NixOS @

nlnet FOUNDATION

How to use NixOS in a small organization

Jos van den Oever


wolvic
ALEKSIS
gosling.
technology
LibreOffice
ROSENPASS
Krill
Mobilizon
gnu net
BERT13
X-WIKI
postmarketOS
XR Fragment
ActivityPub
Lite X
Opaque Sphinx
TAURI
CokoDocs
coriolis
LibreCellular
slint
No Script
SCION
HYPERHYPER SPACE
Software Heritage
ForgeFed
offen.de Suir analytics.
Pixelfed
Libervia
TrenchBoot

We fund those who contribute to the open internet.

The Open Internet

- Communicate directly
- No dependencies, no lock-in
- Self-host or choose a trustworthy, local hoster

We fund open software, hardware, standards

Free Software
Free Society

NEXT GENERATION INTERNET

INTERNET OF HUMANS

Who likes to do system administration?

System administrator appreciation day

System administrator appreciation day

last Friday in July

How to use NixOS in a small organization?

- 10 employees
- external communication:
  - mail
  - website
  - telephone

Which parts are FOSS?

- ✓ website
- ✓ email server
- ✓ mailing lists
- ✓ code forge with CI
- ✓ grant management systems
- ✓ VPN
- ✓ chat
- ✓ video conferences
- ✓ microblogging
- ✓ shared calendar
- ✓ document server
- ✗ router
- ✗ printer
- ✗ fruity devices
- ✗ BIOS
- ✗ chips
- ✗ finances

