سجل الاقلاع الممتد

القسم الممتد والأقراص المنطقية

Extended Boot Record

EXTENDED & LOGICAL PARTITIONS

مسوّدة

يناير/كانون الثّاني 2019

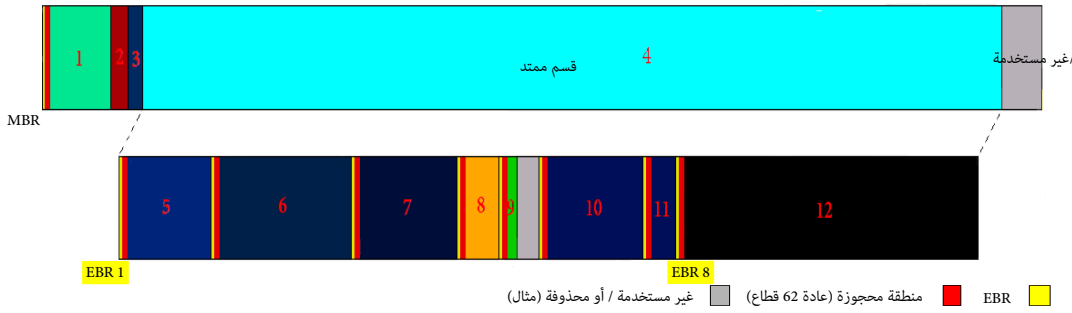
يُوزّع مجّانًا ولا يُباع

سجل الإقلاع الممتد

في نظام تقسيم القرص دوس [2] سجل إقلاع القسم الممتد ERS/EBR/EPBR [1][1] عبارة عن واصف للقسم المنطقي في القسم الممتد. عادة هذا الأخير، باستثناء القسم الأولي الأول [7][8][9]، يمكن أن يأخذ بقية المساحة الغير مقسمة على القرص، أو يأخذ مساحة أحد الأقسام الأولية الثلاثة أو حتى جميع الأقسام الأربعة الأولية. في جدول أقسام MBR. (أنظر للشكل)

المدخلة التي تصف القسم الممتد تملك حقل عنوان بداية LBA [6] وحقل عدد القطاعات أذان يصفان المساحة التي يمكن أن تقع فيها الأقسام المنطقية. (راجع بنية القسم الممتد أدناه).

تقنيا، القسم الممتد لا يمكن استخدامه منفرد لأنه ليس قسم مثل بقية الأقسام الأولية ولكنه وعاء [10] يحتوي على لائحة موصولة من الأقسام المنطقية (أقراص منطقية). هذه القائمة يمكن أن تكون بطول كيفي (حسب المساحة المخصصة للقسم الممتد). لكن بعض نسخ FDISK ترفض إنشاء أقسام منطقية أكبر من عدد المحارف المتوفرة للأقراص، مثلا، في مايكروسوفت دوس القرص الأخير 26، وفي نوفل دوس 7+ ستكون 32.



بنية EBR

سجل إقلاع القسم الممتد EBR يملك نفس بنية MBR؛ لكنه يستخدم فقط مدخلتان أوليتان من جدول أقسام ممتد يبدأ أيضا عند الحيد 1BEh. وتوقيع سجل إقلاع إجباري 0xAA55 في نهاية القطاع [1]، هذا الرقم السحري يظهر في محرر القرص بترتيب 0x55 أولا ثم 0xAA أخيرا، لأن الأجهزة المتوافقة مع أنظمة IBM تخزن قيم 2-بايت الست عشرية بترتيب نهوي صغير.

سجل الاقلاع الممتد الأول EBR 1 (أحيانا يكون الوحيد) يقع دائما في أول قطاع من القسم الممتد.

وبخلاف الأقسام الأولية المحدودة (4 مدخلات فقط) التي يتم تعريفها عن طريق جدول أقسام واحد داخل سجل MBR؛ في القسم الممتد، كل سجل إقلاع ممتد EBR يسبق القسم المنطقي الذي يصفه [2]. وفي حالة وجود قسم منطقي ثاني/تالي، سجل الاقلاع الممتد الأول EBR 1 سوف يتضمن مدخلة تشير إلى سجل الاقلاع الممتد التالي EBR 2؛ وبهذه الطريقة يصبح عندنا عدة سجلات إقلاع ممتدة تشكل قائمة موصولة [3] هذا يعني أن عدد الأقسام المنطقية تحدده فقط مساحة القرص المخصصة للقسم الممتد [4].

أنظمة ويندوز (بما فيها XP) كانت تستخدم قياسات القرص أو CHS في محاذاة الأقسام المنطقية ضمن القسم الممتد، لكن منذ ويندوز فيستا أصبح يستخدم حد 1-ميغابت، ونتيجة لهذا الاختلاف في المحاذاة، مدير الأقراص المنطقية في ويندوز XP يمكن أن يحذف القسم الممتد بدون تحذير المستخدم [3][5].

