Virtualization on the Hurd

Justus Winter <justus@gnupg.org>

2017-02-04

What is the Hurd, and why should I care?

- general-purpose multiserver operating system
- GNU: replacement for traditional OS kernel
  - that didn't happen. . .
- me: an independent long-term research project
  - it exists and is highly compatible (glibc, Debian/Hurd)
  - learn systems programming
  - learn to contribute to (GNU) projects
  - free ones mind from narrow definitions of what an OS can do and be
- Freedom #0: The freedom to run the program as you wish, for any purpose.

Virtualization

- virtualization is everywhere, and not going away anytime soon
- different goals
  - ease management (teh cloud)
  - increase isolation
  - development
  - testing
- granularity
  - coarse-grained (bochs, qemu, . . .)
  - somewhere in between ({LD_LIBRARY_,}PATH)
  - fine-grained (LD_PRELOAD trickery)

Renzo Davoli's definition

Virtualization is the ability to replace / interpose a resource.

My definition

Virtualization is the ability to freely shape the computation environment.

Subhurds: coarse-grained virtualization

- one kernel, multiple logical systems
- a bit like containers, zones, jails, . . .
- tricky to do on monolithic systems
  - a decade of getting namespaces right in Linux
- next to trivial on multiservers

Booting a Subhurd

```
$ boot /dev/sd1s1
/hurd/ext2fs.static --readonly [...] -T device pseudo-root
/lib/ld.so /hurd/exec
Hurd server bootstrap: ext2fs.static[pseudo-root] exec startup
[...]
Debian GNU/Hurd 9 sub-hurdbox console


login:
```

Subhurds: How do they work?

- Hurd runs unmodified inside
- /bin/boot starts a second set of Hurd servers
- environment similar to stock GNU Mach
- most resources are not virtualized (tasks, threads, IPC, memory objects)
- up to 0.9: privileged Subhurds
  - virtualized resources: console, root, time, device master port
- as of 0.9: unprivileged Subhurds
  - virtualized resources: task notifications, privileged host control port, privileged processor set port
- /bin/boot is tiny, 2k6 SLOCs (that includes the bootscript parser, lot's of stubs)

Interludum: Service lookups on the Hurd

- central design aspect: use the VFS as namespace for service lookups
  - /dev/{null,zero,full,console,hd0s1, . . . }
  - /servers/{crash,startup}
  - /servers/socket/{1,2,26,local,inet,inet6}
- distributed, filesystem servers "span" the VFS tree
- *translator records* are recorded in nodes
- *translators* are started on demand

Example: The network stack.

```
$ showtrans /dev/netdde
/hurd/netdde
$ showtrans /dev/eth0
/hurd/devnode -M /dev/netdde eth0
$ showtrans /dev/eth0m
/hurd/eth-multiplexer --interface=/dev/eth0
$ fsysopts /servers/socket/2
/hurd/pfinet --interface=/dev/eth0m/0 --address=192.[...]
```

Translators & the VFS

- *translators* translate between one domain and the VFS
  - /hurd/ext2fs translates between ext2 disk-format and VFS
  - /hurd/httpfs translates between http and VFS
- "*VFS*"?
  - all operations defined in hurd/fs.defs
  - e.g. $dir\_lookup : (Node \times path \times flags \times mode) \rightarrow Node$
  - arbitrary protocols, e.g. /servers/startup also speaks hurd/startup.defs

Example: POSIX filesystem semantics.

```
$ rpctrace stat /etc/hostname
[...]
  100<--153(pid28072)->dir_lookup ("etc/hostname" 64 0) \
    = 0 1 ""    165<--168(pid28072)
  165<--168(pid28072)->io_stat () \
    = 0 {23 5 0 229594 0 1483915082 0 33188 1 0 0 7 0 [...]}
[...]
```

Fine-grained virtualization

- (almost) every Hurd server is reachable via the VFS
- underappreciated translator family: translating between VFS and VFS
    - fakeroot, remap, identity. . .
- every process has a working directory and a root directory
- root directory can be set using "settrans –chroot"
    - note: "settrans –chroot" != UNIX chroot

Example: settrans –chroot in action.

```
$ settrans --chroot cat /etc/hostname -- / \
  /hurd/remap /etc/hostname $HOME/my_hostname
Hello FOSDEM :>
$ remap /servers/socket/2 $HOME/servers/socket/2 -- \
  iceweasel
```

/hurd/identity: the identity translator

- the *identity* translator computes the identity function from VFS to VFS
- simplest possible translator
- a bit like a symlink / firmlink
  - however, links redirect the client
  - *identity* performs actions on behalf of client

Example: /hurd/identity in action.

```
$ ls $HOME/demo
bin  lib
$ settrans -ac mnt --underlying $HOME/demo /hurd/identity
$ ls mnt
bin  lib
$ fsysopts mnt
trans/identity
$ settrans --chroot ls / -- $HOME/demo /hurd/identity
bin  lib
```

/hurd/gpg: the transparent GnuPG translator

- transparent OpenPGP support for every program
  - decrypt
  - encrypt
  - verify

Example: /hurd/gpg in action.

```
$ verify tar tf /ftp://ftp.gnu.org/gnu/hurd/hurd-0.9.tar.bz2
[...]
gpg: Good signature from "Thomas Schwinge [...]"
hurd-0.9/
hurd-0.9/.gitignore
[...]
$ encrypt for demo@example.org -- tar cf foo.tar.xz my_hosts
$ file foo.tar.xz.gpg
foo.tar.xz.gpg: PGP RSA encrypted session key - keyid: [...]
```

Empowering the user to explore her system

- VMM are an easy sell: it's just like another computer
  - a bit deceptive, large attack surface
- capabilities give raise to a relation over processes
- VFS is a tree, a tree is a graph . . .
- first prototype with graphviz
- `https://d3js.org`
- if only the Hurd had a REST interface . . .

Demo time!

`http://venus:8000/translators/ http://venus:8000/spaces/`

Questions & references

- VFS manipulation is fine-grained virtualization on the Hurd
- virtualization is easy and fun on multiservers
- visit us in #hurd, bug-hurd@gnu.org, talk to us.
- what about /hurd/guix
- Hand et al.: Are Virtual Machine Monitors Microkernels Done Right?
- Heiser et al.: Are Virtual Machine Monitors Microkernels Done Right?

Questions?

Might be answered, or not, depending on my mood, how much time is left, and the amount of dust in the universe.

Interested in the Hurd?

Come to Manolis talk about adding GNU/Hurd support to GNU Guix and GuixSD. Sunday, 15:00 - 15:30 in K.4.601.