Nix

Newcastle Cybersecurity Group

Jay Rovacsek

February 24, 2022

Public speaking skills: a solid 2.75/10 - remind me now that I should slow down and chill-out

$whoami

Wait a minute - you used that slide last year.

Yep! This talk is somewhat of a follow-on of the Feb 2021 session

If you have questions at any point feel free to jump in!

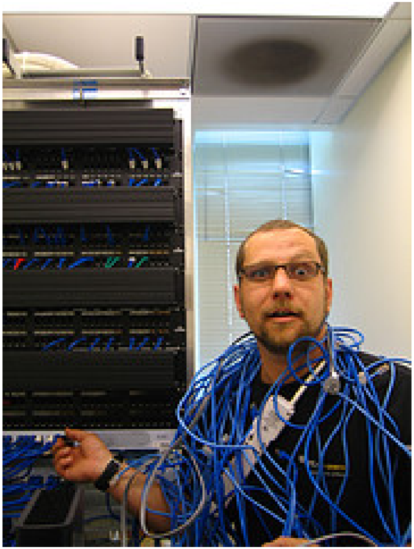
$whoami: I do cybers, but also very keen on:

- Chasing my kids around
- Generation 1 Pokemon, as others are inferior
- Picking up heavy things and putting them down

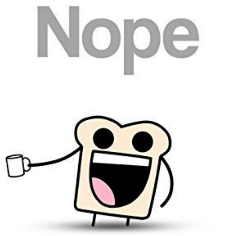
$whoami

What am I still not?

- A sysadmin
- Good at making presentation slides



Does any of this matter?



But why am I here presenting?

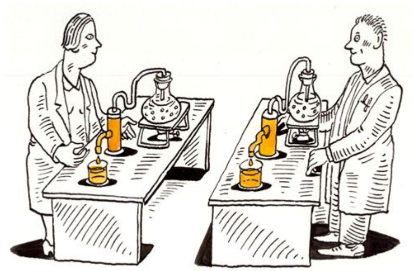
¯_(ツ)_/¯

- ~~Home networks are fun~~
- ~~Breaking your own stuff will assist in teaching you~~
- ~~Can run some cool services for yourself/family/friends~~

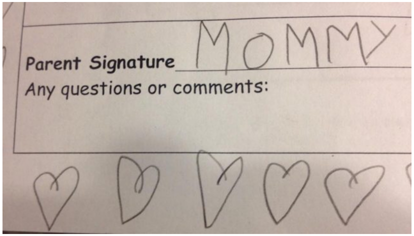
We might care about:

- Reproducibility of software
- Declarative Environments
- Reliable Environments
- Believing the hype on lobste.rs or Y-Combinator!

Reproducibility of software



Is reproducibility really that important?


Parent Signature MOMMY
Any questions or comments:

It depends

Threat modeling some upstream software sources:

- Malicious changes to source code
- User/System compromise within the development chain
- Watering-hole attacks
- Any more?

Reproducibility protects us from 2 of 3 of these

- ~~Malicious changes to source code~~
- User/System compromise within the development chain
- Watering-hole attacks

How does Nix provide reproducibility?

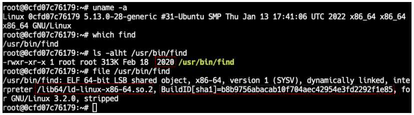
We'll consider a simple binary on most Linux systems: find

/nix/store**/jjvw20r6pz3ff7pn91yhvfx8s7izsqan**-findutils-4.8.0/bin/find

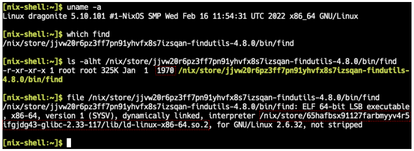
what in tarnation

Anyone know what this would look like on Ubuntu?