بنية سجل إقلاع القسم الممتد العامة			حجم (بايت)	الإزاحة ضمن قطاعات EBR	
				عشري	ست عشري
لا تستخدم؛ ومعبئة بأصفار؛ وقد تتضمن محمل إقلاع آخر، أي VBR، مثلا مقترن بأقسام AAP أو أية شفرة [11]			446	000 – 445	000 – 1BD
المدخلة الأولى # 1	(إزاحة وحجم القسم الحالي)	جدول الأقسام	16	446 – 461	1BE – 1CD
المدخلة الثانية # 2	(إزاحة والحساب حتى نهاية القسم المنطقي التالي)		16	462 – 477	1CE – 1DD
المدخلة الثالثة # 3	لا تستخدم[4] ومعبئة بأصفار		16	478 – 493	1DE – 1ED
المدخلة الرابعة # 4			16	494 – 509	1EE – 1FD
توقيع الاقلاع 55AAh بترتيب بايت شكي (نهوي كبير) مثل نهوي صغير 0xAA55. على القرص (0x55 عند حيد 510 و 0xAA عند 511)			2	510 - 511	1FE - 1FF
حجم إجمالي			512		

بنية مدخلة 16-بايت في جدول أقسام MBR أو EBR		حجم (بايت)	الإزاحة ضمن المدخلة	
			عشري	ست عشري *
مؤشر الاقلاع	(80h = قسم نشيط، 00h = غير نشيط)	1	0	1 ? E
عنوان CHS : بداية القسم		3	1 – 3	1 ? F – 1 ? 1
شفرة نوع القسم		1	4	1 ? 2
عنوان CHS : نهاية القسم		3	5 – 7	1 ? 3 – 1 ? 5
عنوان LBA : بداية القسم [6]		4	8 – 11	1 ? 6 – 1 ? 9
حجم القسم (بحساب عدد القطاعات)		4	12 – 15	1 ? A – 1 ? D

* للحصول على الرقم الست عشري 1 ؟ E تقرأ 1BE أو 1CE. حيد المدخلة الأولى أو المدخلة الثانية، على التوالي

بنية إحدى سجلات EBR حجم 512 بايت ضمن القسم الممتد

القطاع المنطقي 0 عند العنوان الفيزيائي C0/H1/S1

```
      00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F  0123456789ABCDEF
0000  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  |................|
0010  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  |................|
0020  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  |................|
0030  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  |................|
0040  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  |................|
0050  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  |................|
0060  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  |................|
0070  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  |................|
0080  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  |................|
0090  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  |................|
00a0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  |................|
00b0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  |................|
00c0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  |................|
00d0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  |................|
00e0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  |................|
00f0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  |................|
0100  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  |................|
0110  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  |................|
0120  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  |................|
0130  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  |................|
0140  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  |................|
0150  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  |................|
0160  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  |................|
0170  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  |................|
0180  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  |................|
0190  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  |................|
01a0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  |................|
01b0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 fe  |................|
01c0  ff ff 83 fe ff ff 3f 00 00 00 b4 4b b6 0a 00 fe  |................|
01d0  ff ff 05 fe ff ff f3 4b b6 0a 67 98 ee 11 00 00  |................|
01e0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  |................|
01f0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 55 aa  |..............U.|
      00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F  0123456789ABCDEF
```

حشو بايت صفر (منطقة قد تستغلها بعض التطبيقات!) جدول أقسام ممتد (مدخلتان فقط) توقيع إقلاع

أمثلة

الشكل التالي يعرض قسم ممتد مع 6.000 قطاع و 3 أقسام منطقية.

القسم الممتد يبدأ عند عنوان LBA 5.000 وينتهي عند 10.999 لأن حجمه 6.000 قطاع.

تنبيه: تم اختيار القيم التالية فقط لتسهيل الشرح على القارئ، في الحقيقة، لا يوجد قسم ممتد بحجم 3 مغابايتات ولا قرص ثابت 20 قطاع لكل مسار

عدد القطاعات	قطاع البداية			
1.980 = 1 + 6.999 − 5.020	المدخلة الأولى	**20** = 5.020 − 5.000	سجل إقلاع القسم الممتد الأول	5.000
1.000 = 1 + 7.999 − 7.000	المدخلة الثانية	**2.000** = 7.000 − 5.000		
19 قطاع غير مستخدم				
القسم المنطقي الأول مع **1.980** قطاع				5.020
				6.999

عدد القطاعات	قطاع البداية			
980 = 1 + 7.999 − 7.020	المدخلة الأولى	**20** = 7.020 − 7.000	سجل إقلاع القسم الممتد الثاني	7.000
3.000 = 1 + 10.999 − 8.000	المدخلة الثانية	**3.000** = 8.000 − 5.000		
19 قطاع غير مستخدم				
القسم المنطقي الثاني مع **980** قطاع				7.020
				7.999

عدد القطاعات	قطاع البداية			
2.980 = 1+ 10.999 − 8.020	المدخلة الأولى	**20** = 8.020 − 8.000	سجل إقلاع القسم الممتد الأخير	8.000
0	المدخلة الثانية	**0**		
19 قطاع غير مستخدم				
القسم المنطقي الأخير مع **2.980** قطاع				8.020
				10.999

لمعرفة كيفية الحصول على بيانات EBRs، راجع فقرة: "حساب EBR داخل القسم الممتد"

قيم EBR

المعلومات التالية عبارة عن قواعد عامة تطبق فقط على القيم الموجودة في حقول 4 بايت في مدخلات جدول أقسام سجلات EBRs (قارن الجداول أعلاه).

