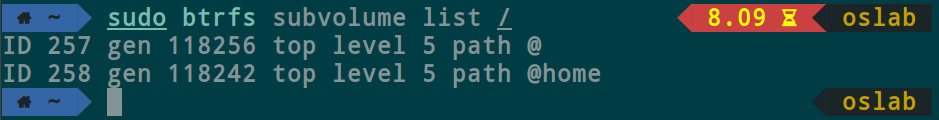
**問題一：列出所有的btrfs的sub-volume (sudo btrfs subvolume list /)**

(1)運行結果



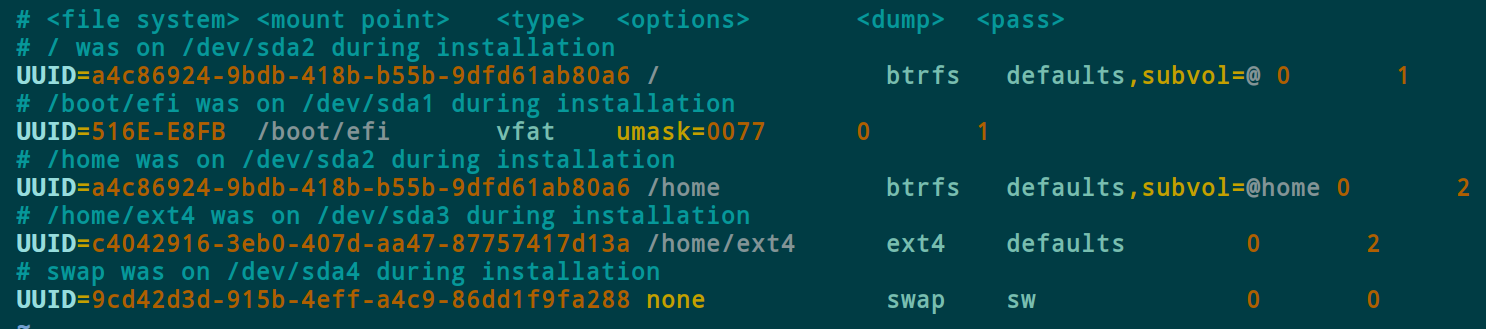
(2)sub-volume意思

sub-volume是file system的一部分，擁有自己獨立的檔案結構，可共享文件範圍，跟LVM logical volume(用block-level snapshot)不同，是file extent-based；snapshot也是一種sub-volume，但是具有original

sub-volume的原始內容。

**問題二：請問/etc/fstab內各個欄位的意思為何？(sudo vim /etc/fstab)**

(1)運行結果



(2)欄位意思

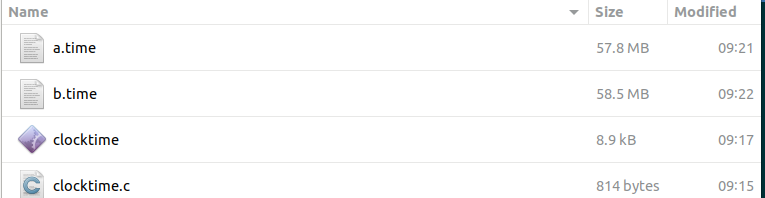
|  |  |  |
| --- | --- | --- |
| **順序** | **欄位** | **描述** |
| 1 | device | 設備的描述，也可以是UUID。 |
| 2 | mount point | 掛載的目錄。 |
| 3 | filesystem | 該磁區的檔案格式，例如：ext2，vfat，ntfs等等 |
| 4 | parameters | 檔案格式參數區。 |
| 5 | dump | 是否可以被 dump 指令備份，  0：不要做備份﹔  1：要做備份﹔  2：要做備份，重要度比 1 小。 |
| 6 | fsck | 是否於開機時以 fsck 檢驗磁區，  0：不檢驗﹔1：先檢驗﹔2：後檢驗。 |

