Latest news about GNU Hurd

Samuel Thibault

2016 August 18th
It's all about freedom #0

“The freedom to run the program, for any purpose”

I.e.:

- Freedom from sysadmin!
  - WTH is fdisk/mke2fs/... hidden in /sbin?
  - I should be able to just work with my disk/network access

- Freedom to innovate
  - Experimental filesystem, personal work-flow, new kind of process combination,...

- Also provide freedom from misbehaving programs and drivers
Micro-kernel layering

Kernel

Tasks, memory, IPC

root

proc

auth

ext2fs

pfinet

user

sh

cp
Micro-kernel layering
Micro-kernel layering

- Server crash? Not a problem
  - “Computer bought the farm” is just an error, not something-of-the-death
- Easier to debug/tune
  - Just run gdb, gprof, …
- Can dare crazy things
  - The Hurd console has dynamic font support
    - See chinese support in pseudo-graphical mode (actually pure VGA textmode!) of Debian installer.
- Kernel only handles Tasks, memory, IPC
Hurd possibilities

ext2fs
pfinet
auth
proc
ftpfs
sh
cp

Kernel
Hurd possibilities

\texttt{\$ settrans -c \~{}/ftp: /hurd/hostmux /hurd/ftpfs /}

(just once for good)

\texttt{\$ settrans -a \~{}/mnt /hurd/iso9660fs}
\texttt{\~{}/ftp://ftp.gnu.org/old-gnu/gnu-f2/hurd-F2-main.iso}

\texttt{\$ ls \~{}/mnt}

\texttt{README-or-FAIL}

...

- Only downloads what is needed.
- Can be permanently stored in ext2fs

\texttt{\$ settrans \~{}/.signature /hurd/run /usr/games/fortune}
But also

- remap `/bin/sh` to `$HOME/bin/sh`
- remap `/bin` to `$HOME/unionbin`

...  

- Check out Stow/Nix/Guix!
How does it work?

Kernel

ext2fs
pfinet
auth
proc
ftpfs
sh
libc
user
cp
libc

root
Rationale

- **Everything** is an (interposable) RPC
- Translators exposed in the FS
  - The user gets to decide what/how to interpose
    - Without need for costly ptrace or fragile libc symbols interposition.
    - **Native** fakeroot/chroot
    - Fully virtualized and fine-grained interface
  - Just need to use what's provided by the admin, e.g.
    - `$HOME/`
    - TCP/IP stack
  and pile over it
Hurd possibilities (cont'ed)

Kernel

ext2fs
pfinet
root
open
auth
proc
vpn
ftpfs
part
ext2fs
sh
cp
user
isofs
Hurd possibilities (cont'ed)

i.e. ISO image inside a partitioned disk image on ftp over a VPN
Hurd userland network support

/servers/socket/2

Kernel

root

eth0

pfinet

w3m

user
Hurd userland network support

DDE layer
Linux 2.6.32 drivers
/dev/eth0

root

Kernel

pfinet

eth0

w3m

user
Hurd userland network support

Kernel

root

eth0

eth-filter

pfinet

w3m

user
Hurd userland network support

Kernel

/root

user

~servers/socket/2

~servers/tun0

eth0

eth-filter

pfinet

openvpn

w3m

dservers/socket/2

d servers/socket/2
Hurd userland network support

```bash
settrans -ca ~/servers/socket/2
~*/bin/pfinet -i ~/servers/tun0
-a 80.67.176.254 -p 80.67.179.1

vpn.sh ~/servers/tun0 &

~*/bin/remap

servers/socket/2 servers/socket/2

/~/etc/resolv.conf ~/resolv.conf

wget www.gnu.org

- My own translators
- Only wget accesses my pfinet (well, the shell too :) )
```
DDE stack

Based on TU-Dresden's DDE stack

- Zheng Da's GSOC
- Ported to Mach kernel
- Ported to Mach device interface
- Updated libdde_linux26 for long-term-supported linux 2.6.32
  - Most drivers (and mostly the really useful ones) just work without patches
- Used by default by Debian GNU/Hurd
DDE stack

- pfinet
  - device_read
  - device_write

- pkg_xmit
- rx_callback

- Linux drivers
- libdde_linux26
- libmachdev
- libddekkit

- Linux API
- ddekkit API

- intr
- vm_allocate_contiguous

- kernel
DDE stack

• Only two additions to the kernel
  • Interrupt delivery and masking
  • Physically-contiguous memory allocation
  • (Direct I/O access was already available)
• Performance similar to in-kernel driver
• Driver in a separate process
  • Can just crash and be happy with it...
  • Can easily debug and profile them
  • Stack smashing protection ;)
• Could benefit from I/O MMU for better isolation.
  - For now drivers can just access all RAM...
More DDE?

