

# Mérési jegyzőkönyv

## Háttértár rendszerek vizsgálata

A feladatokat összeállította: Tóth Dániel, Huszerl Gábor, Szatmári Zoltán, BME MIT, 2008-2012.  
A feladatkiírást átdolgozta: Huszerl Gábor, BME MIT, 2013.

### Tudnivalók:

- A feladatok egy részét csak az ötös jegyhez kell megoldani, ezeket \*-gal jelöltük.
- A mérést mérőpárokban kell végezniük, a beadáson mindkettőjüknek meg kell jelennie.
- A tanszéki oktatási cloud használatához olvassák el a portálon található tájékoztatót. Az ott említett VPN kulcsok és konfigurációs állományok (bejelentkezés után) letölthetők a portálunk feladatkiadási és jegyzőkönyv-leadási felületén a MIT2(!) méréssel közös oldalon.
- A mérés utolsó feladataihoz kettő virtuális gépre lesz szükségük. (Azonos virtuálisgép-sablonból, image-ből.)
- A virtuális gép(ek)en végzett minden munkájuk elvesz kilépéskor illetve legkésőbb a foglalásuk lejártakor. A hosszabb távon szükséges fájljaikat (pl. az elkészült szkripteket) mentsek le időben.
- A konzultáción a megjelenés nem kötelező, viszont ez az a szervezett alkalom, ahol a feladatokkal kapcsolatos kérdéseiket feltehetik. Ehhez természetesen az szükséges, hogy legalább megpróbálják a feladatot a konzultáció előtt elvégezni.
- A konzultációk során felmerülő közérdekű kérdésekre a feladatkiírás oldaláról letölthető dokumentumban (Kérdések-válaszok) adunk tájékoztatást. Ezt a dokumentumot verziószámmal is azonosítjuk. A feladatok elvégzése előtt ellenőrizze ennek a dokumentumnak a tartalmát is. (Megkönnyítheti ezzel a saját munkáját.)
- A beadásra hozzák magukkal
  - a **sárga** részekben leírt feladatokat teljesítő elkészült szkripteket (elektronikus formában)
  - az **ibolyaszínű** részekbe írt kérdésekre a válaszaikat (fejben)
- Beadáskor vizsgáljuk, hogy a szkriptek helyesen elvégzik-e az alábbi feladatokat, és hogy a beadó hallgatók tisztában vannak-e a szkriptek működésével, a mérés eredményeivel.

A mérés célja, hogy a gyakorlatban kipróbálható legyen néhány korszerű háttértár-kezelési technológia, illetve ezek alapvető tulajdonságai:

- Hibatűrő működés megvalósítása szoftveres RAID tömbökkel
- A hibatűrési képességek illetve helyreállítás menetének tesztelése hibainjektálással
- Dinamikus tárkiosztás logikai kötetekkel
- Pillanatkép készítése háttértárról, gyakorlati hasznának vizsgálata
- Egyszerű tárhálózat kiépítése iSCSI segítségével
- Hálózati fájlmegosztás NFS segítségével

A következő feladatokat a `vcl.inf.mit.bme.hu` címen elérhető virtuális laboratóriumban elérhető virtuális gépbe telepített Linux környezetben kell elvégezni, mely tartalmazza a szükséges eszközöket. A virtuális laboratórium előzetes időpontfoglalással használható bármely nap bármely órájában. (Ne felejtse el időben időpontot foglalni magának, a rendelkezésre álló helyek beteltele után nem fogja tudni a mérést elvégezni, pótmérésre kell jönnie.) A rendelkezésre álló összes helyek száma a hallgatók számánál lényegesen nagyobb, de egy időpontban egyszerre (pl. a beadás előtti este) csak a hallgatók egy kis része tudja a mérést végezni. A virtuális laborral kapcsolatos általános információk az alábbi helyeken érhetőek el:

```
https://www.inf.mit.bme.hu/content/oktatasi-cloud
```

A szükséges VPN kapcsolat kialakítása után jelentkezünk be (SSH, PuTTY, ...) a kapott virtuális gépre. Az SSH kapcsolat felépítéséhez először a virtuális labor oldalán meg kell állapítani a virtuális gép IP címét. Ezután egy tetszőleges számítógép grafikus felületén indított terminal ablakban kell kiadni a következő parancsot:

```
ssh meres@<IP>
```

A felhasználó nevét és jelszavát az oktatási cloud foglalási portálja adja meg a lefoglalt virtuális gép létrejötte után. Vizsgáljuk meg az alapvető paramétereit.