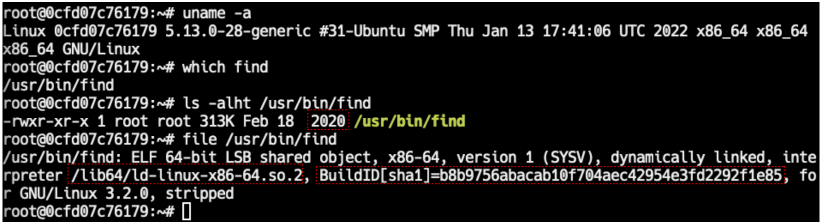
Ubuntu:


```
root@0cfd07c76179:~# uname -a
Linux 0cfd07c76179 5.13.0-28-generic #31-Ubuntu SMP Thu Jan 13 17:41:06 UTC 2022 x86_64 x86_64
x86_64 GNU/Linux
root@0cfd07c76179:~# which find
/usr/bin/find
root@0cfd07c76179:~# ls -alht /usr/bin/find
-rwxr-xr-x 1 root root 313K Feb 18  2020 /usr/bin/find
root@0cfd07c76179:~# file /usr/bin/find
/usr/bin/find: ELF 64-bit LSB shared object, x86-64, version 1 (SYSV), dynamically linked, inte
rpreter /lib64/ld-linux-x86-64.so.2, BuildID[sha1]=b8b9756abacab10f704aec42954e3fd2292f1e85, fo
r GNU/Linux 3.2.0, stripped
root@0cfd07c76179:~# 
```

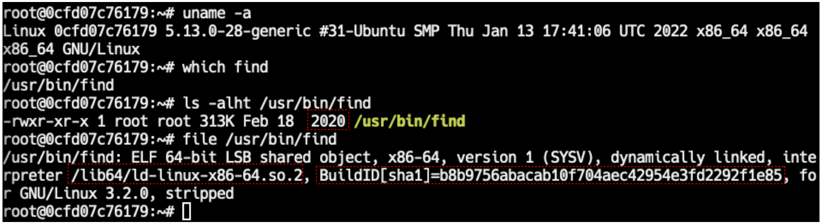
Nix:


```
[nix-shell:~]$ uname -a
Linux dragonite 5.10.101 #1-NixOS SMP Wed Feb 16 11:54:31 UTC 2022 x86_64 GNU/Linux

[nix-shell:~]$ which find
/nix/store/jjvw20r6pz3ff7pn91yhvfx8s7izsqan-findutils-4.8.0/bin/find

[nix-shell:~]$ ls -alht /nix/store/jjvw20r6pz3ff7pn91yhvfx8s7izsqan-findutils-4.8.0/bin/find
-r-xr-xr-x 1 root root 325K Jan  1  1970 /nix/store/jjvw20r6pz3ff7pn91yhvfx8s7izsqan-findutils-
4.8.0/bin/find

[nix-shell:~]$ file /nix/store/jjvw20r6pz3ff7pn91yhvfx8s7izsqan-findutils-4.8.0/bin/find
/nix/store/jjvw20r6pz3ff7pn91yhvfx8s7izsqan-findutils-4.8.0/bin/find: ELF 64-bit LSB executable
, x86-64, version 1 (SYSV), dynamically linked, interpreter /nix/store/65hafbsx91127farbmyyv4r5
ifgjdg43-glibc-2.33-117/lib/ld-linux-x86-64.so.2, for GNU/Linux 2.6.32, not stripped

[nix-shell:~]$ 
```

