Des jails à bhyve

Baptiste Daroussin
bapt@FreeBSD.org

Sysadmin #5
Paris
3 décembre 2013

jails: Historique

- FreeBSD 4.0 (2000) : jail(8), jail(2)
- FreeBSD 5.1 (2003) : jls(8), jail_attach(2)
- FreeBSD 7.2 (2009) :
  - Multiple adresses IPs
  - Assignation de CPUs
- FreeBSD 8.0 (2009):
  - jail hierarchiques
  - Support VIMAGE
- FreeBSD 9.1 (2012):
  - Paramètres extensibles
  - jail.conf

jails: C'est quoi ?

chroot(2) boosté aux stéroïdes

jails: C'est quoi en vrai ?

- Solution de partitionnement: process, system de fichier, resources réseaux
- Restriction des privilèges root dans la "jail"
  - Aucune modifications kernel par accès directe possible (pas de chargement de modules par exemple)
  - Aucune modifications possible de la configuration réseau (sauf avec vnet)
  - Mountage des systèmes de fichiers impossible (sauf explicitement autorisé et/ou avec un délégation zfs)
  - Impossible de crée des nodes
  - Interdiction d'accéder aux sockets de type RAW, DIVERT et ROUTING
  - Interdiction de modifier les paramètres kernel: la majeure partie des appels sysctl sont interdits
  - Interdiction d'accéder aux resources réseau non associées à la jail

jails: Isolation d'un processus

Arborescence:

```
/jails/prison1/etc/resolv.conf
/jails/prison1/bin/process
/jails/prison1/libexec/ld-elf.so.1
/jails/prison1/lib/libc.so.7
/jails/prison1/lib/libz.so.5
```

Démarrage de la jail

```
$ jail -c name=prison1 path=/jails/prison1/ \
  host.hostname=majolieprison ip4=inherit /bin/process [params]
```

Arrêt de la jail

```
$ jail -r prison1
```

jails: Emprisonner une instance FreeBSD

Création de la jail:

```
$ mkdir /jails/prison2
$ tar -xf base.txz -C /jails/prison2
$ jail -c -n prison2 persist mount.devfs ip4=inherit \
  path=/jails/prison2 host.hostname="prison2"
```

Le fichier jail.conf

```
prison2 {
    host.hostname=prison2;
    persist;
    ip4=inherit;
    mount.devfs;
    path=/jails/prison2;
}
```

FreeBSD

jails: Emprisonner CentOS6

Faire dire au linuxulator qu'il utilise un kernel 2.6.32 :

```
$ sysctl -w compat.linux.osrelease=2.6.32
```

Créer la jail :

```
$ tar -xf ~/centos-6-x86.tar.gz -C /jails/linux
```

Le jail.conf :

```
centos {
   host.hostname=centos;
   persist;
   ip4=inherit;
   mount.devfs;
   path=/jails/linux;
   exec.prestart="mount -t linprocfs none /jails/linux/proc; mount -t linsysfs none /jails/linux/sys";
   exec.poststop="umount /jails/linux/proc /jails/linux/sys";
}
```

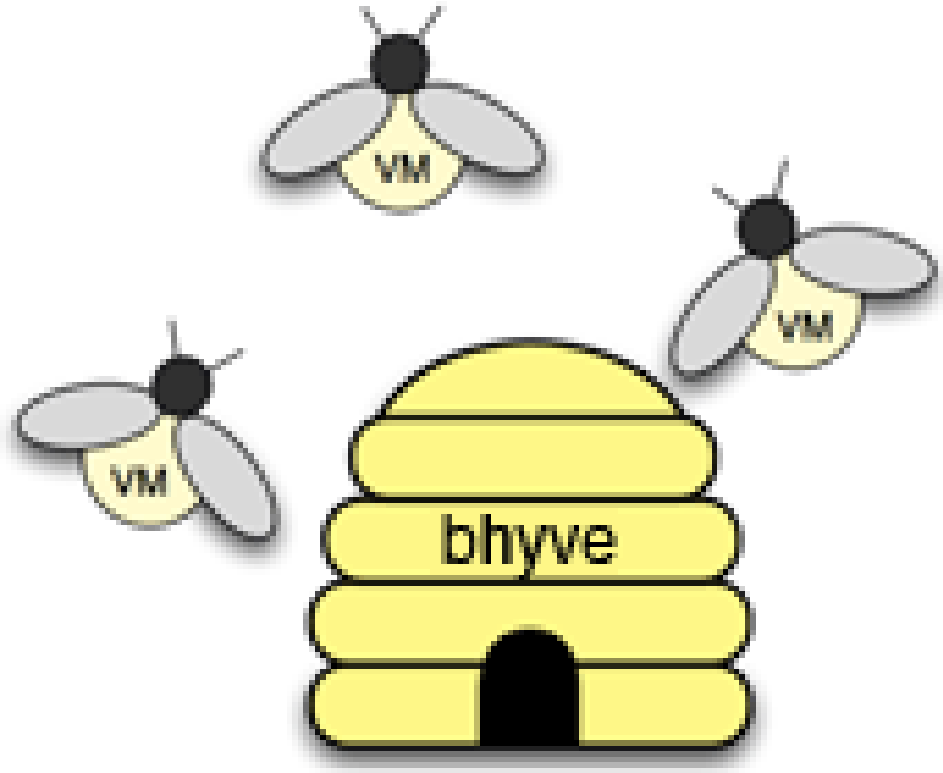
Résultat:

```
$ jexec centos sh -c "uname -a && cat /etc/redhat-release"
Linux centos 2.6.32 FreeBSD 11.0-CURRENT #22 r257864M: Mon Nov 11 21:33:55 CET 2013 i686 i686 i386 GNU/Li
CentOS release 6.4 (Final)
```