A virtuális géphez összesen 7 virtuális SCSI merevlemez eszköz van hozzárendelve:

- `/dev/sda` – az operációs rendszer fájljait tartalmazza, ezen ne végezzünk semmilyen műveletet!
- `/dev/loop0`, ..., `/dev/loop5` – 256-256MB-os üres lemezek a feladatok elvégzéséhez

A virtuális gépen nincs grafikus kezelőfelület telepítve. A parancssori felületet el lehet érni SSH-val távoli belépéssel. A `meres` felhasználó a `sudo` mechanizmuson keresztül képes parancsokat a `root` felhasználó nevében futtatni.

## 1. feladat: Mérési infrastruktúra elindítása, áttekintése

### 1.1

Indítsuk el az StorageMeres virtuális gépet, és jelentkezünk be rá a virtuális labor által megadott felhasználóként. Vizsgáljuk meg az elérhető lemezeszközöket. Milyen eszközöket látunk?

1. szkript:

1. Írja ki a mérőpár két tagjának nevét.
2. Listázza ki a gépen elérhető lemezeszközöket. (Csak azokat, de azt mindet.)

Milyen logikai kötetek léteznek a rendszerben?

1. szkript:

3. Listázza ki a gépen elérhető logikai köteteket.

Milyen hálózati beállításokkal dolgozunk?

1. szkript:

4. Listázza ki a gépen elérhető hálózati csatolók beállításait.

## 2. feladat: Szoftveres RAID Linux alatt

### 2.1 RAID0 tömb tesztelése

Építsünk fel egy RAID0 tömböt megfelelő számú diszkkal! Nézzük meg a tömb illetve az azt alkotó eszközök lekérdezhető paramétereit! (A tömb és az egyes eszközök mérete, állapota, szerepe, egyéb jellemzői is láthatóak legyenek.)

2. szkript:

1. Írja ki a mérőpár két tagjának nevét.
2. Építse fel a RAID0 tömböt.
3. Írja ki a tömb és az azt alkotó eszközök lekérdezhető paramétereit.

Adjunk hozzá egy tartalék diszket! Nézzük meg a tömb és a hozzáadott eszköz lekérdezhető paramétereit! Teszteljük a tömb tartalmának olvashatóságát (pl. a `/dev/null`-ba irányítva az olvasás kimenetét)

2. szkript:

4. Adjon hozzá a tömbhöz egy tartalék diszket.
5. Írja ki a tömb és az azt alkotó eszközök lekérdezhető paramétereit.
6. Tesztelje a tömb tartalmának olvashatóságát.

Értelmezze a tapasztaltakat.

Jelöljük hibásnak az egyik lemezt az `mdadm` segítségével! Vizsgáljuk meg, mit mutatnak most a tömb állapotát jelző különféle interfészek! Kísérreljük meg ismét felolvasni a tömb tartalmát!

2. szkript:

7. Jelölje hibásnak az egyik lemezt.
8. Írja ki a tömb és az azt alkotó eszközök lekérdezhető paramétereit.
9. Tesztelje a tömb tartalmának olvashatóságát.

Értelmezze a tapasztaltakat.

Állítsuk le a tömböt!

2. szkript:

10. Állítsa le a tömböt.

Mit tapasztalunk, az `mdadm` hibásnak jelölő funkciója valódi hibát injektál?

## 2.2 RAID1, RAID5 vagy RAID6 tesztelése

Építsük fel a RAID tömböt megfelelő számú diszkkal valamelyik redundancia típussal!

(A fejezetcímben szereplő típusok közül tetszőlegesen választható.)

Nézzük meg a tömb illetve az azt alkotó eszközök lekérdezhető paramétereit! (A tömb és az egyes eszközök mérete, állapota, szerepe, egyéb jellemzői is láthatóak legyenek.)

3. szkript:

1. Írja ki a mérőpár két tagjának nevét.
2. Építse fel a RAID tömböt.
3. Írja ki a tömb és az azt alkotó eszközök lekérdezhető paramétereit.

Adjunk hozzá egy tartalék diszket! Nézzük meg a tömb és a hozzáadott eszköz lekérdezhető paramétereit! Teszteljük a tömb tartalmának olvashatóságát (pl. a /dev/null-ba irányítva az olvasás kimenetét)

3. szkript:

4. Adjon hozzá a tömbhöz egy tartalék diszket.
5. Írja ki a tömb és az azt alkotó eszközök lekérdezhető paramétereit.
6. Tesztelje a tömb tartalmának olvashatóságát.

Értelmezze a tapasztaltakat.

Jelöljük hibásnak az egyik lemezt az mdadm segítségével! Vizsgáljuk meg, mit mutatnak most a tömb állapotát jelző különféle interfészek! Kísérreljük meg ismét felolvasni a tömb tartalmát!

3. szkript:

7. Jelölje hibásnak az egyik lemezt.
8. Írja ki a tömb és az azt alkotó eszközök lekérdezhető paramétereit.
9. Tesztelje a tömb tartalmának olvashatóságát.

