FreeBSD jails rendszer használata, beüzemelése

készítette: Janky Ferenc Nándor (OA8AT9)

Tartalomjegyzék

1) Bevezetés

A FreeBSD jails egy operációs rendszer szintű virtualizációt megvalósító eszköz, amely a FreeBSD 4-es verziója óta elérhető.

Az OS szintű virtualizáció a mi választásunk, ha megfelel, hogy a guest gépek és a host közös kernelen osztozzon, így nem kell virtuális hardvert biztosítanunk, elég ha az alkalmazások szintjén tudunk virtuális környezetet biztosítani.

A jails eredete, hogy a chroot alkalmazást akarták kiváltani, mert az nem volt kellően rugalmas, csak a fájlrendszer szintjén biztosította a szegmentálást, de ki lehetett jutni a chroot-olt alkalmazásokból. Továbbá a chroot-olt processzek számára továbbra is láthatóak maradtak a rendszer komponensei (IPC, felhasználók, kernel adatszerkezetek, device-ok).

A jail-ben futó processzek nem tudnak interakcióba lépni a jail-en kívűli processzekkel (Ez nem teljesen igaz, mert a shared memory elkülönítése még nem megvalósított), továbbá tilos a globális rendszerállapot megváltoztatása a jail-en belülről, még az ottani rootnak is.

Egy jail-t négy elem határoz meg:

– egy könyvtárfa (lásd chroot)
– hostnév
– IP cím (a jail élettartama során nem lehet megváltoztatni)
– parancs, amit a jailben futtatni szeretnénk

Két típusát különböztetik meg:

– complete jail (egy teljes disztribúció, akár linux alapú is lehet)
– service jail (egy alkalmazás, vagy szolgáltatás futtatása)

2) Jail létrehozása

A létrehozás folyamata:

– setenv D /a/jail/konyvtara

– mkdir -p $D

– cd /usr/src

– make buildworld (a felhasználói állományok fordítása,csak ha még nem volt, vagy update esetén kell futtatni)

– make installworld DESTDIR=$D (A célkönyvtár feltöltése a binárisokkal,man page-ekkel stb)

– make distribution DESTDIR=$D (a /usr/src/etc másolása a $D/etc -be)

Ha szükséges, akkor a devfs-t is felcsatolhatjuk a $D/dev alá

– mount -t devfs devfs $D/dev (ha szeretnénk, hogy reboot után is mountolódjon, akkor fstab-ba is be kell írni), a devfs-hez való hozzáférést a devfs.conf és a devfs.rules fájlokkal manipulálhatjuk, ahol az egyes eszközökhöz linkkeket hozhatunk létre, amikhez aztán különböző hozzáférési jogosultágokat adhatunk meg a szabályokkal.

A jogosultságok finomhangolása a sysctl-en keresztül valósítható meg különböző változókkal, amelyek például meghatározzák a maximális gyermekfolyamatok számát a jailen belül, a SystemV IPC mechanizmus használatát, a mountolás jogát, a hostnév megváltoztathatóságát és így tovább (man 8 jail).

A fájlrendszeren lehetőség van beállítani kvótarendszert, melyet mountoláskor kell jeleznük az opciók között (userquota,groupquota), majd ezután az egyes könyvtárakra van lehetőségünk méret- és inodeszámbeli megkötéseket adnunk, melyek így az egyes jail-ek könyvtárára beállítva biztosítják a méret- és mennyiségkorlátozást.

3) Jailutils

A jailek userspace-béli kezeléséhez a jailutils szoftvercsomag áll a rendelkezésünkre.

Főbb elemei:

jstart – jail : jail elindítása, szükséges paraméterek fent

jkill : a jail leállítása

jails – jls : a futó jailek listája

jexec : egy futó jailben való utasítás végrehajtásához

/etc/rc.d/jail : általános script [fast|force|one](start|stop|restart|rcvar) lehetőségekkel

A jail parancs a /bin/sh /etc/rc parancs argumentummal indítva ugyanazt eredményezni, mint a jstart. A complete jail tehát nem más, mint a FreeBSD

rendszer (újra)indítása a jailben, ahol a gyermekfolyamatok mind öröklik a jail általi korlátokat.