There's a few things to note in these images, taking for example Ubuntu:



```
root@0cfd07c76179:~# uname -a
Linux 0cfd07c76179 5.13.0-28-generic #31-Ubuntu SMP Thu Jan 13 17:41:06 UTC 2022 x86_64 x86_64
x86_64 GNU/Linux
root@0cfd07c76179:~# which find
/usr/bin/find
root@0cfd07c76179:~# ls -alht /usr/bin/find
-rwxr-xr-x 1 root root 313K Feb 18  2020 /usr/bin/find
root@0cfd07c76179:~# file /usr/bin/find
/usr/bin/find: ELF 64-bit LSB shared object, x86-64, version 1 (SYSV), dynamically linked, inte
rpreter /lib64/ld-linux-x86-64.so.2, BuildID[sha1]=b8b9756abacab10f704aec42954e3fd2292f1e85, fo
r GNU/Linux 3.2.0, stripped
root@0cfd07c76179:~# 
```



```
root@0cfd07c76179:~# uname -a
Linux 0cfd07c76179 5.13.0-28-generic #31-Ubuntu SMP Thu Jan 13 17:41:06 UTC 2022 x86_64 x86_64
x86_64 GNU/Linux
root@0cfd07c76179:~# which find
/usr/bin/find
root@0cfd07c76179:~# ls -alht /usr/bin/find
-rwxr-xr-x 1 root root 313K Feb 18  2020 /usr/bin/find
root@0cfd07c76179:~# file /usr/bin/find
/usr/bin/find: ELF 64-bit LSB shared object, x86-64, version 1 (SYSV), dynamically linked, inte
rpreter /lib64/ld-linux-x86-64.so.2, BuildID[sha1]=b8b9756abacab10f704aec42954e3fd2292f1e85, fo
r GNU/Linux 3.2.0, stripped
root@0cfd07c76179:~# 
```

- File metadata regarding the create time has been left in
- The linker is a pretty standard location for this
- We've got a build ID that links to Canonical's build

Do we trust this binary?

I mean, generally yeah we probably could.

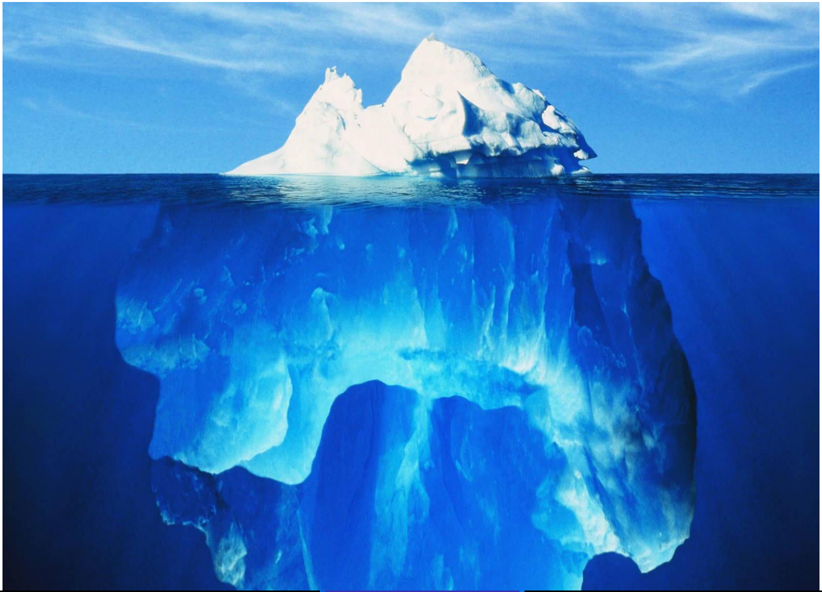
Could we reproduce the *exact same binary* given the source code of **find**?

We could probably, but we'd need to:

- Utilise a build system that applies the same build process as the original
- Ensure we had no volatile inputs (network data is consistent and trustworthy & more)
- Utilise the same makefile (was it the original GNU one or a modified one?)
- Ensure the compiler zeros all memory before value initialisation
- Uses the same version information
- Ensure the compiled output has the same timestamp
- Ensure no locale-specific options have side effects on the compile
- Ensure stable ordering for outputs (Do python dictionaries always yield their keys in the same order?)
- Eliminate PRNG in compile if relevant
- Ensure debug symbols are removed (may contain environment

Was that last slide cutoff?

Yep! Those considerations are just the tip of the iceberg!



So how does Nix avoid all of these pitfalls?

Nix creates *derivations* which is just a fancy word for builds.

Derivations are however distinct to not only the software version you're building. But all of the inputs also

Because of this feature, we can *know* that the method in which we build and install our software of choice is <u>the same</u> as what the author of the derivation intended.