هذه القيم تعتمد على أداة تقسيم القرص المستخدمة في إنشائها أو تغييرها، في الواقع، معظم أنظمة التشغيل التي تستخدم مخطط التقسيم الممتد (وتشمل مايكروسوفت دوس، ويندوز، ولينكس) ستتجاهل قيمة "حجم القسم" في المدخلات التي تشير إلى قطاع آخر EBR. باستثناء أن في أنظمة تشغيل لينكس. يجب أن تكون القيمة واحد أو أكبر.

المدخلة الأولى (first entry) في جدول أقسام EBR تشير إلى القسم المنطقي (logical partition) الذي ينتمي إلى نفس EBR :

- قطاع البداية (Starting sector) = الإزاحة النسبية (relative offset) بين قطاع EBR هذا والقطاع الأول (first sector) من القسم المنطقي.

ملاحظة: غالبا هذه نفس القيمة لكل EBR على نفس القرص الثابت؛ عادة تكون 63 قطاع في نظام ويندوز أكس بي والأنظمة القديمة.

- عدد القطاعات (Number of sectors) = العدد الإجمالي للقطاعات (total count of sectors) لهذا القسم المنطقي.

ملاحظة: أية قطاعات غير مستخدمة (unused sectors) بين EBR و القسم المنطقي لا تعتبر جزء من القسم المنطقي [1].

المدخلة الثانية (second entry) في جدول أقسام EBR إذا كانت آخر EBR في القسم الممتد ستتضمن بايتات صفرية (zero-bytes)؛ خلاف ذلك، سوف تشير إلى سجل EBR التالي في سلسلة EBR.

- قطاع البداية (Starting sector) = العنوان النسبي (relative address) لسجل EBR التالي ضمن القسم الممتد(extended partition).

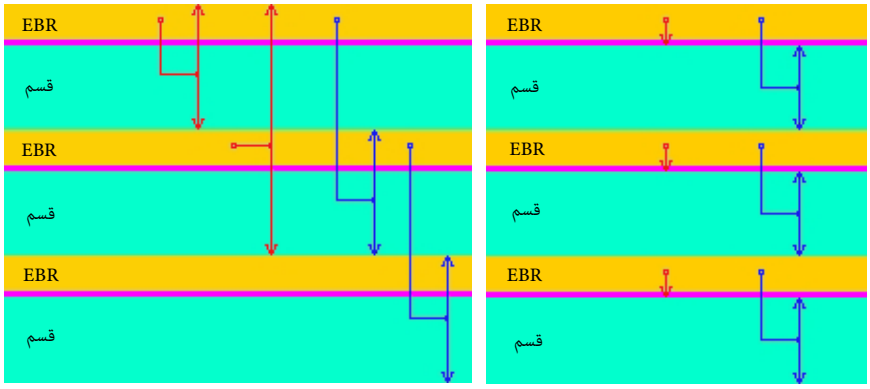
بمعنى آخر: قطاع البداية (Starting sector) = عنوان LBA لسجل EBR التالي ناقص عنوان LBA لأول EBR في القسم الممتد.

- عدد القطاعات (Number of sectors) = العدد الإجمالي للقطاعات (total count of sectors) في القسم المنطقي التالي، لكن التعداد (count) يبدأ من قطاع EBR التالي.

ملاحظة: بخلاف المدخلة الأولى (first entry) في جدول أقسام EBR. حساب عدد القطاعات (number of sectors) هذا يشمل قطاع EBR في القسم المنطقي التالي إلى جانب القطاعات الأخرى في ما يسمى مساره الغير مستخدم (unused track). (قارن تخطيط الشكل 1 و 2 أسفل).

الرسم أدناه ليست مقياس حقيقي للحجم:. الخطوط الرفيعة بالأرجوان بين كل EBR والقسم المنطقي الخاص به تمثل بقية المنطقة الغير مستخدمة، عادة 63 قطاع [2] وتتضمن قطاع EBR واحد (يظهر مضخم).

في بعض الأنظمة، قد تكون هناك فجوة كبيرة غير مستخدمة بين نهاية القسم المنطقي و EBR التالي، أو بين القسم المنطقي الأخير ونهاية كامل القسم الممتد نفسه، في حالة تم حذف/تحجيم أي قسم منطقي نشأ سابقا.



- قيم بداية القطاعات وعددها الإجمالي التي تشير لها المدخلة الأولى
- قيم بداية القطاعات وعددها الإجمالي التي تشير لها المدخلة الثانية

عادة نوع القسم الممتد سيكون 0x05 (عنونة CHS) أو 0x0F (عنونة LBA) [5] نظام دي آر دوس 6.0 DR DOS والنسخ الأحدث تدعم الأقسام الممتدة المؤمنة باستخدام 0xC5، وهي أقسام تكون مخفية عن أنظمة التشغيل الأخرى. بما أن إصدارات DR-DOS حتى الإصدار 7.03 لا تدعم عنونة الكتل المنطقية LBA لا يمكنها التعرف على النوع 0x0F وأنظمة التشغيل الأخرى لا تتعرف على النوع 0xC5، يمكن استغلال هذا لشغل مساحة تصل إلى أول 8 جيجابايت على القرص في DR-DOS (مع الأقراص المنطقية سواء كانت ضمن أقسام مؤمنة أو غير مؤمنة)، واستخدام 0x0F لتخصيص بقية القرص لأنظمة التشغيل التي تدعم LBA دون تعارض بينها. هذا يشبه، فكرة دعم أنظمة لينكس قسم ممتد ثاني نوع 0x85 (هذا النوع مخفي "unknown" عن أنظمة التشغيل التي تدعم فقط قسم ممتد واحد) [6].