• Disk DDE is supposed to be working
  • Should be not very complex
    - device_read / device_write
  • Zheng Da said he didn't manage to make it work
• USB/sound DDE was mentioned as experimental
  • I don't know the status?
  • We would definitely love to have that
• Rather use Rump kernels?
Rump sound support, v0
Rump sound support, v1

- BSD
- src-gnu
- librumpuser
- libpciaccess
- /dev/pcm
- mplayer
- intr
- vm_allocate_contiguous
- kernel
Rump sound support, v2

```
/dev/pcm
rumpuser
/home/samy/dev/pcm
rumpuser
PCI arbiter
IOMMU
intr vm_allocate_contiguous
kernel
```
Rump USB support

- ext2fs
- /dev/usbd0
- rump-usb-storage
- libusb
- /dev/usb
- rumpuser
- PCI arbiter
- IOMMU
- intr
- vm_allocate_contiguous
- kernel
Rump USB support

ext2fs
/dev/usbd0
rump-usb-storage
libusb
/dev/usb
rumpuser

usb-foo
rumpuser
PCI arbiter
IOMMU

intr    vm_allocate_contiguous    kernel

ext2fs
usb-storage
rumpuser
Rump virtio disk support

ext2fs

Id driver
virtio
rumpuser

PCI arbiter
IOMMU

intr vm_allocate_contiguous

kernel
Current State

Hardware support

• i686
• start of 64bit support
  • Kernel boots completely, now missing RPC 32/64bit translation
• DDE Linux 2.6.32 drivers layer for network boards
  • In userland netdde translator!
• IDE, Xorg, …
• AHCI driver for SATA
• Xen PV domU
  • Required GNU Mach changes only
• Preliminary sound support through userland Rump
• No USB yet
Current State

Software support

• Quite stable
  • Have not reinstalled boxes for a decade.
  • Debian buildds keep building packages, no hang after weeks!

• ~80% of Debian archive builds out of tree
  • XFCE, almost gnome, almost KDE
  • Firefox (aka iceweasel), gnumeric, …

• Standard *native* Debian Installer
Recent work

- GNU Guix and GuixSD
  - Guix used daily on Debian GNU/Hurd
  - A pure GNUish GNU/Hurd distro!
  - Proper bootstrap of the Hurd chain
    - Used by Debian GNU/Hurd rebootstrap effort
  - Proper isolation of builds
    - fakerooot+firmlinks instead of fakerooot+bind
  - A bit more work to be bootable
- Fixed native fakerooot
- Using xattr for storing translators
Recent work (2)

- SCM_CRED
- Various optimizations and stabilization
  - Protected payloads
  - Paging management
  - Message dispatch
  - Gsync ~= futex, used in glibc & libpthread
- Automatic code checking
  - Port references: static analysis, runtime check?
Releases

- Nice 0.401 release on April 2011.
- Arch Hurd LiveCD release on August 2011.
- Hurd 0.8, Mach 1.7, MIG 1.7
- Released Debian-unofficial
  - wheezy/sid snapshot CDs on May 2013 \o/
  - jessie/sid snapshot CDs on May 2015 \o/
Future work

- Highmem support
- X86_64 support
- Read-ahead
- \{sound,usb\} Rump drivers
- GNU system: Guix/Hurd?
- Startup in scheme?
- Your own pet project?
Thanks!

- For listening
- And to the people working on all this
- http://hurd.gnu.org/
- http://www.debian.org/ports/hurd/
- The increasing irrelevance of IPC performance for microkernel-based Operating Systems