4) Alkalmazási terület

Fő alkalmazása : service jailek, különböző alkalmazás szerver szoftver konfigurációk használata, anélkül, hogy a hostoló rendszerünket kéne kitenni az telepítés viszontagságainak. Egy rosszul sikerült telepítés után a problémás jail könyvtárszerkezete könnyen eltávolítható, míg a hostoló rendszerünk konfigurációja érintetlen marad.

5) Hálózati konfiguráció

A hálózati alkalmazások konfigurációjánál ügyelni kell arra, hogy a jailben és a hoston is használt hálózati alkalmazást úgy állítsuk be, hogy alapból ne a 0.0.0.0 (any) címen fogadja a kapcsolatokat, mert akkor a hostoló gép fogja kiszolgálni a kérést a jail helyett.

A hálózati konfigurációnál választhatunk, hogy vagy hozzáadjuk a jailnek beállított IP címet aliasként egy fizikai interfészhez, így a jailünk IP címéről szóló ARP kérésre válaszol a MAC címével, és ezután amíg érvényes a bejegyzés, a jailünknek szánt csomagot a hoszt gép fogadja, és a routing tábla alapján továbbítja a loopback interfészre, ahol az egyes jailek fogadni tudják a nekik szóló csomagot, gyakorlatilag ez analóg, a platformvirtualizációnál megismert bridgelt megoldással.

```
freebsd-vmware# ifconfig em0 alias 192.168.2.199 netmask 255.255.255.0
em0: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> metric 0 mtu 1500
      options=9b<RXCSUM,TXCSUM,VLAN_MTU,VLAN_HWTAGGING,VLAN_HWCSUM>
      ether 00:0c:29:a0:22:c9
      inet 192.168.2.107 netmask 0xffffff00 broadcast 192.168.2.255
      inet 10.0.0.2 netmask 0xff000000 broadcast 10.255.255.255
      inet 192.168.2.199 netmask 0xffffff00 broadcast 192.168.2.255
      media: Ethernet autoselect (1000baseTX <full-duplex>)
      status: active
```

A másik esetben a local loopback interfészre konfigurálunk aliast, és a jailünknek a privát címtartományból választunk címet, majd a natd beállításával tudunk hozzáférni a hálózatról. (port forwarding) Ez utóbbi tipikusan a webkiszolgálók alkalmazása esetén áll fent, amikor a host gépen az apache-on bekonfiguráljuk a Virtual Hostokat, és a kiszolgálóhoz történő továbbítás a domain név alapján történik.

```
freebsd-vmware# ifconfig lo0 alias 127.0.0.100 netmask 255.0.0.0
lo0: flags=8049<UP,LOOPBACK,RUNNING,MULTICAST> metric 0 mtu 16384
      inet6 fe80::1%lo0 prefixlen 64 scopeid 0x3
      inet6 ::1 prefixlen 128
```

```
      inet 127.0.0.1 netmask 0xff000000
      inet 127.0.0.100 netmask 0xff000000
```

Ha a host gépen létrehozunk különböző loopback interfészeket, classless címzéssel, akkor megoldható, hogy az egyes jailek ne lássák egymást, vagy csak az egy hálózatban lévők tudjanak kommunikáni egymással (pl. adatbázis jail + webkiszolgáló jail közös loopback hálózaton)

6) Jail indítása

A jail indítása történhet a /etc/rc.d/jail start <jailnév> scripttel, vagy a jail <jail root> <jail hostnév> [<jail ip cím>] <futtatandó parancs>

A parancs, ha a teljes konfigot indítani akarjuk, akkor a /bin/sh /etc/rc lesz.

```
freebsd-vmware# jail /usr/jails/test3/ test3 127.0.0.100 /bin/sh /etc/rc
Loading configuration files.
Creating and/or trimming log files:.
Starting syslogd.
ELF ldconfig path: /lib /usr/lib /usr/lib/compat
a.out ldconfig path: /usr/lib/aout /usr/lib/compat/aout
Clearing /tmp (X related).
Starting local daemons:.
Updating motd.
Starting cron.
Local package initialization:.
```