أنواع القسم الممتد الأخرى التي يمكنها أن تتضمن سجلات EBRs تشمل: المخفية عن قصد 0x91، 0x9B، 0x1F، 0x15، والمقيدة/حصرية النفاذ 0x5F، 0x5E، والمؤمنة 0xD5، 0xCF لكن، يجب معاملتها وفق أنظمة التشغيل والأدوات التي تدعمها ولا يجب وصلها.

من الصعب تفسير عناوين CHS في القسم بدون معرفة قياسات القرص (الظاهرية)، لأن ترجمة CHS إلى LBA يرتكز على عدد الرؤوس وعدد القطاعات لكل مسار SPT. رغم ذلك عنوان بداية LBA وحجم القسم (بعدد القطاعات) المقدمة إن أمكن تسمح أيضا بحساب قياسات القرص التي تطابق عناوين CHS المقدمة.

عنونة CHS بقيم 24 بت تستخدم دائما 6 بت لقطاعات تصل إلى 63 قطاع لكل مسار (1...63)، و INT 13h (نداء النفاذ للقرص) عموما يستخدم 8 بت من أجل رؤوس تصل إلى 256 رأس (0...255)، هذا يترك 10 بت لأسطوانات تصل إلى 1024 أسطوانة (0...1023).

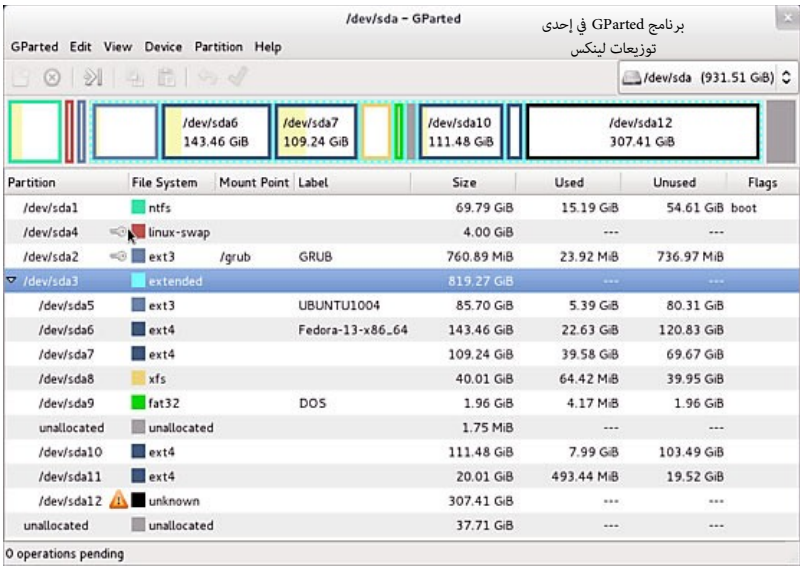
عناوين CHS ATA دائما تستخدم 4 بت من أجل رؤوس تصل إلى 16 رأس (0...15)، هذا يترك 14 بت لأسطوانات تصل إلى 16383 أسطوانة (2^{14}=16383+1) أي (24=6+4+14 بت) [7] هذا في ترجمة عناوين CHS 24 بت مع أقراص ATA-5. (لمعلومات أكثر راجع CHS).

التسمية

لينكس والأنظمة المشابهة تشير إلى القرص الثابت الأول نوع IDE بالشكل /dev/hda، والثاني بالشكل /dev/hdb..إلى آخره. وكما في أقراص SCSI، وفي ألأنويه لاحقا، في أقراص IDE و SATA يعرف القرص الثابت الأول أيضا بالشكل /dev/sda ...إلى آخره. الأقسام الأربعة في MBR ستكون بالشكل /dev/hda4 ,dev/hda3 / /dev/hda2, /dev/hda1. القسم الخامس /dev/hda5، يمثل القرص المنطقي الأول. القسم السادس /dev/hda6 يمثل القرص المنطقي الثاني...إلى آخره، ولا يتم حساب حاوية القسم الممتد الممثل في MBR كأحد الأقسام الأربعة [8]. لمعلومات أكثر راجع الميثاق المستخدم في تسمية الأجهزة في لينكس في الموسوعة الحرة.

