<=$b;i++)) # $a schon result zugewiesen
```

```
do
```

```
    result=$((result $OP $i))
```

```
done
```

```
echo "Ergebnis: $result"
```

```
#echo "$a$OP$((a+1))$OP$((a+2))$OP...$OP$b = $result"
```

```
#bsp: ./add_multi 1 "*" 10
```

```
#bsp: 1*2*3*...*10 = 3628800
```

```
exit 0
```

chmod 755 add_multi

./add_multi 2 "*" 5

./add_multi 2 * 5

(2*3*4*5)

Ergebnis: 120

./add_multi 1 + 3

Ergebnis: 6

(1+2+3)

Oder

if [-z \$1] then echo -n "Zahl a: "; read a;else a=\$1; fi

if [-z \$2] then echo -n "Operator "; read OP;else OP=\$2; fi

if [-z \$3] then echo -n "Zahl b: "; read b;else b=\$3; fi

-z testet ob Variable leer, wenn leer anforderung dieser. wenn nicht leer werden sie zugewiesen

12. Mit welcher Befehlskombination können sie den Inhalt der Festplattenpartition /dev/sda3 auslesen, komprimieren (möglichst schnell, also niedrige Kompressionsrate) und in eine Datei backup.dd.gz schreiben? Das Backup soll ein komplettes binäres Image der Partition sein (also unabhängig vom Dateisystem).

Darf dabei die Partition /dev/sda3 von einem Betriebssystem verwendet werden?

Begründen Sie Ihre Antwort!

dd *ibs=64k* if=/dev/sda3 | gzip --fast | dd *obs=64k* backup.dd.gz

sie darf nur lesend verwendet werden) sie darf nicht verwendet werden, weil sonst die Dateien verändert werden, und damit irgendwelche Tables hinüber sind.

ibs/obs = input/output block size (optional)