jails: Quelques liens

- ezjail `http://erdgeist.org/arts/software/ezjail/`
- warden `http://wiki.pcbsd.org/index.php/Warden`
- Jail Handbook
  `http://www.freebsd.org/doc/handbook/jails.html`
- Jails: Confining the omnipotent root
  `http://phk.freebsd.dk/pubs/sane2000-jail.pdf`

bhyve: The BSD Hypervisor



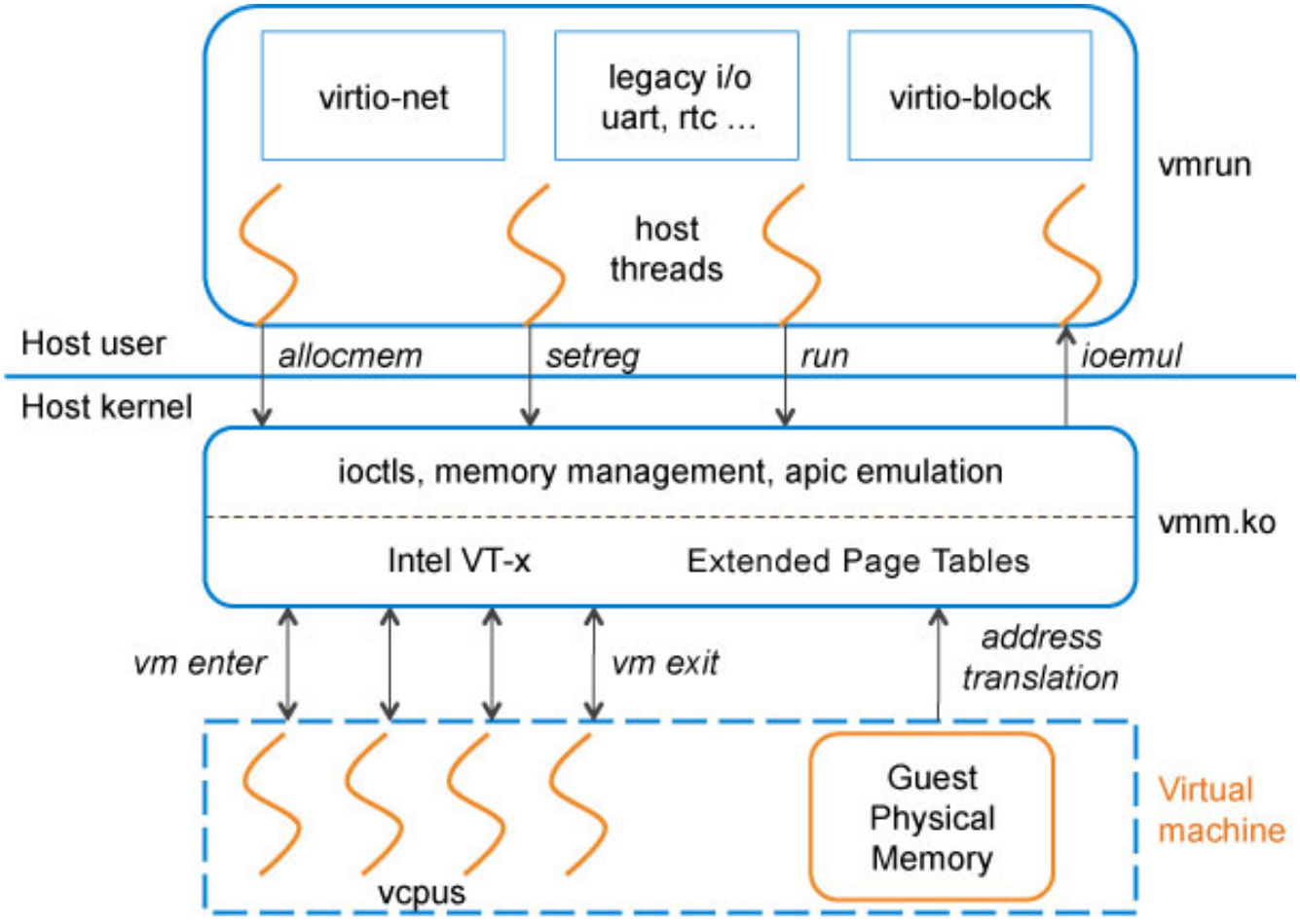
FreeBSD

bhyve: Qu'est que c'est ?