حساب EBR (داخل القسم الممتد)

برنامج GParted في إحدى توزيعات لينكس



القرص sda يتضمن 3 أقسام أولية sda1، sda4، sda2 وواحد ممتد sda3/ يظهر بإطار أزرق فاتح يتضمن 8 أقسام منطقية من sda5 إلى sda12..

```
[root@echidna ~]# fdisk -l /dev/sda

Disk /dev/sda: 1000.2 GB, 1000204886016 bytes
255 heads, 63 sectors/track, 121601 cylinders, total 1953525168 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x000de20f

   Device Boot      Start         End      Blocks   Id  System
/dev/sda1   *          63   146368214    73184076    7  HPFS/NTFS/exFAT
/dev/sda2       154754145   156312449      779152+  83  Linux
/dev/sda3       156312450  1874448134   859067842+   5  Extended
/dev/sda4       146368215   154754144     4192965   82  Linux swap / Solaris
/dev/sda5       156312513   336031604    89859546   83  Linux
/dev/sda6       336031668   636880859   150424596   83  Linux
/dev/sda7       636880923   865983824   114551451   83  Linux
/dev/sda8       865983888   949891319    41953716   83  Linux
/dev/sda9       949891383   954003959     2056288+   b  W95 FAT32
/dev/sda10      954007552  1187801087   116896768   83  Linux
/dev/sda11     1187803136  1229760511    20978688   83  Linux
/dev/sda12     1229762560  1874446335   322341888   83  Linux

Partition table entries are not in disk order
```

جدول 9

تظهر رسالة التحذير: "partition table entries are not in disk order" لأن قسم sda4، فيزيائيا قبل sda3. وهذا ليس خطأ مهم، ويمكن إصلاحه،

```
showebr ()
{
   dd if=$1 skip=$2 bs=512 count=1 2> /dev/null | tail -c 66 | hexdump -n 32 -e
'"%07.7_ax " 8/1 "%2.2x " 2/4 " %10d" "\n"'
}
```

هنا تستخدم **dd** و **tail** لعرض المدخلتان الأوليتان من EBR. ثم **hexdump** لعرض قطاع بداية LBA مع حساب القطاع بالعشري و/أو الست عشري

تنبيه:
استخدم بحذر أداة dd؛ لأن الخطأ هنا قد يعني تدمير القطاعات/البيانات على القرص.

أو نفس الخطوات للحصول على معلومات أول EBR عن طريق طرفية لينكس :

1. fdisk -l /dev/sda
2. dd if=/dev/sda skip=<بداية القسم الممتد> of=/home/user/Desktop/first-ebr bs=512 count=1
3. tail -c 66 first-ebr | hexdump -n 32 -e '"%07.7_ax " 8/1 "%2.2x " 2/4 " %10d" "\n"'

أول قطاع في القسم الممتد سيكون أول EBR، سكريبت showebr يعرض ذلك القطاع باستخدام رقم إزاحة القطاع من الجدول 9 :

```
[root@echidna ~]# showebr /dev/sda 156312450
0000000  00 fe ff ff 83 fe ff ff         63  179719092
0000010  00 fe ff ff 05 fe ff ff  179719155  300849255
```

جدول 11

- المدخلة الأولى في جدول أقسام EBR تعلن أن القسم sda5/ يبدأ عند الحيد 63 من بداية EBR هذا. ويتضمن 179719092 قطاع.
- المدخلة الثانية تعلن أن EBR التالي يبدأ عند 179719155 من أول EBR في القسم الممتد. بينما التعداد هو تعداد القطاعات من بداية EBR التالي إلى نهاية القسم المنطقي التالي.
- الجدول 12 أسفل، يعرض معلومات الأقسام المنطقية والممتد على القرص sda/.
- تعداد القطاع للقسم sda6 هو (636880859 -336031668+1=300849192). مع الأخذ بتعداد القطاع 300849255 في المدخلة الثانية (أنظر جدول 11) يمكنك الاستنتاج أن القسم sda6/ يبدأ عند 63 :

أي (63=300849192-300849255) وهذا هو رقم الحيد المستخدم حتى وقت قريب في معظم الأنظمة، بما فيها ويندوز XP.

الأقسام المنطقية والممتد على القرص /dev/sda

dev/sda3	156312450	1874448134	859067842+	5	Extended
/dev/sda5	156312513	336031604	89859546	83	Linux
/dev/sda6	336031668	636880859	150424596	83	Linux
/dev/sda7	636880923	865983824	114551451	83	Linux
/dev/sda8	865983888	949891319	41953716	83	Linux
/dev/sda9	949891383	954003959	2056288+	b	W95 FAT32
/dev/sda10	954007552	1187801087	116896768	83	Linux
/dev/sda11	1187803136	1229760511	20978688	83	Linux
/dev/sda12	1229762560	1874446335	322341888	83	Linux

جدول 12

باستخدام ما تعلمناه للتو، مع سكريبت showebr وحساب المعطى الفعلي في صدفة لينكس، يمكننا حساب كامل سلسلة EBR على القرص sda :

```
[root@echidna ~]# showebr /dev/sda 156312450
0000000  00 fe ff ff 83 fe ff ff           63    179719092
0000010  00 fe ff ff 05 fe ff ff    179719155    300849255
[root@echidna ~]# showebr /dev/sda $(( 156312450 + 179719155 ))
0000000  00 fe ff ff 83 fe ff ff           63    300849192
0000010  00 fe ff ff 05 fe ff ff    480568410    229102965
[root@echidna ~]# showebr /dev/sda $(( 156312450 + 480568410 ))
0000000  00 fe ff ff 83 fe ff ff           63    229102902
0000010  00 fe ff ff 05 fe ff ff    709671375     83907495
[root@echidna ~]# showebr /dev/sda $(( 156312450 + 709671375 ))
0000000  00 fe ff ff 83 fe ff ff           63     83907432
0000010  00 fe ff ff 05 fe ff ff    793578870      4112640
[root@echidna ~]# showebr /dev/sda $(( 156312450 + 793578870 ))
0000000  00 fe ff ff 0b fe ff ff           63      4112577
0000010  00 fe ff ff 05 fe ff ff    797691510    233797128
[root@echidna ~]# showebr /dev/sda $(( 156312450 + 797691510 ))
0000000  00 fe ff ff 83 fe ff ff         3592    233793536
0000010  00 fe ff ff 05 fe ff ff   1031488638     41959424
[root@echidna ~]# showebr /dev/sda $(( 156312450 + 1031488638 ))
0000000  00 fe ff ff 83 fe ff ff         2048     41957376
0000010  00 fe ff ff 05 fe ff ff   1073448062    644685824
[root@echidna ~]# showebr /dev/sda $(( 156312450 + 1073448062 ))
0000000  00 fe ff ff 83 fe ff ff         2048    644683776
0000010  00 00 00 00 00 00 00 00            0            0
```

