

linux OS kernel final

1. virtual file system

- (O)所有類型的資料結構至少需要一個inode
- 目錄、檔案、symbolic link、socket

2. (O)所有檔案都是透過vfs做操作。

3. kernel配記憶體，在MMU，用哪個function來配置？

- kmalloc
- vmalloc

4. linux mmu配置實體記憶體時是以page為單位(page數量必須是2的次方)

5. jiffix是計算時間的單位，要透過jiffix判斷時間的前後順序，不能只用b-a的結果來判斷，因為jiffix有overflow的問題，需要透過time before, time after_eq, time_before_eq的function來做計算，用以判斷哪些事件比較早或晚。

6. HZ = 100，每10個millisecond發生一次tick

- HZ = 1000，1 millisecond 發生一次tick

7. locking機制避免發生(race condition)，lock保護(critical region)，critical region範圍為(lock到unlock中間)。

8. critical region主要是因為共享資料區域(例如:global count)

9. 有多少條ksoftirq就代表有幾個cpu (\$ cat /proc/cpuinfo)

- ksoftirq的數量可以改變嗎？不能！softirq根據幾位元的電腦決定，最大數量是個constant(被架構所限制)

10

- disable_irq(7)
- disable_irq(5)
- disable_irq(7)

- enable_irq(5)
 - enable_irq(7)
 - 此時command irq(7)是不會執行的，因為irq(7)還是在被disable的狀態。
11. 網路卡封包「copy到buffer」，在中斷的上半部。「封包重組」在中斷的下半部。
 12. (O)interrupt handler 隨時都可能被執行。
 13. irq0是時間中斷，irq1是鍵盤中斷 8042
 14. 註冊中斷處理函式，有一些flags
 - irq_disable: 禁止其他irq
 - irq_random_sample: 加入attribute pool
 - irq_timer: 專門處理時間
 - irq_share: 這個interrupt line是有可能有多個interrupt handler，不會同時發生，一次還是只有一個interrupt發生。
 15. vfs mount的結構中，不能禁止哪個項目？
 - 不可以setuid，不可以setgid
 - 可以指定flag
 - 可以禁止不要存取device
 - 不能禁止讀取檔案的選項(因為mount上去的file至少是read only)
 16. cache
 - 使用中目錄的linked list (dentry, directory structure)
 - inode會不會cache檔案的內容? (不會)
 17. slab的設計會考慮哪些項目？
 - 記憶體是會頻繁的分配跟釋放
 - 避免記憶體產生破碎
 - 可以提高free list的使用效率
 - 有辦法減少記憶體配置所造成的浪費嗎?(不能，因為還是一個page一個page的要，甚至還會花更多的記憶體配置)不會讓記憶體更節省。

18.

(O)zone上面某些硬體只能存取特定的區域

(O)在某些架構下的實體記憶體還虛擬記憶體的方式，在某些情況下無法直接map到kernel memory位址空間，透過VMA...

(X)high memory可以永久對應到kernel memory (沒有辦法永久對應)

(O)zone normal是都可以透過mapping方式mapping到kernel memory

19.

```
HZ = 1000;
delay = jiffies + 2 * HZ;
while(time_before(jiffies, delay)){ //這個while會delay兩秒鐘(2000 milliseconds)
}
}
```

20.

(O)jiffix是用來記錄系統從開機到現在的tick數 (jiffies/HZ)

(?)jiffies * HZ ← 沒有意義

21. jiffies在32bit位元電腦裡面，秒數 乘上 HZ，如果大於 2^{32} ，就overflow，可以算一下多少天會overflow，64bit會需要幾年。

- 假設 $HZ = 100$ ，一天內會發生 864,000 個 jiffies， $Overflow_days = (2^{32} - 1) / Jiffies_per_day = (4294967295 / 864,000) \approx 4970.27$
- 所以，在 32 位元系統中，大約需要 4970.27 天（或約 13.63 年）的時間，才會發生 jiffies 的溢出。
- 64位元，假設 $HZ = 100$ ，則一天內的 jiffies 數量仍然為 864,000。
- $Overflow_days_64bit = (2^{64} - 1) / Jiffies_per_day = (18446744073709551615 / 864,000) \approx 21350398283.43$
- 所以，在 64 位元系統中，大約需要 21350398283.43 天（或約 58494298.71 年）的時間，才會發生 jiffies 的溢出。

21.

- tick rate越高，kernel timer的精確度越高。

- tick rate越高，可以讓pull跟select的時間會更準。
- tick rate越高，process preemption的時間更準。
- (X)tick rate越高，可以減輕系統的排程負擔。

22. tasklet的schedule順序，

- 檢查tasklet的status
- call task scheduler
- 儲存現在的interrupt狀態
- disable local的interrupt
- 呼叫softirq
- 還原原本的interrupt

23. interrupt的top可以放對時間很敏感的，會需要跟硬體配合的放top，需要保證不被其他interrupt打斷的也放top，需要很長時間等待的放在bottom。

24. softirq

在哪個程式段落，不會檢查或執行softirq? D

(A)從硬體中斷返回code的時候，會檢查softirq

(B)kernel ksoftirqd(在後半段，也會檢查)

(C)任何想要預期有沒有擱置的softirq的程式碼(例如:網路的subsystem，會檢查哪些softirq沒執行，封包重組)

(D)中斷處理函式在top裡面檢查softirq (他是後半段的東西，所以不檢查!)

25. kernel 2.5 哪個機制不會被用到? ANS: BH handler

26. 當cpu遇到程式導致的錯誤例如:page fault, division by 0會怎麼處理?

- 不會產生硬體中斷
- (O)使用softirq處理
- 不會發生kernel panic
- 不會延後指令的執行

27. 問答題:

vfs裡面會有一些data structure，列出4種主要的資料結構:

1. filesystem mount 需要super block
2. 有sb後開始建立目錄建檔案，file structure & dentry
3. inode

kernel synchronization的方法:

- Mutex
- Semaphore (counting & binary)
- Spin lock
- Atomic operation