NIXOS

Tristan Pinaudeau @ Capgemini

PRÉSENTATION

- **SRE** à Cdiscount
- **PENTESTER** à Capgemini

INTRODUCTION

CONSTATS

CONSTATS

- Inexhaustivité de la cartographie

CONSTATS

- Inexhaustivité de la cartographie
- Entropie des configurations

CONSTATS

- Inexhaustivité de la cartographie
- Entropie des configurations
- Gestion chaotique des patchs

CONSTATS

- Inexhaustivité de la cartographie
- Entropie des configurations
- Gestion chaotique des patchs
- Obscurité à l'audit

CONSTATS

- Inexhaustivité de la cartographie
- Entropie des configurations
- Gestion chaotique des patchs
- Obscurité à l'audit
- Automatisation complexe

SOLUTIONS EVENTUELLES

SOLUTIONS EVENTUELLES

- GPO / Scripts

SOLUTIONS EVENTUELLES

- GPO / Scripts
- Infrastructure As Code

SOLUTIONS EVENTUELLES

- GPO / Scripts
- Infrastructure As Code
- Containerisation

LE SYSTÈME PARFAIT EXISTE T-IL ?

LE SYSTÈME PARFAIT EXISTE T-IL ?

- Automatisable

LE SYSTÈME PARFAIT EXISTE T-IL ?

- Automatisable
- Versionnable

LE SYSTÈME PARFAIT EXISTE T-IL ?

- Automatisable
- Versionnable
- Liberté de configuration

LE SYSTÈME PARFAIT EXISTE T-IL ?

- Automatisable
- Versionnable
- Liberté de configuration
- Reproductibilité / Idempotance

LE SYSTÈME PARFAIT EXISTE T-IL ?

- Automatisable
- Versionnable
- Liberté de configuration
- Reproductibilité / Idempotance
- Bare Metal & Env. Virtualisé

FONCTIONNEMENT

NPM? = NIX PACKAGE MANAGER

The Purely Functional Software Deployment Model

Het puur functionele softwaredeploymentmodel
(met een samenvatting in het Nederlands)

Proefschrift ter verkrijging van de graad van doctor aan de Universiteit Utrecht op gezag van de Rector Magnificus, Prof. dr. W. H. Gispen, ingevolge het besluit van het College voor Promoties in het openbaar te verdedigen op woensdag 18 januari 2006 des middags te 12.45 uur

door

Eelco Dolstra

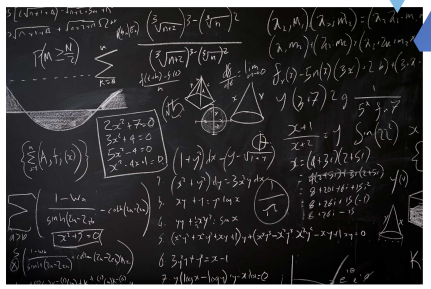
geboren op 18 augustus 1978, te Wageningen