جدول 13

لحظ أن أحد الأقسام يملك إزاحة من 3592 هذا يقابل تقريبا 1.75 ميغابايت مساحة شاغرة بين sda9 و sda10. (أنظر للصورة GParted)

لحظ أيضا أن القسم sda12 يبدأ عند الحيد 2048 بدلا من 63 . يمكن أن تجد هذا أيضا على الأقراص التي تستخدم قطاعات 2048 بايت أو على أقراص الحالة الصلبة SSD حيث محاذاة القطاعات تكون مهمة جدا [10]

سلامة بيانات القسم الممتد

في حالة حدوث مشكلة في تقسيم القرص، عادة بيانات القسم الممتد تبقى موجودة !، لكن إذا تم إنشاء أقسام منطقية، وكتابة قطاعات جدول الأقسام التي تصفها في بداية هذه الأقسام المنطقية، الأكيد أن البيانات التي كانت هناك ستفقد. في المثال التالي برنامج sfdisk يعرض سلسلة كاملة من الأقسام من ضمنها القسم الممتد نوع 05h.

```
# sfdisk -l -x /dev/hda

Disk /dev/hda: 16 heads, 63 sectors, 33483 cylinders
Units = cylinders of 516096 bytes, blocks of 1024 bytes, counting
from 0

   Device Boot Start     End   #cyls   #blocks   Id  System

/dev/hda1          0+     101     102-    51376+  83  Linux
/dev/hda2        102     2133    2032    1024128   83  Linux
/dev/hda3       2134    33482   31349  15799896    5  Extended
/dev/hda4          0       -       0          0    0  Empty
/dev/hda5       2134+    6197    4064-  2048224+   83  Linux
    -           6198    10261    4064   2048256    5  Extended
    -           2134     2133       0          0    0  Empty
    -           2134     2133       0          0    0  Empty
/dev/hda6       6198+   10261    4064-  2048224+   83  Linux
    -          10262    16357    6096   3072384    5  Extended
    -           6198     6197       0          0    0  Empty
    -           6198     6197       0          0    0  Empty
. . .

/dev/hda10     30581+   33482    2902-  1462576+   83  Linux
    -          30581    30580       0          0    0  Empty
    -          30581    30580       0          0    0  Empty
```

هل يوجد جداول أقسام ببنية سيئة !؟

إنشاء جداول أقسام ببنية سيئة ممكن جدا، مثلا، العديد من ألأنويه ستدخل في حلقة تكرار loop إذا كان هناك قسم ممتد يشير إلى نفسه أو إلى قسم سابق في السلسلة.

إمكانية وجود قسمان ممتدان في أحد قطاعات جدول الأقسام هذه!، وبالتالي تشعب جدول الأقسام.

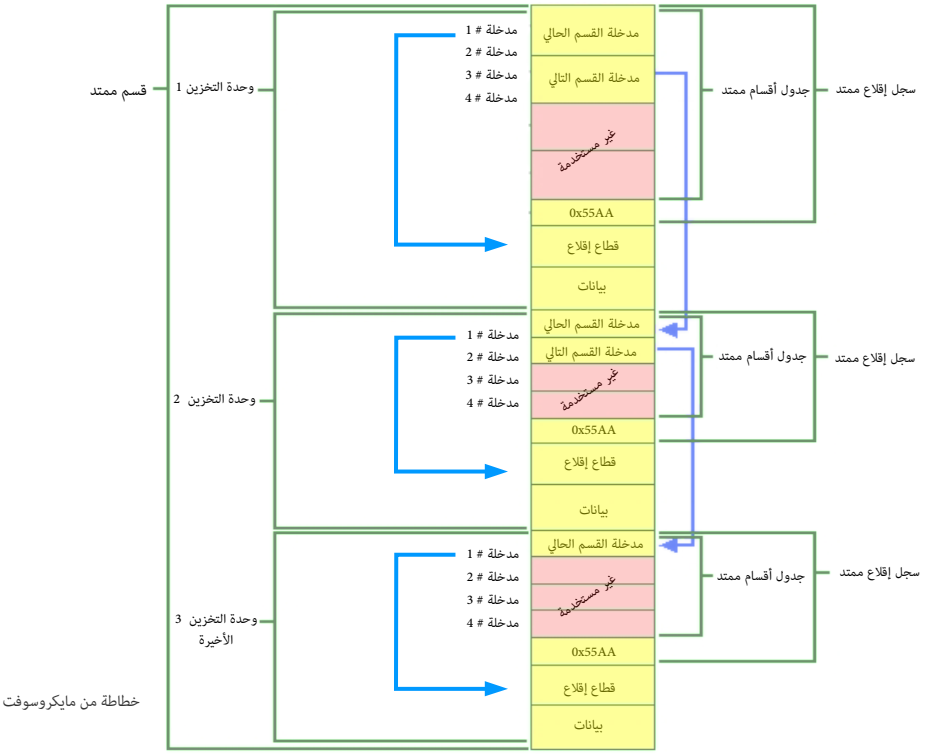
مثلا هذا يمكن أن يحدث مع برنامج fdisk الذي لا يتعرف على نوع القسم الممتد 05h و 0Fh و 85h ، وينشئ النوع 05h إلى جانب 0Fh. في الحقيقة، لا يوجد برنامج معياري fdisk يمكنه التعامل مع هذه الحالة، وسيحتاج المستخدم إلى عمل يدوي لإصلاح ذلك.