Note (YOUTUBE筆記)：

QUID mount 比較好，每顆硬碟也自己的代號，重整線路後仍可重新開機。

**問題三：tar後a.time和b.tme的內容一樣嗎？**

(1)運行結果 不一樣，因tar會有時間差

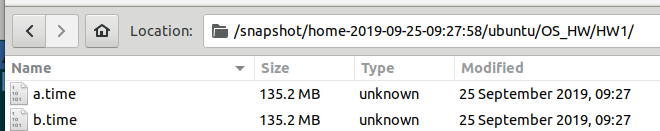


Note (YOUTUBE筆記)：

有可能差很多，若tar需要一小時會差一小時

**問題四：snapshot備份後解壓縮以後a.time和b.tme的內容一樣嗎？（可以使用檔案管理員解開即可）**

(1)運行結果 一樣，因snapshot時間會同時

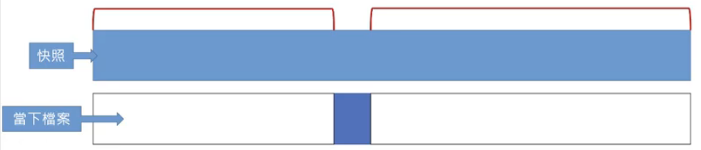


Note (YOUTUBE筆記)：

Snapshot是奈秒等級，filesystem仿佛凍結，對目標目錄做快速紀錄，若有快照進行檔案更新，會額外建立一個檔案，原始檔案與更新檔案共享大部分的資料，有一部分資料沒有共享

缺點：記憶體不連續配置，需跳兩次位址，對於硬碟和快取速度會降低

解決方案：btfrs可以後來把分開的部分變成連續



**問題五：參考上面文章，或其他網路文章，說明資料庫的online backup和offline backup的差別**

1. 比較

|  |  |  |
| --- | --- | --- |
| 項目 | online backup | offline backup |
| 特點 | 可允許client進來操作並且同時備份，被開啟的資料通常是副本，雖然無法確保被寫入的時間順序完全一致，但是數據庫恢復的過程中，修改容易保持一致 | 當備份時，雖然DB2還是開啟的，所有client端無法會連進來，確保所有資料時間一致 |
| 優點 | 1.透過加密保護資料  2.可以異地備份  3.防止物理偷竊與自然災害  4.便宜(每月5至12美元)  5.專業IT團隊管理 | 1.快速備份和恢復，因硬體傳輸快  2.實作簡單，USB即可備份  3.攜帶性高  4.不透過網路更安全 |
| 缺點 | 1.需要透過雲端，無法直接控制資料  2.網路若故障，備份跟恢復速度急遽下降  3.上傳與恢復透過網路速度慢 | 1.容易被物理損壞(被水用到)  2.通常未加密，容易被偷就被破解散播  3.有老化跟壞掉的風險 |

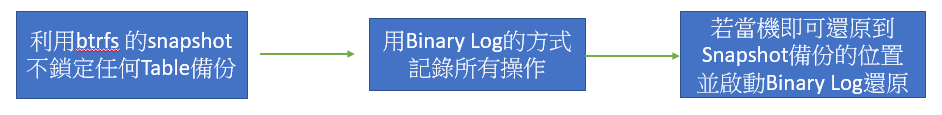
Note (YOUTUBE筆記)：

online backup：邊用邊備份 例：Oracle

offline backup：要停機備份例：MySQL

**問題六：如果使用Btrfs的話，可以在不停機的情況下備份mySQL嗎？(申論題，無對錯)**

1. 可以達到類似效果，可先用btrfs的snapshot的方式在不鎖定任何表的情況下進行快照備份，然後啟動MySQL的Binary Log的機制，紀錄下從此開始的所有動作，若遇到當機可以從snapshot的位置透過Binary Log記下的所有操作來進行還原，但是會有時間差的問題。

示意圖：

另外，Mysql官方有提供**mysqldump備份機制，針對**InnoDB tables可以不鎖住任何table的狀況下進行online backup，只要使用 [--single-transaction](https://dev.mysql.com/doc/refman/5.7/en/mysqldump.html#option_mysqldump_single-transaction) 選項即可。

來源網址：<https://dev.mysql.com/doc/mysql-backup-excerpt/5.7/en/backup-methods.html>

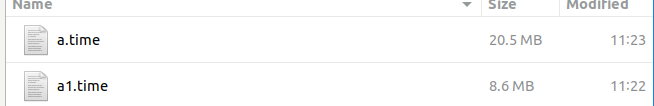
Note (YOUTUBE筆記)：

要思考MySQL記憶體資料突然不見會怎麼樣

**問題七：上述三個指令的時間**

(1)time cp --reflink=always a.time a1.time

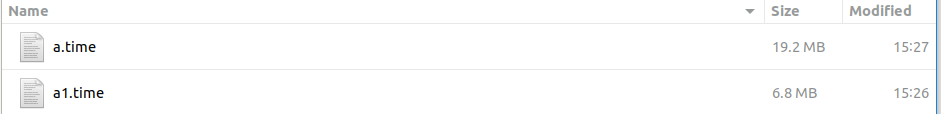
時間不一樣，雖然一開始用--reflink=always複製是共用同一份檔案，但是a.time不斷被修改更新，會把a.time更新的內容存下來，但是會有記憶體碎裂的問題



