Naplózó UFS használata asztali számítógépeken

Kiagias, Manolis <manolis@FreeBSD.org >

Verzió: 43184

Szerzői jog © 2008 Manolis Kiagias

A FreeBSD a FreeBSD Foundation bejegyzett védjegye.

A gyártók és terjesztők által használt megnevezések közül sok védjegy jogot követel. Ahol ilyen megnevezés tűnik fel ebben a dokumentumban, és a FreeBSD Projektnek tudomása volt a védjegyről, a megnevezést a „TM” vagy a „(R)” szimbólum követi.

2013-11-13 írta hrs.

Kivonat

A naplózó állományrendszerek egy napló segítségével rögzítik az összes bennük lezajlott tranzakciót, ezzel igyekszenek megőrizni egy rendszeromlást vagy áramkimaradást követően a rajtuk tárolt adatok épségét. Noha még így is előfordulhat, hogy elveszítjük az adott pillanatban el nem mentett változtatásainkat, a naplózás viszont az állományrendszer számára tökéletes védelmet nyújt a rendszer szabálytalan leállása során keletkező sérülések ellen. Alkalmazása továbbá jelentős mértékben lerövidíti a helyreállításhoz szükséges ellenőrzés idejét. A FreeBSD alaprendszerben megtalálható UFS állományrendszer önmagában nem tartalmaz naplózást, azonban a FreeBSD 7.*X* változataiban már megjelent egy olyan GEOM-osztály, amellyel az állományrendszertől független módon ezt meg tudjuk valósítani. Ebben a cikkben bemutatjuk miként alakítsunk ki UFS alapú naplózást egy hétköznapi asztali számítógépen.

Fordította: Páli Gábor, utolsó ellenőrzés: 2010.11.28.

Tartalom

1. Bevezetés 2
2. Naplózás FreeBSD alatt 3
3. A FreeBSD telepítése során elvégzendő lépések 4
4. A naplózás beállítása 7
5. A naplózás használata során felmerülő hibák kezelése 10
6. Ajánlott olvasmányok 12

1. Bevezetés

Míg az ipari szervereket általában alaposan felkészítik az esetlegesen bekövetkező váratlan leállásokra, addig egy átlagos asztali számítógép teljesen kiszolgáltatott az áramkimaradások, a véletlen újraindítások és minden más olyan beavatkozás számára, amelyek a rendszer nem szabályos leállását eredményezik. A Soft Updates ilyen helyzetekben általában hatékonyan védelmezi az állományrendszert, ez azonban a legtöbb esetben egy hosszadalmasabb háttérbeli ellenőrzést von maga után. Nagyon ritkán viszont az állományrendszer olyan mértékben károsodik, hogy a helyreállításához már felhasználói beavatkozás szükségeltetik és gyakran adatvesztéssel is jár.

A GEOM alrendszerhez nemrég hozzáadott új naplózási képesség az ilyen szituációkban remekül alkalmazható, és ennek köszönhetően szinte egyáltalán nem igényel időt a sérült állományrendszer ellenőrzése, illetve gondoskodik a legutolsó stabil állapot villámgyors visszaállításáról.

Ebben a cikkben bemutatunk egy eljárást, amely segítségével UFS állományrendszerekhez tudunk naplózást beállítani hagyományos asztali számítógépeken (feltételezzük, hogy az operációs rendszer és az adatok egyetlen merevlemezen helyezkednek el). A folyamatot a FreeBSD telepítésén keresztül ismertetjük, és olyan lépesekre bontottuk, hogy lehetőleg kerüljük a bonyolultabb parancssori műveleteket.

A cikk elolvasása során megismerjük:

- a FreeBSD telepítése során hogyan hagyjunk helyet a napló számára;
- hogyan töltsük be és engedélyezzük a `geom_journal` modult (vagy építsük be egy saját rendszermagba);
- hogyan alakítsuk át a már meglevő állományrendszereinket naplózóvá, az `/etc/fstab` állományban milyen beállításokat kell megadnunk a csatlakoztatásukhoz;
- hogyan állítsuk be a naplózást új (üres) partíciókon;
- hogyan oldjuk meg a naplózással kapcsolatban leggyakrabban jelentkező problémákat.

A cikk elolvasásához ajánlott:

- a UNIX® és a FreeBSD alapvető fogalmainak ismerete;
- a FreeBSD telepítés menetének és a sysinstall alkalmazás ismerete.

