

Hamis külső memóriák

PCLinuxOS Magazine – 2020. december

Írta: davecs

a Random Thoughts From My World c. [blogjából](#)

Nagy pénz van a különféle típusú memóriákban. Ezek közé tartoznak az USB kulcsok, az SDXC, valamint mikro SDXC (más néven TF) típusúak. Ám van sok másfajta is. Néhány olcsóbbal sebességgondok vannak, de ha ez nem zavar, akkor használd. Ám ezek között van pár hamis memória meghajtó. Akkor most lássuk, miről van szó.

A hamis memória, egy memória meghajtó, aminek az adatait hamisítják. Egy darabig működni fog. A tárolókapacitásának méretét hamisítják meg. A számítógéped, a telefonod, vagy bármilyen eszköz, amit használasz, a memória elején tárolt információra támaszkodva állapítja meg, mennyi hely van rajta. Ugyancsak itt található a fájlindex rendszer is. Ha valaki képes ezt az információt felülírni, akkor a meghajtó hamis méretadatokat adhat a rendszernek.

Nem létező helyre ír

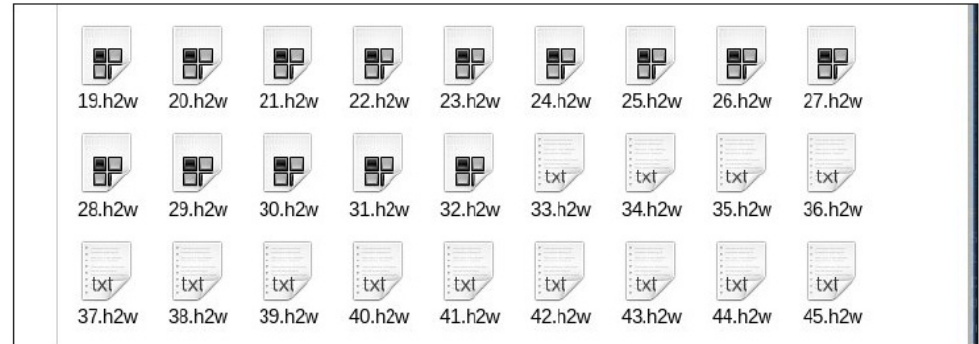
Tegyük fel, hogy vettem, ahogy tényleg történt, egy 128 GB-nek mondott meghajtót, ami valójában csak 32 GB. Amikor pakolok a meghajtóra, az tároló területét az elejétől sorban tölti fel és amíg fel nem tölti teljesen a 32 GB-ig úgy néz ki, mintha működne. Ezután kezdenek a dolgok rosszra fordulni. Legalább 2 dolog van ami történhet ezután, bár esetleg lehetnek továbbiak is. A meghajtóra írt adatok nem létező helyre íródnak, vagy lapozással a létező adatokat írják felül, felhasználva minden helyet 4, vagy 5 eltérő cím alatt. Mindkét eshetőség adatvesztéssel jár.

Az én esetem az első csoportba tartozott. Amikor ellenőriztem az adatokat, az első 31 körüli GB-t rendben olvasta, de minden e fölött elveszett. Ha a másik típusú lett volna, akkor az utolsóként felírt adatot lehetett volna csak visszaolvasni.

A meghajtót egy f3 nevű, egyszerű linuxos programmal teszteltem. Ez egy terminálból futtatható ellenőrző program. Két parancsot ismer, az f3write-ot és az f3read-et. Be kell írni az ellenőrizendő lemez útvonalát. Tegyük fel, hogy a lemez a **/media/3463-3531** útvonalon található (ezt az információt a fájlkezelő mondja meg), akkor ezt a sort írd be a terminálba:

f3write /media/3463-3531

Nagyon sokáig tart, mire a parancs lefut, ám ha a fájlkezelődet a lemez útvonalán hagyod megnyitva, akkor valami ilyesmit fogsz látni:



1.h2w-től kezdve sorban fájlokat ír ki addig, amíg a fájlrendszer nem jelez telítettséget. Láthatod, hogy a 33.h2w-től kezdve a fájl típus módosult, ami azt jelzi, hogy a fájlkezelő azon fájlokat más fájl típusúnak látja. Ez egy figyelmeztetés!