نواة لينكس ستقبل التشعب. هذا يعني إمكانية وجود سلسلتان من الأقسام المنطقية [6] أحيانا هذا يكون مفيد، مثلا، يمكن استخدام 05h حتى يتعرف عليه نظام دوس، بينما 85h سيكون مخفي عن دوس، بهذه الطريقة DOS FDISK لن يعلق أو ينهار لأن الأقسام المنطقية ستكون خلف الأسطوانة 1024. في العادة لإنشاء هذا تحتاج إلى أداة مثل sfdisk.

ملاحظات

1. ^ تعبير EPBR يستخدم في برمجيات التشخيص من باور كويست (سيمانتك)، مثل، وسيلة PartitionInfo (في بارتيشن ماجيك)، عند عرض معلومات القسم الممتد. أما ERS استخدم في أتاري TOS
2. ^ أ ب ، EBR يقع في القطاع الأول في ما يسمى المنطقة الغير مستخدمة التي تساوي عدد القطاعات لكل مسار؛ عادة تكون 63 قطاع. في هذا الصدد، كل قسم منطقي يحاكي تخطيط بنية قرص ثابت للقسم الأولى الأول. لأن سجل MBR يقع في القطاع الأول للقرص، القطاع الأول من المسار 0 (عادة يكون متبوع بـ 62 قطاع غير مستخدمة) ثم يأتي قطاع إقلاع القسم الأولى الأول للقرص.
3. ^ لذلك، أي نظام تشغيل أو وسيلة تحتاج إلى النفاذ أو عد جميع الأقراص المنطقية، يجب أن تتبع هذه السلسلة من المدخلات حتى آخر EBR، (الذي سيتضمن مدخلة واحدة فقط).

باستثناء القرص المنطقي الأخير في القسم الممتد، شكل جدول الأقسام الممتد مكرر مع كل قرص منطقي



4. ^ في معظم أنظمة دوس وويندوز السابقة، كان عدد الأقراص المنطقية في القسم الممتد مقيد بـ 23، لأن FDISK [9] لا يستطيع إنشاء أقراص تزيد عن عدد المحارف المتوفرة ؛ إذا كان C: قرص أولي، ستكون المحارف من D: إلى Z: متوفرة فقط من أجل 23 قرص. هذا التقييد غير موجود في FDISK DR-DOS، لأن النظام لا يربط بين الأقسام والمحارف (باستثناء الأقسام الأولية). الوظائف المتقدمة مثل وحدات التخزين المضغوطة أو المؤمنة ودعم تعدد المنصات زاد تعقيد قواعد إسناد محارف الأقراص، وجعلها غير عملية أو حتى مضللة عند ربط المحارف مع أقسام في FDISK . وحقيقة أن DR-DOS FDISK لا ينشأ أقسام فقط ولكن أيضا يهيئ الأقسام النشأة، جعل معرفة المحرف المقابل للقرص بلا أهمية.

نوفيل دوس Novell DOS 7 والنسخ الأحدث رسميا تدعم حتى 32 وحدة تخزين باستخدام توجيهات CONFIG.SYS LASTDRIVE=32 .

في ويندوز أن تي والنسخ اللاحقة. عدد الأقراص المنطقية التي يمكن إنشائها أصبح غير محدود باستخدام امتداد إدارة القرص ؛ لكن عمليا، نادرا ما يحتاج المستخدمين إلى أكثر من 23 قسم، لأن صدفة ويندوز أن تي يمكنها فقط النفاذ إلى الأقراص التي تملك محارف من A: إلى Z:.

النسخ الحديثة من ويندوز تدعم نظام شبيه يونكس في وصل للأقسام إلى مسارات في نظام ملفات آخر، بدلا من استعمال المحارف، وتسمح أيضا بمسارات UNC مثل: \\?\Volume{ uuid}\.

5. ^ الأقسام التي تحذفها إدارة القرص في ويندوز إكس بي يمكن استعادتها بأدوات مثل MiniTool، هذا إذا لم يتم المساس بجداول الأقسام. ويوصى بعمل تفحص CHKDSK للإقسام بعد استرجاعها.
6. ^ ! ب، حقول بداية LBA في مدخلات جدول الأقسام الممتد مرتبطة ببداية القسم الممتد نفسه.