Értelmezze a tapasztaltakat.

Állítsuk le a tömböt!

3. szkript:

10. Állítsa le a tömböt.

Mit tapasztalunk, az mdadm hibásnak jelölő funkciója valódi hibát injektál?

### 3. feladat: Logikai kötetkezelés LVM2-vel

#### 3.1 LVM

Hozzunk létre logikai kötetet (természetesen előtte fizikai kötetet és kötetcsoportot) valamelyik blokkos eszközre! Ez történhet akár RAID tömbre is, de maradjon még eszköz szabadon. Hozzunk létre fájlrendszert a logikai köteten, és csatoljuk fel! Másoljunk rá adatot.

4. szkript:

1. Írja ki a mérőpár két tagjának nevét.
2. Hozza létre a logikai kötetet.
3. Listázza ki a gépen elérhető fizikai köteteket, kötetcsoportokat és logikai köteteket.
4. Hozzon létre fájlrendszert a köteten, és csatolja azt fel.
5. Másoljon tetszőleges adatot a felcsatolt logikai eszközre.

Ha a logikai kötetünk később kicsinek bizonyul, akkor a méretét meg tudjuk növelni anélkül, hogy a futó rendszer életébe komolyabban bele kellene avatkoznunk. Próbálják ki a technológia ezen fontos képességét is!

4. szkript:

6. Adjon új fizikai kötetet a kötetcsoporthoz.
7. Növelje meg a logikai kötetet úgy, hogy nagyobb legyen az első fizikai kötet méreténél!
8. Listázza ki a gépen elérhető fizikai köteteket, kötetcsoportokat és logikai köteteket.
9. Növelje meg a fájlrendszert, hogy kitöltse a kötet új méretét.
10. Mutassa meg a művelet eredményét.

Nézzük meg, hogy a megnövelt logikai kötet allokációs egységei fizikailag hová esnek!

4. szkript:

11. Írja ki a logikai kötet részletes adatait, látszódjék az allokációs egységek fizikai elhelyezkedése is.

#### 3.2 LVM Snapshot

Hozzunk létre egy pillanatképet a logikai kötetről! Ha nem lenne elég hely hozzá, mindig lehet bővíteni újabb fizikai kötetek hozzáadásával! Módosítsuk a fájlrendszer tartalmát, akár valamilyen nagyobb fájl létrehozásával is! Vizsgáljuk meg az eredeti és a pillanatkép kötet foglaltságának változását!

4. szkript:

12. Hozzon létre egy pillanatképet a logikai kötetről.

13. Listázza ki a gépen elérhető logikai köteteket.
14. Módosítsa a fájlrendszer tartalmát.
15. Listázza ki a gépen elérhető logikai köteteket.

A pillanatkép kötet foglaltsági jelzése hogyan változik? Mit jelent ez, mi történik a fájlrendszer módosításakor?

Csatoljuk fel a pillanatképet egy másik könyvtárba! Mit találunk?

4. szkript:

16. Csatolja fel a pillanatkép kötetet egy könyvtárba.
17. Listázza ki a felcsatolt kötet tartalmát.

Mi látszik a pillanatkép kötet felcsatolásával? Miért? Mire használható ez?

### 3.3\* Bónusz feladat

Hozzunk létre egy olyan elrendezést, hogy az alap fájlrendszer logikai kötetek egy fizikai eszközön legyenek találhatóak (hogyan hozható ilyen létre?), a pillanatkép logikai kötetek pedig egy másik fizikai eszközre kerüljenek!

5. szkript:

1. Írja ki a mérőpár két tagjának nevét.
2. Hozzon létre egy logikai kötetet, felcsatolás után hozzon létre rajta tartalmat.
3. Készítsen a logikai kötetéről pillanatképet úgy, hogy az a másik fizikai kötetre kerüljön.
4. Írja ki a logikai kötet részletes adatait, látszódjék az allokációs egységek fizikai elhelyezkedése is.

Próbáljuk meg a kernel hibainjektorral elérhetetlenné tenni azt a fizikai kötetet, amin az első (nem pillanatkép) logikai kötet található. Mi történik ilyenkor a pillanatképpel?

5. szkript:

5. A kernel hibainjektor segítségével tegye elérhetetlenné az eredeti logikai kötetet tartalmazó fizikai kötetet.
6. Listázza ki a pillanatkép tartalmát, tesztelje a pillanatkép tartalmának olvashatóságát.

Értelmezze a tapasztaltakat.

## 4. feladat: iSCSI tárhálózat kiépítése

A mérési környezetben az iscsi target init scriptje (indító, leállító) a következő:

```
/etc/init.d/iscsitarget
```

Az initiator init scriptje pedig itt található:

```
/etc/init.d/open-iscsi
```

A segédletben ismertetettől kicsit eltérő módon ebben a Linux disztribúcióban az *iscsi-target* a konfigurációs fájlját nem a */etc/ietd.conf* alatt keresi, hanem a */etc/iet/ietd.conf* elérési úton.