Options

- NixOS
- Guix
- Closed cloud
- Open cloud

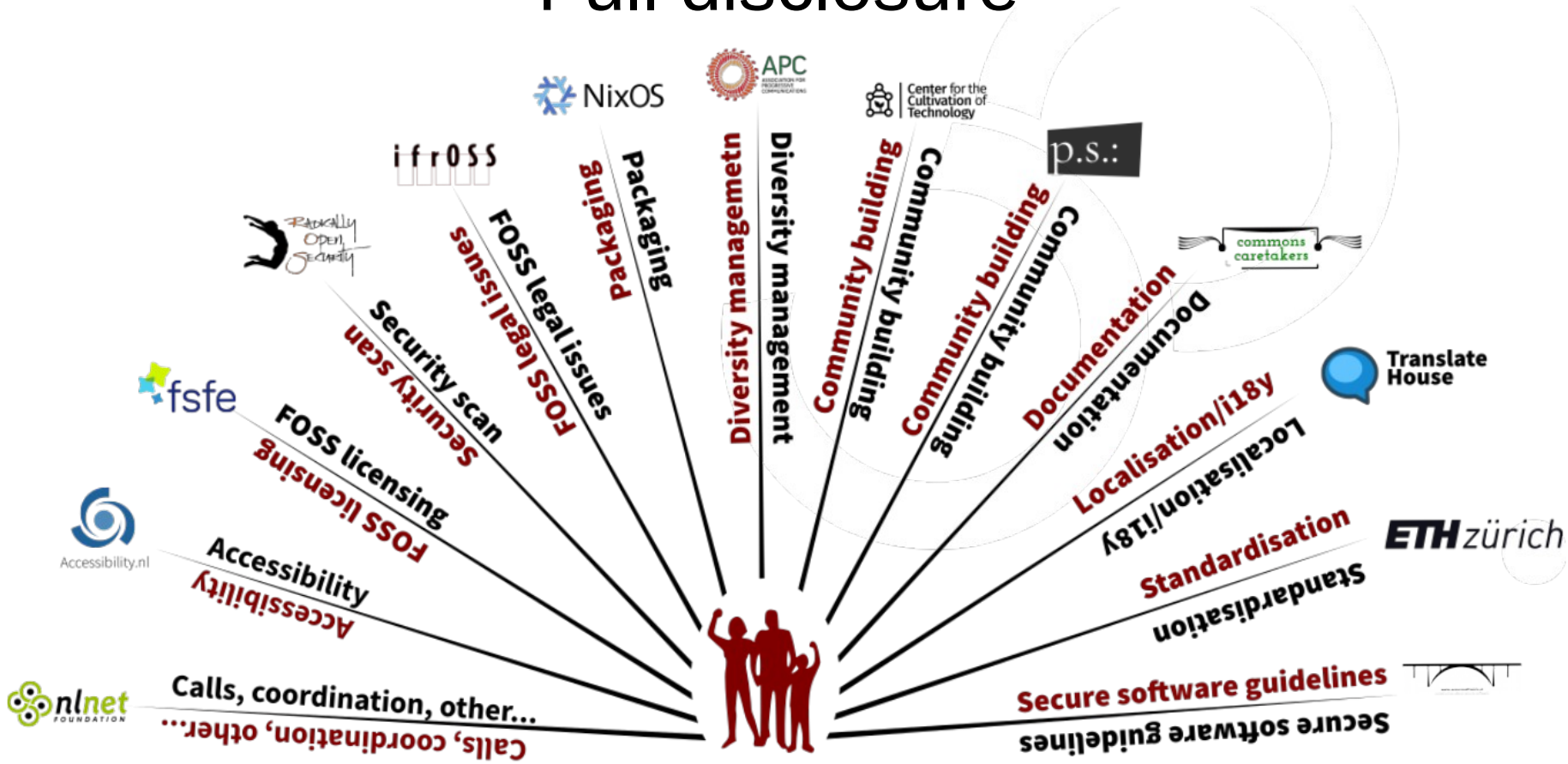
NixOS and NixPkgs

- ✓ declarative
- ✓ mostly reproducible
- ✓ many packages
- ✓ many services
- ✓ mix versions

- ✓ Nix language
- ✓ familiarity
- ✓ flake.lock
- ✓ proprietary packages disabled by default

- ✗ Microsoft GitHub
- ✗ careful where you tread
- ✗ no storage handling

Full disclosure



When nix clicked

Realization: Everything is a function.

YAML and JSON files are functions that take 0 arguments.

"Nix is a purely functional package manager."

```
{ hostname, definition }:
{ config, pkgs, lib, ... }:
{
  config = {
    networking = {
      hostName = hostname;
      domain = "nlnet.nl";
      nameservers = definition.nameservers;
    };
    environment.systemPackages = [ pkgs.neovim ];
  };
}
```

nix files → nixos-rebuild → running system

NixOS, but how?

https://www.lambda-solene.eu/

nixos-rebuild, krops, Cachix deploy, colmena, NixOps, Morph, NixUS, deploy-rs, Bento

Overview

- all systems defined in one git repository
- all machines in one flake.nix
- each machine has configuration.nix and hardware-configuration.nix
- imports/ for shared configuration
- machines.json for high level configuration

```
$ nix flake show
git+file:///home/nlnet/src/administrative?ref=refs/heads/
main&rev=5b90b83993f4fd1a2afed7b03648548b71b16c49
├───checks
│   └───x86_64-linux
│       ├───mailserver: derivation 'vm-test-run-ldap'
├───devShells
│   └───x86_64-linux
│       └───default: development environment 'nix-shell'
├───nixosConfigurations
│   ├───server001: NixOS configuration
│   ├───server002: NixOS configuration
│   ├───server003: NixOS configuration
│   ├───server004: NixOS configuration
│   ├───server005: NixOS configuration
└───packages
    └───x86_64-linux
        └───default: package 'develop'
```

nixos-rebuild switch -v --flake .#server001 --target-host server001

machines.json

```
{
  "server001": {
    "arch": "x86_64",
    "externalNetworkDevice": "ens3",
    "ip": "6.255.203.93",
    "gw": "6.255.203.1",
    "ipv6": "2a09:62c0:108:b1af::cafe",
    "gw6": "2a09:62c0:108::1",
    "nameservers": [
      "110.88.203.3",
      "2a03:3788:fff0:7::3"
    ],
    "keys": {
      "root": [
        "id_ed25519_server001_home",
        "repokey_server001_home",
        "alertManagerSecrets",
        "mailpwd",
        "wireguardPrivateKey"
      ]
    },
    "wireguard": {
      "ip": "10.100.123.1",
      "publicKey": "NjnULoR24NrLe9qIbknw1/q7CdmxEt5lXPx5dkcazwI="
```

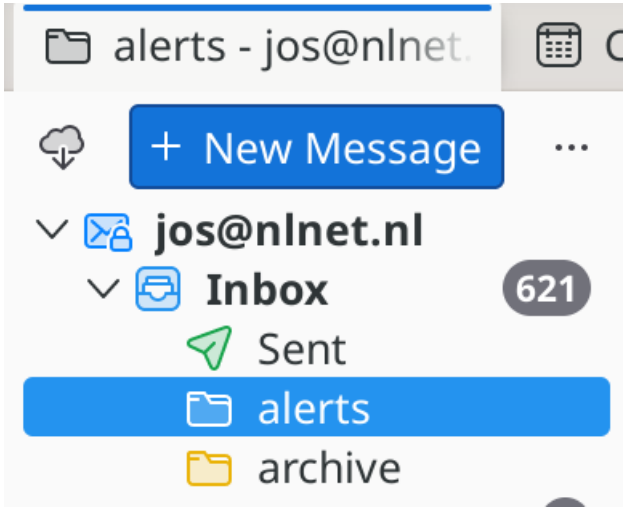
flake.nix, pt 2

```
{
  inputs = {
    nixpkgs.url = "github:NixOS/nixpkgs?ref=nixos-23.05";
    nixpkgs-23_11.url = "github:NixOS/nixpkgs?ref=nixos-23.11";=
    utils.url = "github:numtide/flake-utils";
    nlnet_forms.url =
      "git+ssh://gitlab@gitlab.nlnet.nl/NLnet/dashboard?ref=main";
    ngi0review.url =
      "git+https://codeberg.org/NGI0Review/ngi0review.git?ref=main";
    mailserver = {
      url =
        "git+https://gitlab.com/vandenoever/nixos-mailserver.git?
        ref=combined&rev=398d85ee9c1de0d1b0ec649fdbac3d858abb7d85";
    };
  };
  outputs = { self, nixpkgs, nixpkgs-23_11, utils, nlnet_forms, ngi0review
    , mailserver }:
```