~~PACKAGE~~ DERIVATIONS

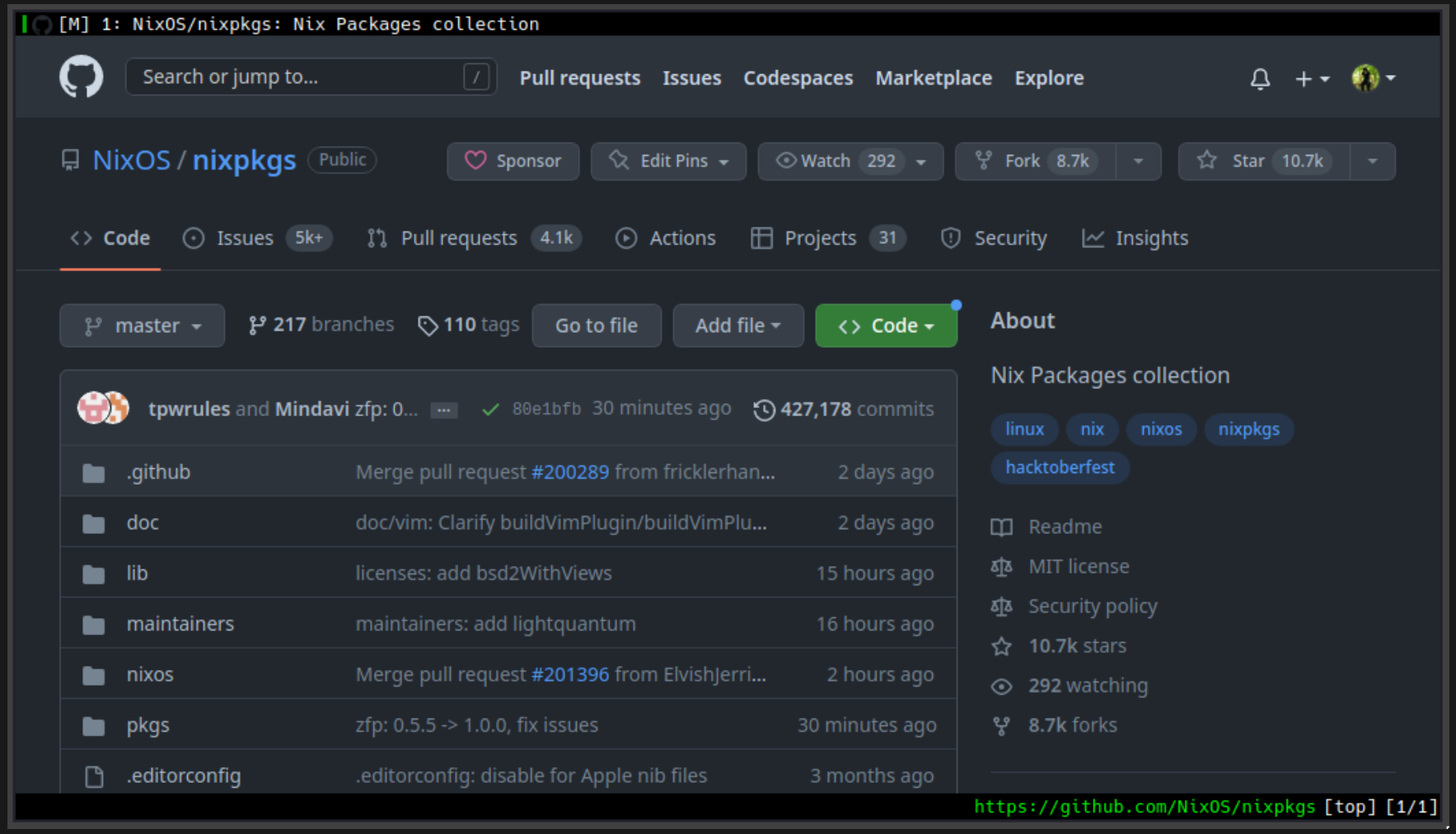
dpkg

dependancy hell

Derivations



MIRROIR MON BEAU MIRROIR...



NIX STORE

```
[tristan@master:~]$ tree /nix/store/ | head
/nix/store/
├── 0002rzpmcn4b7bq25qalw8j0v0vsl06c-ghidra.desktop.drv
├── 000iv8q3v5c9ipvq3jx2hvibnp9i9vma-source.drv
├── 001gp43bjqzx60cg345n2slzg7131za8-nix-nss-open-files.patch
├── 002jkych43y6k0y7x256hnq7vg78ijjx-lua-setup-hook.sh.drv
├── 00ix6n35smbd73xy16d0f9hkj0h7hafv-llvm-14.0.1-dev
│   ├── bin
│   │   └── llvm-config
│   ├── include
│   │   ├── llvm
```

VOUS FAITES LE LIEN ?

```
[tristan@demo:~]$ gcc --version
gcc (GCC) 11.3.0

[tristan@demo:~]$ which gcc
/home/tristan/.nix-profile/bin/gcc

[tristan@demo:~]$ ls -l /home/tristan/.nix-profile/bin/gcc
/home/tristan/.nix-profile/bin/gcc -> /nix/store/ykcrnkiicqg
```