- Hyperviseur de type2
- Écrit par Neel Natu et Peter Grehan
- Première annonce BSDCan 2011
- Requiert Intel Extended Page Tables (EPT)
- Présent dans FreeBSD 10.0-RELEASE
- Utilise les interruptions MSI/MSI-X
- Support VirtIO (fortement recommandé)

FreeBSD

bhyve: apperçu



FreeBSD

bhyve: composants

- */usr/sbin/bhyve* user-space sequencer and I/O emulation
- */usr/sbin/bhyveload* user-space FreeBSD boot loader
- */usr/sbin/bhyvectl* utilitaire de control bhyve
- */usr/local/sbin/grub-bhyve* user-space grub2

bhyve: OS invités

Supportés:

- Toutes les versions de FreeBSD supportant virtio (8.4+)
- Toutes les versions d'OpenBSD depuis Octobre 2013
- GNU/Linux amd64
- Requiert Intel Extended Page Tables (EPT)
- Présent dans FreeBSD 10.0-RELEASE

Pas encore supportés:

- Microsoft Windows
- NetBSD: pas de support MSI-X ni virtio
- DragonFlyBSD: manque une version userspace du loader
- SmartOS: pas de support grub2

bhyve: Créer une VM FreeBSD

```
# Création du disque
$ truncate -s 8G fbsd.img
# Création de l'interface tap
$ ifconfig tap0 create
$ ifconfig bridge0 up addm em0 addm tap0
# Chargement du kernel
$ bhyveload -m 512 -d ./FreeBSD-10.0-BETA3-amd64-bootonly.iso fbsd
# Lancement de l'installation
$ /usr/sbin/bhyve -c 2 -m 512 -AI -H -P \
  -s 0:0,hostbridge \
  -s 1:0,virtio-net,tap0 \
  -s 2:0,virtio-blk,./fbsd.img \
  -s 3:0,virtio-blk,./FreeBSD-10.0-BETA3-amd64-bootonly.iso \
  -S 31,uart,stdio \
  fbsd
```

vmrun.sh:

```
#!/bin/sh
while :; do
  bhyvectl --destroy --vm=fbsd
  bhyveload -m 512 -d ./fbsd.img fbsd
  /usr/sbin/bhyve -c 2 -m 512 -AI -H -P \
  -s 0:0,hostbridge \
  -s 1:0,virtio-net,tap0 \
  -s 2:0,virtio-blk,./fbsd.img \
  -S 31,uart,stdio \
  fbsd || break
done
```

FreeBSD

bhyve: Créer une VM Debian

```
# Création du disque
$ truncate -s 8G debian.img
# Le fichier map
$ echo "(hd0) ./debian.img" > debian.map
$ echo "(hd1) ./debian-7.2.0-amd64-netinst.iso" > debian.map
# Chargement du kernel
$ grub-bhyve -r hd1 -m debian.map -M 512 debian
# Lancement de l'installation
$ /usr/sbin/bhyve -c 2 -m 512 -AI -H -P \
  -s 0:0,hostbridge \
  -s 1:0,virtio-net,tap0 \
  -s 2:0,virtio-blk,./debian.img \
  -s 3:0,ahci-cd,./debian-7.2.0-amd64-netinst.iso \
  -S 31,uart,stdio \
  debian
```

vmrun.sh:

```
#!/bin/sh
while :; do
  bhyvectl --destroy --vm=debian
  grub-bhyve -r hd0,1 -m debian.map -M 512 debian
  /usr/sbin/bhyve -c 2 -m 512 -AI -H -P \
  -s 0:0,hostbridge \
  -s 1:0,virtio-net,tap0 \
  -s 2:0,virtio-blk,./debian.img \
  -S 31,uart,stdio \
  debian || break
done
```

FreeBSD

bhyve: Quelques liens

- Site de bhyve `http://bhyve.org/`
- Wiki sur bhyve `https://wiki.freebsd.org/bhyve`
- Petite cloud (bhyve frontend)
  `http://www.petitecloud.org/`
- Scripts bhyve `http://bhyve.org/bhyve-script.tar`
- Grub2 bhyve
  `https://github.com/grehan-freebsd/grub2-bhyve`
- The bhyve Operator's manual
  `http://bhyve.org/bhyve-manual.txt`

FreeBSD

Merci de votre attention !

Questions ?