ZERO TRUST INNOVATORS

CYBERSECURITY Meetup

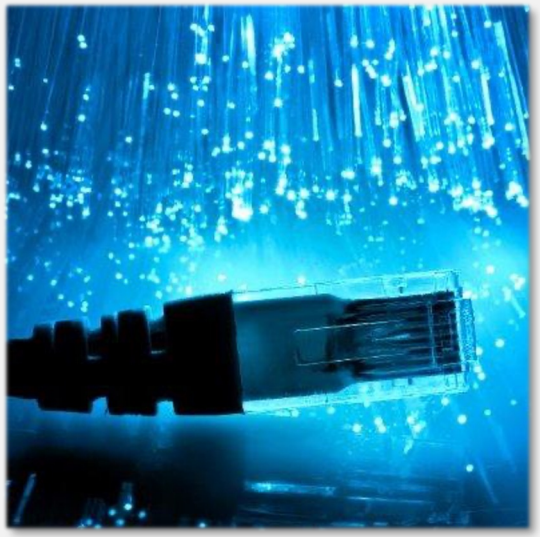
Hands-on science for adults

Agenda

- Introduction
- Nix, nix and NixOS
- Nix store
- Why?
- Diner
- Labs

Introduction

- Who am I?
- Who are you?
- Amenities
- Diner



Jeroen Simonetti

https://github.com/jsimonetti

Security Engineer/Architect and
Zero Trust Advisor @ ON2IT B.V.

Nix, nix and NixOS

- Nix – the package manager
- nix – the expression language
- NixOS – the distribution

Nix, nix and NixOS

- Nix – the package manager
  - Purely functional (uses the nix expression language to define packages)
  - Software installed in unique directories
  - Created June 15, 2003; 19 years ago; by Eelco Dolstra (PhD thesis subject, Faculty of Science, Utrecht, The Netherlands)
  - Runs on Linux (i686, x86_64, aarch64) and macOS (x86_64, aarch64)

GNU Guix is a similar alternative to Nix (used by the GNU Guile distribution)

Nix, nix and NixOS

- Nix – the expression language

  - pure, lazy, functional language
    - pure: no side-effects like variable assignment
    - lazy: arguments to functions are evaluated only when they are needed
    - functional: functions are “normal” values that can be passed around and manipulated
  - not a full-featured, general purpose language (describe packages, compositions of packages, and the variability within packages)
  - designed for Nix (the package manager)