A futó jaileket a jls segítségével kérdezhetjük le.

```
freebsd-vmware# jls
   JID  IP Address      Hostname                      Path
    14  127.0.0.100     test3                         /usr/jails/test3
    12  127.0.0.2       test                          /usr/jails/test
    11  192.168.2.199   webserver                     /usr/jails/webserver
```

A jails parancs szintén a futó jaileket listázza, csak tömörebb formában:

```
freebsd-vmware# jails
test3
test
webserver
```

A hostról a jexec segítségével tudunk programokat futtatni a jailben, amihez szükségünk van a jail id-re. Hogy parancssort kapjunk , egyszerűen csak meg kell hívni a jexec segítségével:

```
freebsd-vmware# jexec 14 csh
test3#
```

7) Szoftver telepítés

A installworld, és a distribution után csak egy alap userlandet kapunk, ahol csak a szükséges alkalmazások vannak telepítve. Ha szeretnénk, akkor a sysinstall vagy a pkg_add -r segítségével a hálózatról tudunk csomagokat telepíteni, ha a hálóztati kapcsolatunk a jail-ben lehetővé teszi az internet elérést. Ha nem, akkor a FreeBSD telepítő cd-jét mountolhatjuk kívülről a jailbe, és a jail-béli root, a sysinstall segítségével telepítheti a kívánt alkalmazásokat.

```
freebsd-vmware#mount -t cd9660 /dev/acd0 /usr/jails/test/dist

freebsd-vmware# jexec 12 csh

test#sysinstall
```

Itt a configure/packages menüpont alatt az install from existing filesystem -nél adjuk meg az előbb mountolt /dist könyvtárat, és máris válogathatunk a telepíthető csomagok közül.

8) Hibák

– nem lehet cpu és ram limitet állítani az egyes jailekhez

– ha updateljük a host rendszert, akkor az összes jailünket updatelnünk kell, ami elég időigényes feladat

Az utóbbira megoldás: készítünk egy master templatet, és az egyes jailek könyvtárszerkezetébe linkeljük a megfelelő binárisokat tartalmazó könyvtárakat, az egyes jaileknek csak a konfigurációjuk lesz egyedi. Így update esetén csak egyszer kell a make userland procedúrán végigmennünk.

9) EZjail

A fent említett technikához készült egy keretrendszer, melynek neve ezjail. Telepítés után az ezjail-admin update -p paranccsal létrehozhatujuk az ezjail.conf fájlban található base_dir -be (alapértelmezetten a /usr/jails) a szükésges könyvtárszerkezetet, amely a basejail és a flavours könyvtárakból áll. A flavour-ök előre megírt “ízesítések” a létrehozandó jailekhez, amelyben megadhatunk felhasználókat, csoprtokat, fájlokat, telepítendő csomagokat, és post-installációs utasításokat. A példányosításnál megadott flavour-ök beállításai fognak alkalmazódni a létrehozott jailre. (pl. httpd flavour, mysql flavour)

Ha azt szeretnénk, hogy a létrehozott jailek bootolásnál induljanak, akkor a /etc/rc.conf -ba az ezjail_enabled=”YES” sort fel kell vennünk, de ekkor ügyelnünk kell arra, hogy az interfészbeállításokat, is elvégezzük az rc.conf -ban a hibamentes indulás érdekében.

ezjail-admin create -f flavour jailname jailip

Ez lérterhozza a jail számára könyvtárakat, linkeli a szükséges basejail bejegyzéseket, és végrehajta az egyes flavour-ökben megadott konfigurációkat.

10) Összefoglalás

A freebsd jails könnyen elsajátítható operációs rendszer szintű virtualizációt biztosít. A natív eszközökkel körülményesen adminisztrálható rendszer, az ezjail használatával, helytakarékossá, és effektívebbé válik a basejail használata által.

Nagy hátránya, hogy nem lehet cpu és ram korlátokat állítani, ami komoly hátrány a más OS virtualizációs rendszerekkel szemben, ahol ez biztosított.

További olvasásra:

- man 8 jail
- man 1 ezjail-admin
- http://www.freebsd.org/doc/handbook/jails.html
- http://wiki.freebsd.org/Jails
- http://erdgeist.org/arts/software/ezjail/