Figyelem

Az itt megadott eljárás alapvetően egy új rendszer telepítésének esetére vonatkozik, ahol még semmilyen felhasználói adatot nem

tárolunk a lemezen. Természetesen ez a módszer átültethető működő, éles rendszerekre is, azonban ilyenkor mindig készítsünk *biztonsági mentést* mielőtt nekikezdenénk. Ugyanis amikor ilyen alacsony szinten dolgozunk lemezekkel és partíciókkal, bármilyen hiba könnyedén végzetesnek bizonyulhat az adatainkra nézve.

2. Naplózás FreeBSD alatt

A FreeBSD 7.*X* változataiban a GEOM részéről felkínált naplózási lehetőség (eltérően például a Linux® típusú rendszerekben található ext3 állományrendszertől) nem kötődik konkrét állományrendszerhez, de blokkok szintjén üzemel. Habár ez arra utal, hogy különböző állományrendszerek esetén is használható, a FreeBSD 7.0-RELEASE kiadásában még csak az UFS2 felett működik.

Ezt a funkciót a `geom_journal.ko` modul betöltésével (vagy rendszermagba építésével) tudjuk aktiválni, majd a `gjournal` paranccsal érjük el az állományrendszerek konfigurációjához szükséges felületet. Általában nagyobb állományrendszereken, például a `/usr` partíción érdemes engedélyezni a naplózást. Nem szabad elfelejtenünk, hogy ehhez azonban fenn kell tartanunk némi szabad területet a lemezen (erről a következő szakaszban lesz szó).

Amikor egy állományrendszeren bekapcsoljuk a naplózást, magát a naplót is tárolnunk kell valahol a lemezen. A tényleges adatokat tároló lemezterületet *adatterületnek*, míg a naplót tároló területet pedig *naplóterületnek* nevezzük. Ha egy meglevő (nem üres) partícióhoz akarunk naplózást társítani, akkor az adat- és naplóterületeknek külön partíciókon kell lenniük. Amikor viszont egy teljesen új partícióhoz kapcsolunk naplózást, lehetőségünk van egyetlen területen tárolni az adatokat és a naplót. Bármelyik esettel is van dolgunk, a `gjournal` parancs a naplózó állományrendszer véglegesített változatát ezen két fajta terület egyesítéséből hozza létre. Például:

- A `/dev/ad0s1f` eszközön található `/usr` állományrendszeren szeretnénk naplózást használni (amely már eleve tartalmaz hasznos adatokat).
- A partíciók létrehozása során a `/dev/ad0s1g` eszközön lefoglaltunk valamennyi helyet.
- A `gjournal` parancs segítségével készítünk egy `/dev/ad0s1f.journal` eszközt, ahol a `/dev/ad0s1f` eszközön tároljuk az adatokat és a `/dev/ad0s1g` eszközön a naplót. A továbbiakban ezt az új eszközt fogjuk használni.

A napló számára fenntartott hely mennyisége nem az adatok méretétől, hanem az állományrendszer terheltségétől függ. Például egy átlagos irodai számítógép esetén a `/usr` állományrendszerhez nagyjából egy 1 GB méretű naplózási terület remekül megfelel, viszont egy terheltebb rendszer (amellyel például videoanyagok vágását

végezzük) számára ennél több kellhet. A naplóterület idő előtti kimerülése a rendszermag összeomlásával jár.

Megjegyzés

A cikkben javasolt méretek használatával nagyon valószínűtlen, hogy hétköznapi feladataink (böngészés az interneten, szövegszerkesztés, különböző multimédia anyagok lejátszása) közben bármilyen problémát észlelnénk. Ha viszont a lemezünk tartósabb terhelés alatt van, a következő szabály betartásával érhetjük el a legjobb eredményt: a számítógépünkben levő központi memória teljes tartalmának mindig el kell tudnia férni a naplóterület egyharmadán. Tehát például ha a rendszerünk 1 GB memóriával rendelkezik, akkor egy közel 3,3 GB méretű naplóterület ajánlott. (Általánosan: Úgy kapjuk meg a naplóterület méretét, ha megszorozzuk a memória méretet 3,3-mal.)

A naplózásról részleteiben a gjournal(8) man oldalon olvashatunk.

3. A FreeBSD telepítése során elvégzendő lépések

3.1. Lemezterület lefoglalása a naplónak

Az asztali számítógépekben többnyire csupán egyetlen merevlemez található, amelyen maga az operációs rendszer és az adatok helyezkednek el. A sysinstall által felajánlott alapértelmezett partícionálási séma alkalmassága vitatható: egy asztali gép esetén például nincs szükségünk akkora `/var` partícióra, viszont a `/usr` foglalja el a merevlemez legnagyobb részét, hiszen a felhasználók adatai és a rendszerre telepített csomagok ide fognak kerülni.