VOUS FAITES LE LIEN ?

```
[tristan@demo:~]$ gcc --version
gcc (GCC) 11.3.0

[tristan@demo:~]$ which gcc
/home/tristan/.nix-profile/bin/gcc

[tristan@demo:~]$ ls -l /home/tristan/.nix-profile/bin/gcc
/home/tristan/.nix-profile/bin/gcc -> /nix/store/ykcrnkiicqg
```

VOUS FAITES LE LIEN ?

```
[tristan@demo:~]$ gcc --version
gcc (GCC) 11.3.0

[tristan@demo:~]$ which gcc
/home/tristan/.nix-profile/bin/gcc

[tristan@demo:~]$ ls -l /home/tristan/.nix-profile/bin/gcc
/home/tristan/.nix-profile/bin/gcc -> /nix/store/ykcrnkiicqg
```

CAS PRATIQUE

SITUATION INITIALE

```
├── docker-compose.yml
├── Makefile
└── template.env
```

"TALK IS CHEAP, ..."

```
{pkgs, fetchFromGitHub, ...}:
let
  core = "${pkgs.coreutils}/bin";
  argProjectName = "--project-name '$name'";
  argComposeFile = "--file '$src/docker-compose.yml'";
  dockercmd = "compose ${argProjectName} ${argComposeFile}
in
derivation {
  name = "docker-nextcloud";

  system = builtins.currentSystem;

  src = fetchFromGitHub {
    owner = "0b11stan";
    repo = "docker-nextcloud";
```

"TALK IS CHEAP, ..."

```
 2 let
 3   core = "${pkgs.coreutils}/bin";
 4   argProjectName = "--project-name '$name'";
 5   argComposeFile = "--file '$src/docker-compose.yml'";
 6   dockercmd = "compose ${argProjectName} ${argComposeFile}
 7 in
 8 derivation {
 9   name = "docker-nextcloud";
10
11   system = builtins.currentSystem;
12
13   src = fetchFromGitHub {
14     owner = "0b11stan";
15     repo = "docker-nextcloud";
16     rev = "main";
```

"TALK IS CHEAP, ..."

```
 9   name = "docker-nextcloud";
10
11   system = builtins.currentSystem;
12
13   src = fetchFromGitHub {
14     owner = "0b11stan";
15     repo = "docker-nextcloud";
16     rev = "main";
17     sha256 = "sha256-Sh+9Apb71QJHeShgaUbqLXQJMEjrBfkY/tW4Pi
18   };

19
20   builder = "${pkgs.bash}/bin/bash";
21
22   args = [ "-c"
```

"TALK IS CHEAP, ..."

```
  };

  builder = "${pkgs.bash}/bin/bash";

  args = [ "-c"
    ''
      ${core}/mkdir $out \
        && echo "${pkgs.docker}/bin/docker ${dockercmd}" \
        > $out/$name.sh \
        && ${core}/chmod +x $out/$name.sh
    ''
  ];
}
```

CONFIGURATIONS

```
{config, lib, pkgs, ...}:
let
  secretMySQLRootPassword = builtins.getEnv "MYSQL_ROOT_PASSWO
  secretMySQLPassword = builtins.getEnv "MYSQL_PASSWORD";
in {
  imports = [./hardware-configuration.nix];

  ...

  system.stateVersion = "22.05";
}
```

CONFIGURATION - NEXTCLOUD

```
1  nixpkgs.overlays = [(self: super: {
2    docker-nextcloud = super.callPackage ./docker-nextcloud.r
3  })];
4
5  environment.systemPackages = [pkgs.docker-nextcloud];
6
7  systemd.services.nextcloud = {
8    enable = true;
9    restartIfChanged = true;
10   wantedBy = ["multi-user.target"];
11   after = ["docker.service"];
12   bindsTo = ["docker.service"];
13   documentation = ["https://github.com/0b11stan/docker-next
14   script = "${pkgs.docker-nextcloud}/docker-nextcloud.sh";
15   environment = {
```