Az írás végeztével a másik f3 parancsot adtam ki terminálban:

f3read /media/3463-3531

Ez egy részlet a terminálban megjelenő szövegből:

```
Validating file 28.h2w ... 2097152/ 0/ 0/ 0
Validating file 29.h2w ... 2097152/ 0/ 0/ 0
Validating file 30.h2w ... 2097152/ 0/ 0/ 0
Validating file 31.h2w ... 2097152/ 0/ 0/ 0
Validating file 32.h2w ... 1941312/ 155840/ 0/ 0
Validating file 33.h2w ... 0/ 2097152/ 0/ 0
Validating file 34.h2w ... 0/ 2097152/ 0/ 0
Validating file 35.h2w ... 0/ 2097152/ 0/ 0
Validating file 36.h2w ... 0/ 2097152/ 0/ 0
Validating file 37.h2w ... 0/ 2097152/ 0/ 0
Validating file 38.h2w ... 0/ 2097152/ 0/ 0
```

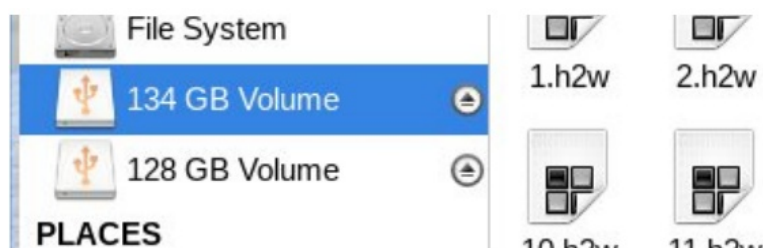
A három pont után az adatok négy oszlopba rendezve jelennek meg. Az elsőben a szám azokat a szektorokat jelzi, amik rendben vannak. A többiek azokat, amik hibásak, megváltoztak, vagy felül lettek írva. Alapvetően bármi, ami nem az első oszlopban szerepel, hibás. Abban az esetben, ha egy sor néhány olyan szektort tartalmaz, amik nem az első oszlopban vannak, de utána minden normális, ez a lemezen sérült területet jelez. Le lehet formázni úgy a lemezt, hogy azok a szektorok ne kerüljenek használatba. Ám a

fenti leíratban a 32.h3w fájl olvasása azt jelzi, hogy a meghajtón 1941312 igazi szektor van és 155840 nem létező és az ezt követő további sor azt mutatja, hogy olyan szektorból próbál olvasni, ami egyszerűen nincs.

Ha az első oszlopban egy kisebb szektorszámot látsz, amit a többiben 0-k követnek és nincs több sor, akkor a terület végére ért. Néha kapsz hibajelzést, próbálva megfeleteni a leállás okát, például hogy a meghajtó esetleg túlmelegedett. Velem előfordult ilyen, amikor egy jó meghajtót teszteltem, miközben a fájlkezelő szerint tele volt, tehát nincs miért aggódni.

Egy másik kulcs

Ez egy elég technikai jellegű bekezdés.



Minden tárolóra igaz „jellemző”, hogy az írásra rendelkezésre álló tényleges terület sokkal kisebb, mint a hirdett méret. A lemezinformáció, a tartalomjegyzék és a formázás maga is elvesz a területből. Az XFCE4 asztalkezelő által használt Thunar egyik tulajdonsága, hogy a kiló, mega és giga byte-okat 1 000-, 1 000 000- és 1 000 000 000-ként jelzi. Az igazi bináris Gigabyte nem 1 000 000 000, hanem 2^{30} , ami 1 073 741 824. A képen két bedugott eszköz látható a Thunar képernyőképén. Az egyik 134 GB, ami az igazi bináris Gigabyte-ban 124,79 GB lenne. Ez ugyan 128 alatt van, de hihetetlenül magas érték. Vagyis, akik a meghajtót manipulálták egy kicsit túllőttek a célon, amikor igazának akarták láttatni a terméküket. A 128 GB-t mutató bináris mérete 119,2 GB, ami sokkal hihetőbb.

Végeredmény

Írtam a hamis meghajtó eladójának Ebay-en keresztül, hogy vegye vissza. Azonnal elfogadta, így nem nyújthatok be panaszt, de az általam megvett meghajtó továbbra is eladó. Tehát, legyünk óvatosak.

The PCLinuxOS Magazine Special Editions!

Get Your Free Copies Today!