flake.nix, pt 2

```
outputs = { self, nixpkgs, nixpkgs-23_11, utils, nlnet_forms, ngi0review
  , mailserver }:
  let
    systems = [ "x86_64-linux" "aarch64-linux" ];
    lib = nixpkgs.lib;
    machines = builtins.fromJSON (builtins.readFile ./machines.json);
    constants = import ./constants.nix;
    imports = [
      (import ./imports/version.nix self)
      (import ./imports/users.nix)
    ];
    mkSystem = hostname: definition:
      lib.nixosSystem {
        system = definition.arch + "-linux";
        modules = [
          (import (./hosts + ("/" + hostname + "/configuration.nix"))
            hostname definition)
        ];
      };
  in {
    nixosConfigurations = lib.mapAttrs mkSystem machines;
  };
```

Alerts



Prometheus

Inactive (12) Pending (0) Firing (0)

Filter by name or labels

Show annotations

/nix/store/pm6rbignrajadc2f1ll7m60317bq2ggb-rules-checkrules-checked > basic — inactive

- **node_down** (0 active)
- **disk_full** (0 active)
- **disk_error** (0 active)
- **failing_systemd_unit** (0 active)
- **node_textfile_scrape_error** (0 active)
- **nixos_flake_committed** (0 active)
- **postfix_up** (0 active)
- **openldap** (0 active)
- **probe** (0 active)
- **ssl_expiration** (0 active)
- **dnssec_expiration** (0 active)

/nix/store/pm6rbignrajadc2f1ll7m60317bq2ggb-rules-checkrules-checked > borgbackup — inactive

- **backup_age** (0 active)

- **node_down** (0 active)
- **disk_full** (0 active)
- **disk_error** (0 active)
- **failing_systemd_unit** (0 active)
- **node_textfile_scrape_error** (0 active)
- **nixos_flake_committed** (0 active)
- **postfix_up** (0 active)
- **openldap** (0 active)
- **probe** (0 active)
- **ssl_expiration** (0 active)
- **dnssec_expiration** (0 active)

/nix/store/pm6rbignrajadc2f1ll7m60317bq2ggb-rules-checkrules-checked > borgbackup inactive

Backups

- Borg for backups
- btrbk for snapshots

- NixOS is great at handling the software setup,
- It has no notion of storage location
  - Have to repeat setup
  - Top-level directory definitions that reused

Mail

- Of course we self-host
- Dovecot, Postfix, LDAP, rspamd
- Paid for LDAP support to be added

Stalwart (stalw.art)
Mox (xmox.nl)

Internet.nl
IS YOUR INTERNET UP TO DATE?

Email test: nlnet.nl

Congratulations, your domain will be added to the **Hall of Fame** soon!

- Reachable via modern internet address (IPv6)
- All domain names signed (DNSSEC)
- Authenticity marks against email phishing (DMARC, DKIM and SPF)
- Mail server connection sufficiently secured (STARTTLS and DANE)
- Authorised route announcement (RPKI)

Explanation of test report

Permalink test result (2024-02-03 22:01 UTC)

Seconds until retest option: 119

Testing

- NixOS includes excellent integration testing tools
- Python scripts to bring up machines and make them interact
- NixPkgs repository has many examples
- Part of CI via flake checks

Oops

```
nixos-rebuild switch -v --flake .#server001 \
                        --target-host server002
```

Updates

```
nix flake lock --update-input nlnet_forms
```

Conclusions

- Keep it simple → use the basic tools
- Put most configuration in json files

- NixOS is technically great for NLnet
- Too complex for an average office
- Opportunity for open cloud providers

NGI Fediversity

Creating the hosting stack of the future

Questions? Tips?

DNS

Secrets

Wireguard