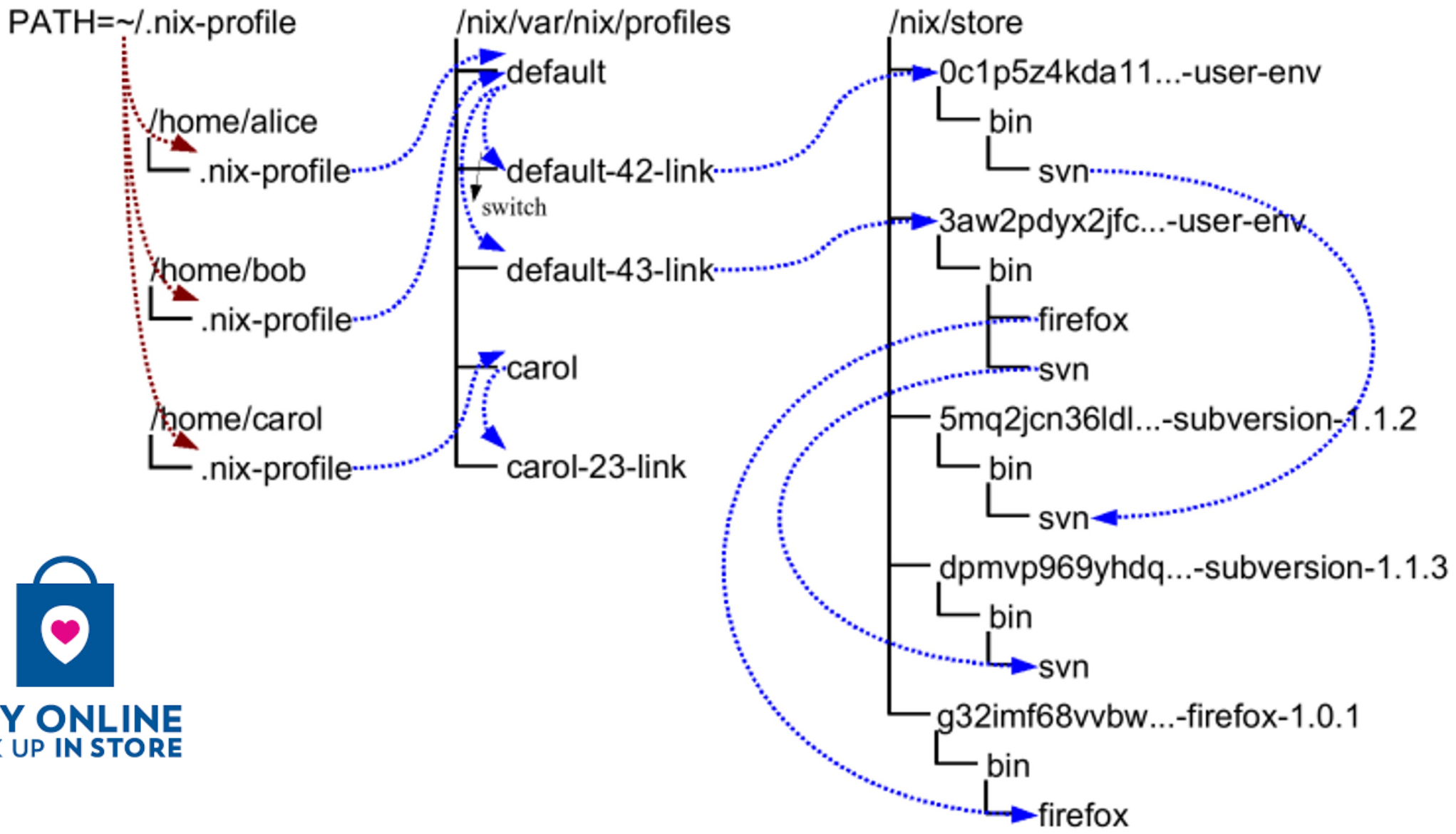
Nix, nix and NixOS

- NixOS – the distribution
  - Uses Nix (the package manager) for package management (currently over 80.000 packages exist in the nixpkgs repository)
  - Uses nix (the language) to manage the OS configuration.
  - Uses modules (built with the nix language) to ease configuration and built the OS root filesystem
  - Has the concept of generations
  - Entire OS configuration is (usually) done in `/etc/nixos/configuration.nix`
  - Booting NixOS only requires `/boot` and `/nix` to exist (`/` (root) is created dynamically with symlinks)

Nix store

- Located at `/nix/store`
- Contains *everything*
- Uses symbolic links to 'built' your generations
  - Quickly 'switch' to a new generation just by changing a few links (`/run/current-system`)
- Packages can be shared between systems
  - Serve nix store over HTTP, SSH, S3
  - Copy packages manually using SSH
  - (or be creative with container image layers)
- Garbage collection and optimisation (hardlinking) available to keep it lean.

Nix store



Why?

- Configure your OS configurative declaratively, using nix expressions
- 'Desired state' configuration for your system
- Nix makes making mistakes cheap (atomic upgrades and rollbacks using generations)
- Makes your entire system reproducable (just rebuild your last `configuration.nix`)
- Your entire system can now be (re)stored using a VCS like git (with all added benefits)
- Run multiple software versions / variations simultaneously
  - Extremely helpful for development environments to prevent impurity leaking from the OS into the software production

Diner

Bon Appétit.

Labs

- Configure your system in `/etc/nixos/configuration.nix` (unless otherwise stated)
- Use a text editor to modify your configuration (both `vi` and `nano` are available by default)
- Reference https://search.nixos.org/options to find module documentation
- Use command `nixos-rebuild switch` to activate your configuration changes

- Your lab machine is accessible at ssh://lab##.meetup.on2it.io
- The console is accessible at ssh://labuser##@console.meetup.on2it.io (this simulates your keyboard/mouse)

- Entire lab configuration, including lab guide and slides is available at https://github.com/on2itsecurity/meetup-nixos

LAB0

- Use our steppingstone to reach your lab
- Please check if you can reach your assigned lab and login to it's console

https://github.com/on2itsecurity/meetup-nixos
machine: ssh://lab##.meetup.on2it.io
console: ssh://labuser##@console.meetup.on2it.io

LAB1 - Configuring SSH

- Hints
  - make changes in `/etc/nixos/configuration.nix`
  - use https://search.nixos.org/options to search for '`openssh`'
  - use `nixos-rebuild switch` to activate your configuration

https://github.com/on2itsecurity/meetup-nixos
machine: ssh://lab##.meetup.on2it.io
console: ssh://labuser##@console.meetup.on2it.io