Why is it provable with Nix?

In short: the generated path is linked to both the software as well as all inputs within the dependency tree

/nix/store**/jjvw20r6pz3ff7pn91yhvfx8s7izsqan**-findutils-4.8.0/bin/find

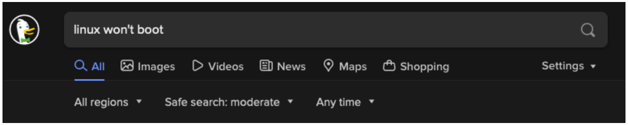
Steps to generate this hash (paraphrasing):

- SHA256 the descriptor of a build
- Recursively SHA256 files required in the build process in a deterministic order, truncating output then base32
- ???
- ???
- Profit!

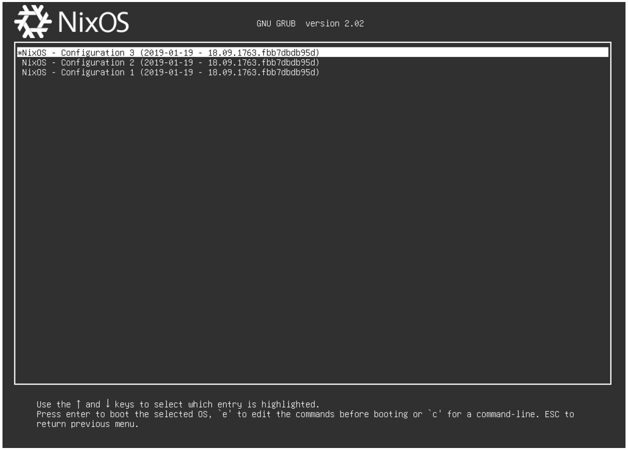
Are all Nix configs and packages reproducible?

Questions on any of this?

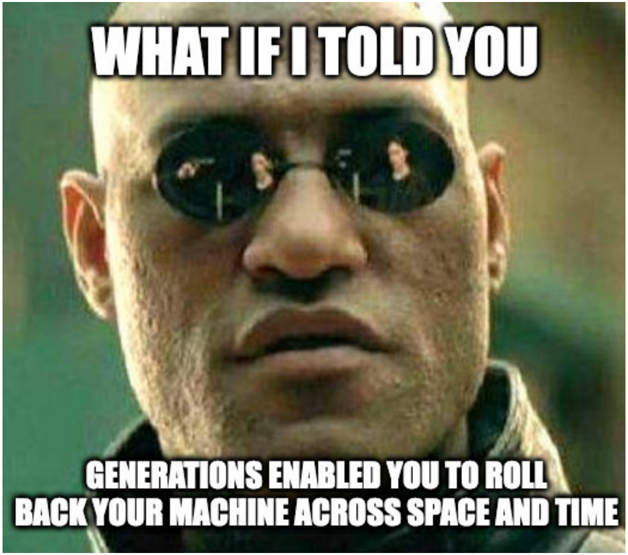
Reliability



linux won't boot



```
NixOS                    GNU GRUB  version 2.02

*NixOS - Configuration 3 (2019-01-19 - 18.09.1763.fbb7dbdb95d)
 NixOS - Configuration 2 (2019-01-19 - 18.09.1763.fbb7dbdb95d)
 NixOS - Configuration 1 (2019-01-19 - 18.09.1763.fbb7dbdb95d)

    Use the ↑ and ↓ keys to select which entry is highlighted.
    Press enter to boot the selected OS, `e' to edit the commands before booting or `c' for a command-line. ESC to
    return previous menu.
```


WHAT IF I TOLD YOU
GENERATIONS ENABLED YOU TO ROLL BACK YOUR MACHINE ACROSS SPACE AND TIME

Downsides of generations?

- Well, mostly just disk space

But who doesn't have the ability to utilise multi-gigabyte drives nowadays?

Resolving disk space issue?

Easy! Nix has a garbage collector!

Where to from here?

Fin!