Az alapértelmezés szerinti felosztás (amely a Disklabel partíciószerkesztőben az A billentyűvel érhető el) nem hagy semennyi lemezterületet szabadon. Ahány partíciót naplózással akarunk ellátni, annyi további partícióra lesz szükségünk a naplókhoz. Mivel a `/usr` lesz közülük a legnagyobb, próbáljuk meg ezen partíció méretének csökkentésével helyet csinálni a naplónak.

A példában most egy 80 GB méretű lemezt láthatunk. Az ábrán most a telepítés közben a Disklabel szerint alapértelmezetten kiosztott partíciókat láthatjuk:

Amennyiben ez körülbelül megfelelő a számunkra, akkor innen már nagyon egyszerű előkészíteni a napló helyét. A nyilak használatával válasszuk ki a `/usr` partíciót és a D billentyű lenyomásával töröljük le.

Ezután válasszuk ki a képernyő felső részében a lemez nevét, majd a C billentyű lenyomásával hozzunk létre egy új partíciót a `/usr` számára. Ez viszont legyen most 1 GB-tal (ha napló csak a `/usr` mellé lesz) vagy 2 GB-tal (ha egyaránt naplózni akarjuk a `/usr` és `/var` partíciókat is) kisebb. A felbukkanó ablakban válasszuk az állományrendszer létrehozását és a `/usr` könyvtárat adjuk meg csatlakozási pontként.

Megjegyzés

Szükségünk van-e naplóra a `/var` partícióhoz? A naplózásnak alapvetően csak óriási méretű partíciók esetében van értelme. Ennek megfelelően nem kell feltétlenül engedélyeznünk a naplózást a `/var` partíción is, habár egy asztali gép esetében ez sosem árthat. Ha ezt az állományrendszert alig használjuk (ami nagyon valószínű egy asztali gépnél), kevesebb területet is rendelhetünk a naplóhoz.

A példánkban a `/usr` és `/var` partíciókhoz is kapcsoltunk naplókat. Természetesen a módszer ezen lépése igény szerint megváltoztatható.

Mivel továbbra sem szeretnénk elbonyolítani a lépéseket, ezért a naplózás bevezetéséhez szükséges partíciók létrehozását szintén a sysinstall segítésével végezzük. A telepítés közben a sysinstall feltétlenül ragaszkodik ahhoz, hogy minden létrehozott partícióhoz

csatlakozási pontot is megadjunk. A naplókat tároló partíciókhoz viszont ilyen nem tartozik, sőt, *egyáltalán nem is kell.* Ezek ugyanis nem olyan hétköznapi partíciók, amelyeket bármikor is csatlakoztatni fogunk.

A sysinstall használata során ezt a problémát úgy tudjuk elkerülni, ha a naplózásnak szánt partíciókat lapozóterületként adjuk meg. A lapozóterületet sem kell soha csatlakoztatni, és a sysinstall ezekből tetszőleges mennyiségűt képes készíteni. A telepítést követő újraindítás után természetesen majd át kell szerkesztenünk az `/etc/fstab` állományban az így létrehozott partíciók jellemzőit.

Lapozóterület kialakításához ismét a nyílbillentyűk használatával navigáljunk a Disklabel alkalmazáshoz tartozó képernyő felső részébe és válasszuk ki a lemez nevét. Ezután nyomjuk le az N billentyűt, majd adjuk meg a kívánt méretet (*1024M*) és a következő menüből válasszuk a „swap space” (lapozóterület) típust. Ismételjük meg az iménti műveletet az összes napló esetén. A példánkban ezen a módon készítettünk egy naplót a `/usr`, és még egyet a `/var` állományrendszer számára. A végeredmény a következő képen látható:

Javasoljuk, ahogy befejeztük a partíciók létrehozását, jegyezzük fel a neveiket és a hozzá tartozó csatlakozási pontokat, így a soron következő konfigurációs lépésekben könnyebben tudunk majd velük dolgozni. Ez egyben segít mérsékelni a telepítést károsító hibák elkövetésének esélyét. A következő táblázatban a példában említett konfigurációhoz vettük fel ezeket az adatokat:

1. táblázat - Partíciók és naplók

Partíció	Csatlakozási pont	Napló
ad0s1d	/var	ad0s1h
ad0s1f	/usr	ad0s1g

Ezután a megszokott módon folytassuk a telepítést. Javasoljuk azonban, hogy a külső alkalmazásokat (csomagokat) addig még ne tegyünk fel a rendszerünkre, amíg teljesen be nem fejeztük a naplózás beállítását.