CONFIGURATION - NEXTCLOUD

```
1  nixpkgs.overlays = [(self: super: {
2    docker-nextcloud = super.callPackage ./docker-nextcloud.r
3  })];
4
5  environment.systemPackages = [pkgs.docker-nextcloud];
6
7  systemd.services.nextcloud = {
8    enable = true;
9    restartIfChanged = true;
10   wantedBy = ["multi-user.target"];
11   after = ["docker.service"];
12   bindsTo = ["docker.service"];
13   documentation = ["https://github.com/0b11stan/docker-next
14   script = "${pkgs.docker-nextcloud}/docker-nextcloud.sh";
15   environment = {
```

CONFIGURATION - NEXTCLOUD

```
 6
 7 systemd.services.nextcloud = {
 8   enable = true;
 9   restartIfChanged = true;
10   wantedBy = ["multi-user.target"];
11   after = ["docker.service"];
12   bindsTo = ["docker.service"];
13   documentation = ["https://github.com/0b11stan/docker-next
14   script = "${pkgs.docker-nextcloud}/docker-nextcloud.sh";
15   environment = {
16     MYSQL_ROOT_PASSWORD = secretMySQLRootPassword;
17     MYSQL_PASSWORD = secretMySQLPassword;
18   };

19 };
```

CONFIGURATION - DOCKER

```
virtualisation.docker.enable = true;
```

CONFIGURATION - SSH

```
services.openssh = {
  enable = true;
  passwordAuthentication = false;
  permitRootLogin = "no";
};
```

CONFIGURATION - RÉSEAU

```
networking = {
  hostName = "nixos-harden";
  networkmanager.enable = true;
  useDHCP = lib.mkDefault true;
  firewall = {
    enable = true;
    allowedTCPPorts = [8080 22];
  };
};
```

CONFIGURATION - USER

```
users.users = {
  tristan = {
    isNormalUser = true;
    extraGroups = ["wheel" "docker"];
    packages = [pkgs.neovim];
    openssh.authorizedKeys.keyFiles = [
      ./ssh-keys/silver-hp.pub
    ];
  };
};
```

RÉSULTATS

```
> wc -l src/*.nix src/*.sh

  69 src/configuration.nix
  34 src/docker-nextcloud.nix
  42 src/hardware-configuration.nix
   2 src/init.sh
 147 total
```

SYNTHÈSE

BONUS - ISOLATION LOGICIELLE

```
$ echo $PATH | tr ':' '\n'

  /run/wrappers/bin
  /home/tristan/.nix-profile/bin
  /etc/profiles/per-user/tristan/bin
  /nix/var/nix/profiles/default/bin
  /run/current-system/sw/bin
```

BONUS - NIX SHELL

```
[tristan@demo:rustproject]$ tail -n 2 shell.nix
  LIBPCAP_LIBDIR = "${pkgs.libpcap}/lib";
}

[tristan@demo:rustproject]$ echo $LIBPCAP_LIBDIR

[tristan@demo:rustproject]$ nix-shell
this path will be fetched (0.06 MiB download, 0.30 MiB unpa
  /nix/store/x8mymrkpsmpwyvqssjbwsq851kscf1kw-bash-interact
copying path '/nix/store/x8mymrkpsmpwyvqssjbwsq851kscf1kw-b

[nix-shell:rustproject]$ echo $LIBPCAP_LIBDIR
/nix/store/pby...ipx-libpcap-1.10.1/lib
```

BONUS - NIX SHELL

```
[tristan@demo:rustproject]$ tail -n 2 shell.nix
  LIBPCAP_LIBDIR = "${pkgs.libpcap}/lib";
}

[tristan@demo:rustproject]$ echo $LIBPCAP_LIBDIR

[tristan@demo:rustproject]$ nix-shell
this path will be fetched (0.06 MiB download, 0.30 MiB unpa
  /nix/store/x8mymrkpsmpwyvqssjbwsq851kscf1kw-bash-interact
copying path '/nix/store/x8mymrkpsmpwyvqssjbwsq851kscf1kw-b

[nix-shell:rustproject]$ echo $LIBPCAP_LIBDIR
/nix/store/pby...ipx-libpcap-1.10.1/lib
```

