Software engineering tools based on Syscall instrumentation*

* on Linux

cedric.vincent@st.com
STMicroelectronics
Compilation Expertise Center
FOSDEM 2014

Syscalls

“The system call is the fundamental interface between an application and the Linux kernel.”

— man syscalls

Syscalls give access to

- file-system
- network
- process management
- memory management
- ...

Syscalls are used to access everything but CPU registers and mapped memory

The “ptrace” syscall can be used to observe & modify syscalls* without privileges on all Linux versions

* of other processes

PRoot*

* pronounced “p root” ☺

Path remapping

```
$ proot -m ~/hosts:/etc/hosts <command>
```

ex.:
open(”/etc/hosts”, …)
↳ open(”/home/cedric/hosts”, …)

Path remapping

It works with directories too:

```
$ proot -m ~/opt:/opt <command>
```

It’s like “mount --bind”
without any privileges!

Path remapping

What if one maps a directory over “/”?

```
$ proot -m ~/rootfs:/ <command>
```

Path remapping

What if one maps a directory over “/”?

```
$ proot -m ~/rootfs:/ <command>
```

It’s like “chroot”
without any privileges!

PRoot use cases

- cross-build
- cross-validation
- cross-debug
- ...

* without setup nor privileges

Incompatible kernel?

```
$ proot -R ~/rootfs echo OK
FATAL: kernel too old
```

Incompatible kernel?

```
$ proot -R ~/rootfs echo OK
FATAL: kernel too old
```

```
$ proot -R ~/rootfs -k 2.6.32 echo OK
OK
```

ex.:
openat(fd, ”foo”, …)
↳ open(”<fd>/foo”, …)

Incompatible CPU?

```
$ proot -R ~/arm-fs echo OK
Exec format error
```

Incompatible CPU?

```
$ proot -R ~/arm-fs echo OK
Exec format error
```

```
$ proot -R ~/arm-fs -q qemu-arm echo OK
OK
```

ex.:
execve(”/bin/echo”, ”OK”)
↳ execve(”/usr/bin/qemu-arm”, ”/bin/echo”, …)

PRoot is a generic instrumentation engine, it can be extended and be used to implement other tools…

CARE

Comprehensive Archiver for Reproducible Execution

CARE use cases

- bug reports
- demos
- portable apps
- ...

Example: archive

```
$ ./care -o foo.bin commands.sh
[…]
tar -xf perl-5.18.1.tar.gz
cd perl-5.18.1
./configure.gnu
make -j 4
[…]
```

Example: re-execute

```
elsewhere$ du -h foo.bin
42M    foo.bin
elsewhere$ ./foo.bin
info: extracted: foo/rootfs/usr
info: extracted: foo/rootfs/usr/local
[…]
elsewhere$ ls foo/
README.txt rootfs/ proot re-execute.sh
```

Example: re-execute

```
elsewhere$ ./foo/re-execute.sh
tar -xf perl-5.18.1.tar.gz
cd perl-5.18.1
./configure.gnu
make -j 4
[…]
```

CARE requires
no setup and
no privileges,
both for archiving
and re-executing

Other tools & extensions

Fake id0

fakes root privileges

DepsTracker

observes & computes build deps.

PStrace

is an eye-candy strace-like

<insert your own here>

Contact & details

PRoot: http://proot.me

CARE: http://reproducible.io