3.2. A rendszer első indítása

A rendszerünk a szokásos módon fog indulni, de a naplók számára hozzáadott plusz lapozóterületekhez tartozó bejegyzéseket el kell távolítanunk az `/etc/fstab` állományból. A lapozóterületek közül ténylegesen lapozásra általában a „b” (tehát a példánkban az `ad0s1b`) partíciót érdemes meghagyni. Az összes többit egyszerűen töröljük ki, indítsuk újra a rendszerünket és a FreeBSD már nem fogja tovább használni ezeket.

Ahogy a rendszer újra elindul, készen is állunk a naplózás beállítására.

4. A naplózás beállítása

4.1. A `gjournal` futtatása

A naplózást nagyon könnyű lesz beállítani miután már előkészítettük az ehhez szükséges partíciókat. Váltsunk át egyfelhasználós módba, tehát jelentkezzünk be `root` felhasználóként és gépeljük be:

```
# shutdown now
```

Ezután az Enter billentyű lenyomásával megkapjuk az alapértelmezett parancsértelmezőt. Válasszuk le azokat a partíciókat, amelyeken engedélyezni kívánjuk a naplózást. Ezek a példánkban a `/usr` és `/var` partíciók voltak:

```
# umount /usr /var
```

Töltsük be a naplózáshoz szükséges modult:

```
# gjournal load
```

Most pedig a korábbi feljegyzéseink alapján állapítsuk meg melyik naplóhoz melyik partíciót fogjuk rendelni. A példánkban a `/usr` csatlakozási ponthoz az `ad0s1f` eszköz tartozik, és ennek a naplója az `ad0s1g` eszköz lesz, miközben a `/var` ponthoz az `ad0s1d` eszközt rendeltük, és ezt az `ad0s1h` eszközön naplózzuk. Ennek megfelelően a következő parancsokat kell kiadnunk:

```
# gjournal label ad0s1f ad0s1g

GEOM_JOURNAL: Journal 2948326772: ad0s1f contains data.
GEOM_JOURNAL: Journal 2948326772: ad0s1g contains journal.
```

```
# gjournal label ad0s1d ad0s1h

GEOM_JOURNAL: Journal 3193218002: ad0s1d contains data.
GEOM_JOURNAL: Journal 3193218002: ad0s1h contains journal.
```

Megjegyzés

A gjournal hibát fog jelezni, ha bármelyik partíció utolsó szektora már használatban van. Ilyen helyzetekben az -f kapcsoló segítségével felülírásra tudjuk kényszeríteni a parancsot, mint például:

```
# gjournal label -f ad0s1d ad0s1h
```

Mivel most telepítettük a rendszerünket, elég kicsi a valószínűsége, hogy így bármit is ténylegesen felülírnánk.

Létrejött két új eszköz, név szerint az ad0s1d.journal és az ad0s1f.journal . Ezek képviselik azokat a /var és /usr partíciókat, amelyeket valójában csatlakoztatnunk kell. A csatlakoztatásuk előtt azonban állítsuk be hozzájuk a naplózást és tiltsuk le a Soft Updates használatát:

```
# tunefs -J enable -n disable ad0s1d.journal

tunefs: gjournal set
tunefs: soft updates cleared

# tunefs -J enable -n disable ad0s1f.journal

tunefs: gjournal set
tunefs: soft updates cleared
```

Ezt követően parancssorból csatlakoztassuk az új eszközöket a nekik megfelelő pontokra (itt most már használhatjuk az async beállítást):

```
# mount -o async /dev/ad0s1d.journal /var
# mount -o async /dev/ad0s1f.journal /usr
```

Nyissuk meg az /etc/fstab állományt, és az előbbiek szerint javítsuk ki a /usr és a /var állományrendszerekhez tartozó bejegyzéseket:

```
/dev/ad0s1f.journal     /usr            ufs     rw,async      2
  2
/dev/ad0s1d.journal     /var            ufs     rw,async      2
  2
```

Figyelem

Figyelmesen ellenőrizzük a bejegyzéseket, mert ha hibásan adjuk meg ezeket, akkor az újraindítás után gondok lehetnek a rendszer indításával!

