

Nix

Question

- ▶ If I would ask you to check out a 1 year old commit of a project
- ▶ Would you be able to build it to track down a bug?
- ▶ Why / Why not?

Current state of affairs

- ▶ Operating systems are built on `#globals#` and `#mutation#`
- ▶ `apt-get install nginx` just in-place mutates `/usr/bin/nginx` and `/usr/lib/nginx/`
- ▶ In-place mutation can't be undone.
- ▶ Only one version installed at the same time
- ▶ What happens if “`apt install`” fails halfway through? => Inconsistent

Current state of affairs

- ▶ Install of one package, can break another, if you're not careful
- ▶ Nothing protects one package from editing the same folder as another package
- ▶ Can not easily rollback to previous versions of the system

Current state of affairs

- ▶ Declarative system configurations (Saltstack, Ansible, Puppet, chef) are hard to get right.
- ▶ They describe desired state, and try to keep your mutable environment consistent
- ▶ However, our hosts tend to diverge from our playbooks over time
- ▶ TODO: example

Current state of affairs

- ▶ What happens if an `apt install` fails half way? Because the power is out?
- ▶ You end up with a broken system, probably, maybe
- ▶ Due to editing things in place, there is no way to rollback, unless you have backups

Causes

- ▶ Packages are installed globally in the same place
- ▶ Packages are mutable

How about Docker?

- ▶ Isolates installing of dependencies per project
- ▶ Solves the problem of *global* mutability.
- ▶ My project won't break your project
- ▶ However, can still occur that you can build the project today, but can't build it tomorrow

How about Docker?

- ▶ What's can go wrong with this Dockerfile?

```
FROM php:7.0-fpm
RUN apt-get update && apt-get install -y \
    libfreetype6-dev \
    libjpeg62-turbo-dev \
    libmcrypt-dev \
    libpng-dev \
    libmemcached-dev zlib1g-dev \
    && docker-php-ext-install -j$(nproc) iconv mcrypt \
    && docker-php-ext-configure gd --with-freetype-dir=/usr/
    && docker-php-ext-install -j$(nproc) gd
    && pecl install memcached-2.2.0 \
    && docker-php-ext-enable memcached
```

So how do we fix this?

Shortly explain nix and what it does

- ▶ A language with similar syntax to json
- ▶ But with functions and templates
- ▶ Used to describe builds

Syntax

```
let x = { a = 3; b = 4; }  
in 9 + (if true then x.a else b.a)
```

Lets make a small derivation!

```
> gcc = import ./gcc.nix;  
> our-little-project =  
  derivation {  
    name = "our-little-project";  
    builder = builtins.toFile "builder.sh" ''  
      ${gcc}/bin/gcc -o our-project  our-project.c  
      '';  
    system = builtins.currentSystem; // ignore this. used 1  
  }  
  
:b our-little-project
```

But Arian, we use Docker for deployments!

- ▶ Great! that's really awesome. there's lots of tooling.
- ▶ Nix can build Docker images for you!

```
pkgs.dockerTools.buildImage {  
  name = "ing-auth";  
  tag  = "latest";  
  config = {  
    Cmd = [ "${pkgs.redis}/bin/redis-server" ];  
  };  
}
```

Explain Our

Our first derivation

Fin