Ettől a feladattól két virtuális gépet kell használniuk. A scriptek A és B részét úgy készítsék el, hogy azok a megfelelő sorrendben a két gépen külön-külön lefuttathatóak legyenek. (A szkriptek használhatnak parancssori bemeneteket pl. a másik gép IP-címének megadásához. Számíthat arra, hogy a szkript futtatásakor a szükséges paramétereket a felhasználó paraméterként megadja.) A szkripteket csak párban futtatjuk, tehát pl. a megosztások nevének nem kell szabadon paramétrezhetőnek lennie.

### 4.1 iSCSI

Válasszunk ki egy eszközt, amit ki akarunk ajánlani! Ez lehet RAID tömb vagy logikai kötet is. Konfiguráljuk fel a *target*-et, és indítsuk el!

Hozunk létre egy fájlrendszert rajta, és másoljunk rá valamit! Ha már egy korábbi feladtból van ilyen kéznél, akkor nyugodtan használjuk fel azt.

6/A. szkript:

1. Írja ki a mérőpár két tagjának nevét.
2. Írja ki a gép hálózati interfészeinek alapvető paramétereit.
3. Hozzon létre egy iSCSI célpontot, és indítsa el.
4. Másoljon tartalmat a kijánlott könyvtárba.
5. Listázza ki a kijánlott könyvtár tartalmát.

Indítsunk egy második virtuális gépet. Indítsuk el az *initiator*-t a második gépen!

A konfigurált célpontra kapcsolódva a kernel log végén meg kell jelennie a bejegyzésnek egy új SCSI eszközzel.

6/B. szkript:

1. Írja ki a mérőpár két tagjának nevét.
2. Listázza ki a gépen elérhető SCSI eszközöket.
3. Indítsa el az iSCSI klienst a gépen.
4. Kapcsolódjék rá a másik gépen kijánlott célpontra.
5. Listázza ki a gépen elérhető SCSI eszközöket.



Csatoljuk fel az újonnan létrejövő eszközt, és nézzünk bele!

6/B. szkript:

6. Csatolja fel az új SCSI eszközt.
7. Listázza ki a felcsatolt eszköz tartalmát.

Hogyan tudja megnézni, hogy műveletvégzés közben melyik virtuális gép diszkje aktív?

## 4.2\* Bónusz feladat

Készítsük teljes másolatot az iscsi célpontot futtató virtuális gép gyökér fájlrendszeréről (RootVG kötetcsoport RootVol eszközeiről) a másik virtuális gépre!

Írja meg a 7/A és 7/B szkripteket megfelelően.

Mi kell ahhoz, hogy ez a másolás sikerüljön, és a fájlrendszer a másik gépen felcsatolható állapotban elérhetővé váljék?

## 5. feladat: NFS fájlmegosztás megvalósítása

### 5.1 NFS megosztás

Az eddigi első virtuális gépen válasszunk ki egy könyvtárat, amit hálózaton keresztül elérhetővé szeretnénk tenni! Állítsuk be az NFS szerveret úgy, hogy csak az adott számítógép hálózatából legyen elérhető. („/8” alhálózati maszk elegendő.) Ossa meg a könyvtárat.

8/A. szkript:

1. Írja ki a mérőpár két tagjának nevét.
2. Írja ki a gép hálózati interfészeinek alapvető paramétereit.
3. Hozzon létre egy NFS megosztást, és tegye elérhetővé a helyi hálózaton belülről.
4. Másoljon tartalmat a kijánlott könyvtárba.
5. Listázza ki a kijánlott könyvtár tartalmát.

A második virtuális gépről csatoljuk fel a kijánlott könyvtárat! Ellenőrizzük a tartalmát!

8/B. szkript:

1. Írja ki a mérőpár két tagjának nevét.
2. Csatolja fel a másik gépen megosztott könyvtárat.
3. Listázza ki a felcsatolt könyvtár tartalmát.

### 5.2\*\* Opcionális feladat

Hasonlítsuk össze írási és olvasási teljesítmény szempontjából a helyi lemezeket, az iSCSI technológiával felcsatolt távoli lemezeket és az NFS-sel felcsatolt könyvtárakat. A mérésnek virtuális környezetben nincs sok értelme, inkább az elv, a megvalósítás érdekes a mérés során.

Hogyan definiál benchmarkot ehhez a méréshez?

A 9. szkript végezze el mérendő eszközök létrehozását, végezze el a mérést, és mutassa meg az eredményeket.

Milyen eredményeket kapott, és ezek hogyan értelmezhetőek?