Végezetül gondoskodjunk róla, hogy a gjournal(8) modul minden egyes indítás során betöltődjön. Ehhez nyissuk meg a `/boot/loader.conf` állományt és adjuk hozzá a következő sort:

```
geom_journal_load="YES"
```

Gratulálunk, sikeresen beállítottuk a rendszerünkön a naplózást! Innen vagy az **`exit`** begépelésével lépjünk vissza a többfelhasználós módba, vagy egy újraindítással próbáljuk ki a konfiguráció eredményét (mi ezt javasoljuk). A rendszerindítás során a következőhöz hasonló üzeneteket kell majd látnunk:

```
ad0: 76293MB XEC XE800JD-00HBC0 08.02D08 at ata0-master SATA150
GEOM_JOURNAL: Journal 2948326772: ad0s1g contains journal.
GEOM_JOURNAL: Journal 3193218002: ad0s1h contains journal.
GEOM_JOURNAL: Journal 3193218002: ad0s1d contains data.
GEOM_JOURNAL: Journal ad0s1d clean.
GEOM_JOURNAL: Journal 2948326772: ad0s1f contains data.
GEOM_JOURNAL: Journal ad0s1f clean.
```

Szabálytalan leállások esetén az iménti üzenetek némileg változhatnak, például:

```
GEOM_JOURNAL: Journal ad0s1d consistent.
```

Ez általában arra utal, hogy a gjournal(8) a naplóterületen tárolt információk segítségével helyreállította az állományrendszert.

4.2. A naplózás engedélyezése frissen létrehozott partíciókon

Míg az előbbiekben tárgyalt megoldást leginkább olyan partíciók esetén alkalmazhatjuk, amelyek már eleve tartalmaznak adatokat, addig egy újonnan létrehozott partíciót némileg könnyebb naplózással ellátni, mivel ilyenkor az adat- és a naplóterület egyazon partíción is kialakítható. Például most tegyük fel, hogy hozzáadtunk egy újabb lemezt a rendszerünkhöz, amelyen készítettünk egy új `/dev/ad1s1d` nevű partíciót. A napló létrehozása ekkor csupán ennyi:

```
# gjournal label ad1s1d
```

A napló mérete alapértelmezés szerint 1 GB lesz, amelyet viszont a `-s` opció használatával tetszés szerint átállíthatunk. Az értéket megadhatjuk byte-okban, vagy a `K`, `M`, illetve `G`

hozzáfűzésével kilobyte-okban, megabyte-okban, illetve gigabyte-okban is. Arra azonban figyeljünk, hogy a `gjournal` nem enged túlságosan kis méretű naplót létrehozni.

Például egy 2 GB méretű napló az alábbi paranccsal hozható létre:

```
# gjournal label -s 2G ad1s1d
```

Mellé hozzunk létre egy állományrendszert az új partíción, ahol a `-J` kapcsolóval engedélyezzük a naplózást:

```
# newfs -J /dev/ad1s1d.journal
```

4.3. A naplózás támogatásának beépítése a rendszermagba

Amennyiben nem kívánjuk betölteni a `geom_journal` modult, lehetőségünk van közvetlenül a rendszermagba beépíteni a hozzá tartozó funkcionalitást. Ehhez nyissunk meg (vagy hozzunk létre) egy saját rendszermag-konfigurációs állományt, és vegyük fel benne a következő két sort:

```
options UFS_GJOURNAL # Megjegyzés: Ez része a GENERIC rendszermagnak
options GEOM_JOURNAL # Ezt se felejtsük ki
```

A FreeBSD kézikönyvben szereplő utasítások mentén fordítsuk le és telepítsük az új rendszermagot.

Ha korábban használtuk volna a modult, akkor ezzel együtt ne felejtsük el kivenni a `/boot/loader.conf` állományból sem a hozzá tartozó sort.

5. A naplózás használata során felmerülő hibák kezelése

Ebben a szakaszban a naplózás alkalmazásakor jelentkező gondokra vonatkozó gyakran ismételt kérdéseket foglaljuk össze.

K: A rendszer folyamatosan összeomlik komolyabb lemezterhelés mellett. Van ennek valamilyen köze a naplózáshoz?

V: A napló ilyenkor valószínűleg gyorsabban betelik, mint ahogy kiíródhatna a lemezre. Nem szabad elfeledkeznünk róla, hogy a napló méretének sosem az adatterület méretével kell arányosnak lennie, hanem a lemez terheltségével. Ha tehát a lemezeink nagyobb terhelés alatt vannak, akkor egy nagyobb területet kell hozzárendelnünk a naplóhoz. Ezzel kapcsolatban lásd a Naplózás FreeBSD alatt című szakaszt.