استخدام حقول القطاعات النسبية و المجموع في القسم الممتد يختلف عن استخدامها في الأقسام الأولية. حقل القطاعات النسبية في مدخلة جدول الأقسام الممتد تعرض عدد البايتات من إزاحة بداية القسم الممتد إلى القطاع الأول في القرص المنطقي (يتضمن جدول أقسام ممتد للقرص المنطقي). الرقم الذي في حقل قطاعات المجموع يشير إلى عدد القطاعات التي تشكل القرص المنطقي (من بداية قطاع إقلاع القسم إلى نهاية القرص المنطقي). قيمة حقل قطاعات المجموع يساوي عدد القطاعات من بداية قطاع الإقلاع المعرف من قبل مدخلة جدول الأقسام الممتد إلى نهاية القرص المنطقي.

7. ^ القسم الأولى يستخدم عادة كقسم نظام (خصوصا في ويندوز). لكن إذا لم يكن هناك قسم للنظام، يمكن أن يكون كامل القرص قسم ممتد.
8. ^ بعض حواسيب صانعي القطع الأصلية OEM. تملك قسم تضبيط EISA (البنية الصناعية المعيارية الممتدة) سيكون دائما هو الأول على القرص الثابت.
9. ^ برنامج DOS FDISK يسمح بإنشاء قسم أولي واحد فقط مع قسم ممتد واحد).
10. ^ القسم الممتد لا يرتبط بنظام ملفات ولا يمكن تهيئته، ولا يمكن حذفه حتى يتم حذف كافة الأقراص المنطقية.
11. ^ مدير إقلاع IBM (المضمن في أنظمة OS/2 وبعض النسخ الأولى من Partition Magic)، يضيف مدخلة 9-بايت واحدة على الأقل (بداية من الحيد 0x18A) لكل قطاع EBR. المدخلة مركبة من بايت قيمة علم (يشير لوجود القسم على قائمة مدير إقلاع IBM) متبوعة بسلسلة أسكي 8-بايت تشكل اسم على القائمة. إذا لم يكن القسم في قائمة الاقلاع (هذه البيانات عبارة عن أقسام فقط)، بايت العلم سيكون صفر؛ في هذه الحالة، حقل 8- بايت التالي يمكن أن يتضمن شفرة أسكي تمثل رقم قطاع بداية ذلك القسم (في الست عشري).

مراجع

1. ^ أ ب ت ، " مفاهيم وحل مشاكل القرص" مراجعة فنية للخدمات التطبيقات و ويب في ويندوز 2000 . مايكروسوفت تكنيت عام 2000. جدد في 19-07-2011.
2. ^ "أفضل دليل في تقسيم القرص الثابت" قاعدة معارف تقانة المعلومات بيتري عام 2009. جدد في 19-07-2011.
3. ^ "قسم ويندوز فيستا يمكن أن يختفي إذا استخدم ويندوز إكس بي في إنشاء قسم على حاسوب عليه تنصب كل من ويندوز فستا و إكس بي". موقع مايكروسوفت جدد في 29-05-2015.
4. ^ "حل مشاكل القرص وأنظمة الملفات". عدة موارد ويندوز إكس بي. مايكروسوفت تكنيت. في 03-11-2005. جدد في 15-09-2011.
5. ^ أندريس إيفرت بروير Andries Brouwer (2011) "لائحة بمعرفات الأقسام في الحاسوب الشخصي"، جدد في 19-07-2011. اقتباس:"[0x05] مدعوم في معظم أقراص 8.4 جيجابايت: مع 05 نظام دوس/ويندوز سوف لن يستخدم النداء الممتد في BIOS، حتى وإن كان متوفر. أنظر للنوع 0f أدناه. استخدام 05 لأجل الأقسام الممتدة وراء 8 جيجابايت قد يؤدي إلى تلف في البيانات في م،س دوس".
6. ^ أ ب ، أندريس إيفرت بروير (2004). "الأقسام المنطقية و الممتدة" صفحات Large Disk HOWTO. جدد في 19-07-2011.
7. ^ "ATA-5" (ملف T13/1321D (PDF. واجهات التخزين T10 SCSI اللجنة الفنية INCITS. عام 2000. جدد في 30-07-2011.
8. ^ يورجن هاس Juergen Haas "الأقسام الممتدة".موقع linux.about.com جدد في 19-07-2011.
9. ^ أنتوني Anthony Lissot (20005). "التقسيم باستخدام fdisk". صفحات Linux Partition HOWTO. مراجعة 3.5. جدد في 19-07-2011.
10. ^ وثائق موقع: developerworks

تَنْبِيهٌ

لا توجد أية مصادر عربية في هذه الكتيبات ! باستثناء بعض المصطلحات القليلة من قاموس عرب آنز

احتمال وجود أخطاء في هذا الكتيب وارد. وسواء كان الخطأ من المصدر الانجليزي أو من الترجمة العربية. إذا كنت متخصص أو مدون يمكنك مراجعة ومقارنة الكتيب بالمصدر الانجليزي للترجمة. وتصحيحها في كتابتكم مع الإشارة إلى المصدر أو تصحيحها وإرسالها بالبريد الإلكتروني أو على المدونة

جهاد

في فبراير/شباط 2017

يناير/كانون الثّاني 2019

تمت بحمد الله