LAB2 - Upgrading NixOS

- Hints 2.1
  - make changes in `/etc/nixos/configuration.nix`
  - use https://search.nixos.org/options to search for '`systemPackages`'
  - use `nixos-rebuild switch` to activate your configuration

- Hint 2.2
  - make changes in `/etc/nixos/flake.nix`

https://github.com/on2itsecurity/meetup-nixos
machine: ssh://lab##.meetup.on2it.io
console: ssh://labuser##@console.meetup.on2it.io

LAB3 - Breaking stuff

- Hint 3.1
  - use https://search.nixos.org/options to search for '`useDHCP`'
- Hint 3.2
  - use the `nixos-rebuild` command *(possibly use CTRL+C to stop the command if it hangs)*
- Hint 3.3
  - use the `nix-env --list-generations --profile /nix/var/nix/pr` command

  (should this hang, please alert the host)

https://github.com/on2itsecurity/meetup-nixos
machine: ssh://lab##.meetup.on2it.io
console: ssh://labuser##@console.meetup.on2it.io

LAB4 - Using secrets

- Hint
  - this is really advanced stuff. Don't be afraid to 'show the answer'

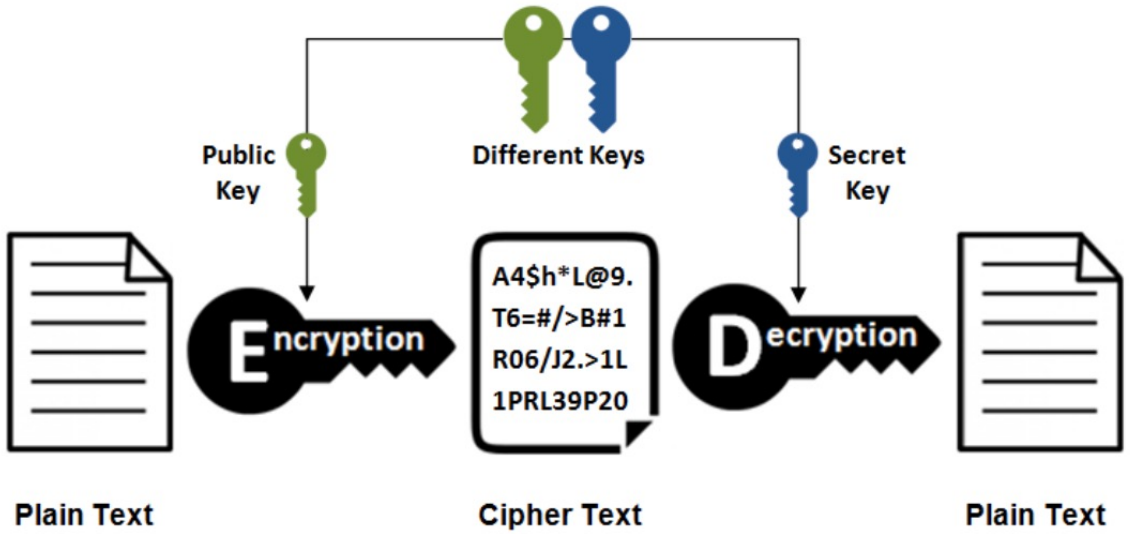
- Hint 4.1
  - use https://search.nixos.org/options to search for '`environment.etc`'

- Hint 4.2
  - use https://github.com/ryantm/agenix#tutorial for reference

https://github.com/on2itsecurity/meetup-nixos
machine: ssh://lab##.meetup.on2it.io
console: ssh://labuser##@console.meetup.on2it.io

LAB5 – Introduction

- Age uses asymmetric encryption based on SSH public and private keys.
- Agenix is a NixOS module that can use Age to store encrypted files on the filesystem and only decrypt then (into RAM) at runtime.



https://github.com/on2itsecurity/meetup-nixos
machine: ssh://lab##.meetup.on2it.io
console: ssh://labuser##@console.meetup.on2it.io

LAB5 – Home-manager

- Hints
  - Home-manager config is located in `.config/nixpkgs/home.nix`
  - this is really advanced stuff. Don't be afraid to 'show the answer'
  - Use `` `home-manager switch` `` to activate your user configuration

- Hint 5.1
  - the bootstrap is for the 'meetup' user.

- Hint 5.2
  - use https://rycee.gitlab.io/home-manager/options.html for reference

https://github.com/on2itsecurity/meetup-nixos
machine: ssh://lab##.meetup.on2it.io
console: ssh://labuser##@console.meetup.on2it.io

Next meetup

CYBERSECURITY Meetup

Next meetup is at 12 December (provisional date)

See you there!


ZERO TRUST INNOVATORS