BONUS - NIX SHELL

```
[tristan@demo:rustproject]$ tail -n 2 shell.nix
  LIBPCAP_LIBDIR = "${pkgs.libpcap}/lib";
}

[tristan@demo:rustproject]$ echo $LIBPCAP_LIBDIR

[tristan@demo:rustproject]$ nix-shell
this path will be fetched (0.06 MiB download, 0.30 MiB unpa
  /nix/store/x8mymrkpsmpwyvqssjbwsq851kscf1kw-bash-interact
copying path '/nix/store/x8mymrkpsmpwyvqssjbwsq851kscf1kw-b

[nix-shell:rustproject]$ echo $LIBPCAP_LIBDIR
/nix/store/pby...ipx-libpcap-1.10.1/lib
```

BONUS - NIX SHELL

```
[tristan@demo:rustproject]$ tail -n 2 shell.nix
  LIBPCAP_LIBDIR = "${pkgs.libpcap}/lib";
}

[tristan@demo:rustproject]$ echo $LIBPCAP_LIBDIR

[tristan@demo:rustproject]$ nix-shell
this path will be fetched (0.06 MiB download, 0.30 MiB unpa
  /nix/store/x8mymrkpsmpwyvqssjbwsq851kscf1kw-bash-interact
copying path '/nix/store/x8mymrkpsmpwyvqssjbwsq851kscf1kw-b

[nix-shell:rustproject]$ echo $LIBPCAP_LIBDIR
/nix/store/pby...ipx-libpcap-1.10.1/lib
```

BONUS - ROOT EN READONLY

```
 1 $ DERIVATION=$(ls -tp /nix/store/ | grep 'openssh.*/$')
 2
 3 $ ls -l /nix/store/$DERIVATION/etc/ssh/
 4 total 504
 5 -r--r--r-- 2 root root 505489  1 janv.  1970 moduli
 6 -r--r--r-- 2 root root 1531  1 janv.  1970 ssh_config
 7 -r--r--r-- 2 root root 3226  1 janv.  1970 sshd_config
 8
 9 $ sudo chmod +w /nix/store/$DERIVATION/etc/ssh/sshd_config
10 [sudo] Mot de passe de tristan :
11 chmod: modification des droits [...] Read-only file system
```

BONUS - ROOT EN READONLY

```
$ DERIVATION=$(ls -tp /nix/store/ | grep 'openssh.*/$')

$ ls -l /nix/store/$DERIVATION/etc/ssh/
total 504
-r--r--r-- 2 root root 505489  1 janv.  1970 moduli
-r--r--r-- 2 root root 1531  1 janv.  1970 ssh_config
-r--r--r-- 2 root root 3226  1 janv.  1970 sshd_config

$ sudo chmod +w /nix/store/$DERIVATION/etc/ssh/sshd_config
[sudo] Mot de passe de tristan :
chmod: modification des droits [...] Read-only file system
```

BONUS - ROOT EN READONLY

```
$ DERIVATION=$(ls -tp /nix/store/ | grep 'openssh.*/$')

$ ls -l /nix/store/$DERIVATION/etc/ssh/
total 504
-r--r--r-- 2 root root 505489  1 janv.  1970 moduli
-r--r--r-- 2 root root 1531  1 janv.  1970 ssh_config
-r--r--r-- 2 root root 3226  1 janv.  1970 sshd_config

$ sudo chmod +w /nix/store/$DERIVATION/etc/ssh/sshd_config
[sudo] Mot de passe de tristan :
chmod: modification des droits [...] Read-only file system
```

LES INCONVENIENTS

LES INCONVENIENTS

- Moins "Flexible"

LES INCONVENIENTS

- Moins "Flexible"
- Croissances du Nix store

LES INCONVENIENTS

- Moins "Flexible"
- Croissances du Nix store
- Surcharge de Nixpkgs

LES INCONVENIENTS

- Moins "Flexible"
- Croissances du Nix store
- Surcharge de Nixpkgs
- Adoption = changement d'OS

CONCLUSION