1. time cp a.time a1.time

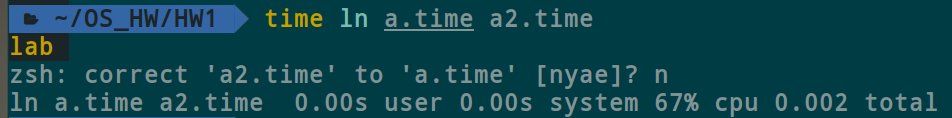
時間不一樣，僅複製當下下指令的檔案，並且拆成兩份不同的檔案，a.time不斷更新，a1.time是當下下指令的檔案，故不一樣。

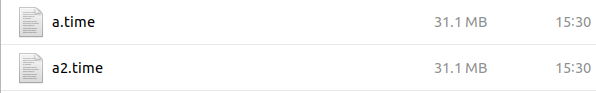


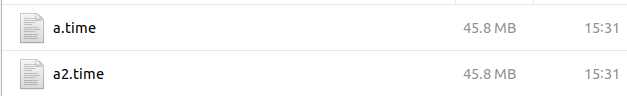


1. time ln a.time a2.time

時間一樣，因為這是軟連結，其實兩個是相同的檔案，若修改其中一個，另一個也會更動，因為是link同一份檔案。



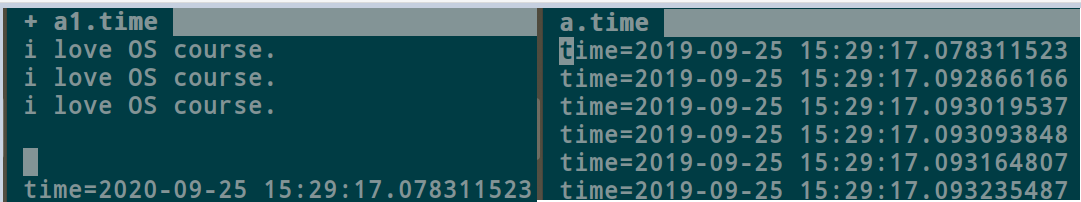




**問題八：a.time是否會隨之改變內容**

(1)vim a1.time /\*任意修改a1.time\*/

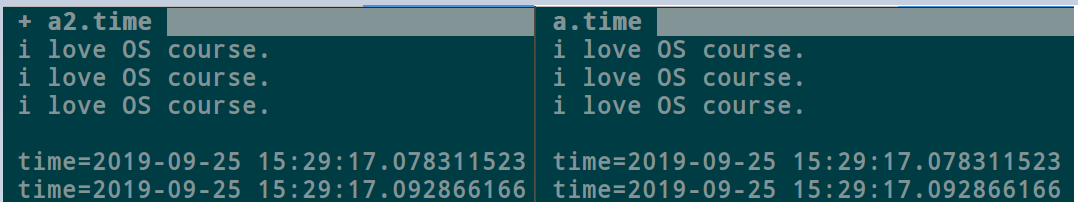
不會，因為--reflink=always複製是共用同一份檔案，但是後面的修改是各自儲存的。



**問題九：a.time是否會隨之改變內容**

(2)vim a2.time /\*任意修改a1.time\*/

會，因為是軟連結，兩個其實是一樣的檔案，會造成修改其中一個，另一個一起變動。



Note (YOUTUBE筆記)：

Vim.rc建立使用者 每個人都會複製一份，用reflink=always，若讀取a.time和a1.time是讀取同一份檔案，若修改a.time，a1.time不會被更改，類似快照，修改

的檔案存在外面，但是會碎裂，需額外資源處理。

延伸閱讀 ZFS Note (YOUTUBE筆記)：

ZFS伺服器等級用的檔案系統

家庭用EXT4(快照功能依賴管理員較弱)

BTRFS學ZFS 不到10年，太新還需時間穩定