K: Valamit nem sikerült rendesen beállítani a konfiguráció során, ezért most nem indul a rendszer. Meg lehet valahogy javítani?

V: Ilyenkor vagy elfelejtettük (vagy netalán elírtuk) a `/boot/loader.conf` állományban szükséges bejegyzést, vagy az `/etc/fstab` állományunk hibákat tartalmaz. Az ilyen jellegű problémákat viszonylag könnyű helyrehozni. Az Enter billentyű lenyomásával hozzuk elő az egyfelhasználós módhoz tartozó parancsértelmezőt. Ha ez sikerült, akkor kutassuk fel a probléma okát:

```
# cat /boot/loader.conf
```

Ha innen hiányzik vagy nem helyesen szerepel a `geom_journal_load` bejegyzés, akkor a naplózás használatához szükséges eszközök nem fognak létrejönni. Töltsük be a modult manuálisan, csatlakoztassuk az összes partíciót és folytassuk a többfelhasználós mód indítását:

```
# gjournal load

GEOM_JOURNAL: Journal 2948326772: ad0s1g contains journal.
GEOM_JOURNAL: Journal 3193218002: ad0s1h contains journal.
GEOM_JOURNAL: Journal 3193218002: ad0s1d contains data.
GEOM_JOURNAL: Journal ad0s1d clean.
GEOM_JOURNAL: Journal 2948326772: ad0s1f contains data.
GEOM_JOURNAL: Journal ad0s1f clean.

# mount -a
# exit
(a rendszerindítás folytatódik)
```

Ha viszont ezzel a bejegyzéssel kapcsolatban semmilyen hibát nem találtunk, akkor nézzük meg az `/etc/fstab` állományt. Akkor valószínűleg itt fogunk találni egy hibásan beírt vagy éppen hiányzó bejegyzést. Amennyiben erről lenne szó, csatlakoztassuk kézzel a fennmaradó partíciókat és folytassuk a többfelhasználós mód indítását.

K: Visszavonható a naplózás, vissza lehet valahogy térni a Soft Updates használatához?

V: Hogyne. A most következő módszer segítségével megfordítható az egész folyamat. Miután végeztünk, a naplózás részére korábban kialakított partíciók tetszés szerint felhasználhatóak.

Jelentkezzük be `root` felhasználóként és váltsunk egyfelhasználós módba:

```
# shutdown now
```

Válasszuk le a naplózást alkalmazó partíciókat:

```
# umount /usr /var
```

Írassuk ki lemezre a naplók tartalmát:

```
# gjournal sync
```

Állítsuk le a naplózóterületek használatát:

```
# gjournal stop ad0s1d.journal
# gjournal stop ad0s1f.journal
```

Töröljük le az eszközökön tárolt összes naplózási metainformációt:

```
# gjournal clear ad0s1d
# gjournal clear ad0s1f
# gjournal clear ad0s1g
# gjournal clear ad0s1h
```

Tiltsuk le az állományrendszer naplózását és állítsuk vissza a Soft Updates használatát:

```
# tunefs -J disable -n enable ad0s1d

tunefs: gjournal cleared
tunefs: soft updates set

# tunefs -J disable -n enable ad0s1f

tunefs: gjournal cleared
tunefs: soft updates set
```

Manuálisan csatlakoztassuk újra a régi eszközöket:

```
# mount -o rw /dev/ad0s1d /var
# mount -o rw /dev/ad0s1f /usr
```

Az `/etc/fstab` állományban állítsunk vissza mindent az eredeti állapotába:

```
/dev/ad0s1f     /usr            ufs     rw      2       2
/dev/ad0s1d     /var            ufs     rw      2       2
```

Végül a `/boot/loader.conf` állományból távolítsuk el a `geom_journal` modul betöltésére vonatkozó bejegyzést és indítsuk újra a rendszert.

6. Ajánlott olvasmányok

A naplózás még viszonylag újdonságnak számít a FreeBSD esetében, ezért nem feltétlenül találunk róla túlságosan sok dokumentációt. Ettől függetlenül azonban a következő források elolvasása azért hasznosnak bizonyulhat:

- A FreeBSD kézikönyv naplózással foglalkozó szakasza.

- Pawel Jakub Dawidek, a gjournal(8) fejlesztőjének a freebsd-current levelezési listára küldött levele.
- Ivan Voras a freebsd-questions levelezési listára küldött levele.
- A gjournal(8) és geom(8) man oldalak.