

Home Manager 101



Jakub Vokoun

jakub.vokoun@wftech.cz

Motivace

- **reprodukovatelná a deklarativní** správa aplikací a jejich konfigurací na úrovni uživatele
- relativně velké množství aplikací má built-in konfiguraci přímo v Home Manageru
- *zbytek* lze konfigurovat pomocí i dynamicky generovaných konfiguračních souborů, které Home Manager spravuje
- Home Manager zvládá i komplexní konfigurace:
 - `nixvim` pro Neovim
 - KDE Plasma
- nejsme limitováni jen NixOS, Home Manager můžeme použít na téměř jakékoliv jiné distribuci GNU/Linuxu nebo třeba na macOS

Nix 101

Jazyk Nix

- dynamicky typovaný
- funkcionální
- lazy evaluation
- DSL (Domain Specific Language)

Nix 101

Vyhledávání

```
nix search nixpkgs ripgrep
```

Spuštění programu bez nutnosti instalce

```
nix run nixpkgs#cowsay Hello Linux Days 2024!
```

Nix 101

Spuštění programu bez nutnosti instalce II

```
curl -s https://api.github.com/users/jakubvokoun | nix run nixpkgs#jq '.name'
```

Nix shell

```
nix-shell -p cowsay  
cowsay Hello Linux Days 2024!
```

Nix 101

Flakes

Nix flakes provide a standard way to write Nix expressions (and therefore packages) whose dependencies are version-pinned in a lock file, improving reproducibility of Nix installations. (<https://nixos.wiki/wiki/Flakes>)

- `flake.nix`
- `flake.lock`

Nix 101

Flakes

```
nix flake init
```

Nix 101

Flakes

```
# flake.nix
{
  description = "A very basic flake";

  inputs = {
    nixpkgs.url = "github:nixos/nixpkgs?ref=nixos-unstable";
  };

  outputs = { self, nixpkgs }: {
    packages.x86_64-linux.hello = nixpkgs.legacyPackages.x86_64-linux.hello;
    packages.x86_64-linux.default = self.packages.x86_64-linux.hello;
  };
}
```


Nix 101

Flakes

```
# flake.nix
{
  description = "Some legacy PHP app";
  inputs = {
    nixpkgs.url = "github:nixos/nixpkgs?ref=nixos-unstable";
    phps.url = "github:fossar/nix-phps";
  };
  outputs = { self, nixpkgs, phps, ... }: {
    devShell.x86_64-linux = nixpkgs.legacyPackages.x86_64-linux.mkShell {
      buildInputs = [
        phps.packages.x86_64-linux.php56
      ];
    };
  };
}
```

Nix 101

Flakes

```
nix develop
```

Instalce - NixOS

```
nix-channel --add \
  https://github.com/nix-community/home-manager/archive/release-24.05.tar.gz \
  home-manager
nix-channel --update
nix-shell '<home-manager>' -A install
```

Instalce - Ubuntu

Nix

```
sh <(curl -L https://nixos.org/nix/install) --daemon
```

Home Manager

```
nix-channel --add \
  https://github.com/nix-community/home-manager/archive/release-24.05.tar.gz \
  home-manager
nix-channel --update
nix-shell '<home-manager>' -A install
```

Instalce - macOS

Nix

```
curl --proto '=https' --tlsv1.2 -sSf -L \
  https://install.determinate.systems/nix | sh -s -- install
```

Home Manager

```
nix-channel --add \
  https://github.com/nix-community/home-manager/archive/release-24.05.tar.gz \
  home-manager
nix-channel --update
```

```
imports = [ <home-manager/nix-darwin> ];
```

Konfigurace - NixOS

```
# /etc/nixos/configuration.nix
{ config, pkgs, ... }:

{
  nix.settings = {
    # Enable flakes and new 'nix' command
    experimental-features = "nix-command flakes";
    # Deduplicate and optimize nix store
    auto-optimize-store = true;
    # Allowed users - needed for Home manager
    allowed-users = [ "@wheel" ];
  };
}
```

Minimální konfigurace

```
# ~/.config/home-manager/home.nix
{ config, pkgs, ... }:

{
  home.username = "vagrant";
  home.homeDirectory = "/home/vagrant";
  home.stateVersion = "24.05";
  home.packages = [];
  home.file = {};
  home.sessionVariables = {};

  programs.home-manager.enable = true;
}
```

Modul v `configuration.nix`

```
sudo nix-channel --add \  
    https://github.com/nix-community/home-manager/archive/release-24.05.tar.gz \  
    home-manager  
sudo nix-channel --update
```


Modul v `configuration.nix`

```
# /etc/nixos/configuration.nix

# ...
imports = [ <home-manager/nixos> ];

# ...
users.users.eve.isNormalUser = true;
home-manager.users.eve = { pkgs, ... }: {
  home.packages = [ pkgs.atool pkgs.httpie ];
  programs.bash.enable = true;
  home.stateVersion = "24.05";
};
```

Začínáme

NixOS

```
nix-shell -p git
```

Ubuntu

```
sudo apt -y install git
```

Konfigurace

```
vim ~/.config/home-manager/home.nix  
home-manager switch
```

Instalace programů

```
# ~/.config/home-manager/home.nix
{ config, pkgs, ... }:

{
  programs.git = {
    enable = true;
    userName = "Jakub Vokoun";
    userEmail = "jakub.vokoun@wfttech.cz";
  };

  home.packages = [
    mc
  ];
}
```

Vlastní konfigurace

```
{ config, pkgs, ... }:  
  
{  
  home.file = {  
    "test.txt" = {  
      text = '  
        Hi there!  
      ';  
      executable = false;  
    };  
  };  
};  
}
```

Derivace a rollback

Výpis existujících deravací

```
home-manager generations
```

```
2024-10-11 13:24 : id 139 -> /nix/store/ycy5bhyw9a93iibc0sfd6bp7ws4r7v75-home-manager-generation
2024-10-11 13:22 : id 138 -> /nix/store/3fn6v8hnv0qrvrddcd34xpfk11jdnhmnw-home-manager-generation
2024-10-09 15:27 : id 137 -> /nix/store/gzrql8nazarqw6x16b18cl6vc8lACLSS-home-manager-generation
2024-10-02 16:28 : id 136 -> /nix/store/mpisjz6944a0039cgashqzafjxs7b5q-home-manager-generation
2024-10-02 10:11 : id 135 -> /nix/store/crzianrjzm25vkq1ir4anlqp5va0ldb1-home-manager-generation
```

Rollback k jedné z minulých generací

```
/nix/store/3fn6v8hnv0qrvrddcd34xpfk11jdnhmnw-home-manager-generation/activate
```

Mazání starých generací

Odstraň všechny generace, které jsou starší než 15 dní

```
home-manager expire-generations '-15 days'
```

Nix garbage collector

```
sudo nix-collect-garbage --delete-older-than 15d
```

systemd unity

```
{
  inputs,
  lib,
  config,
  pkgs,
  ...
}: {
  systemd.user.services.foo = {
    Unit = { Description = "Foo service"; };

    Service = {
      Type = "oneshot";
      WorkingDirectory = "/home/jakub/foo-app";
      ExecStart = "/home/jakub/Work/foo-app/foo run";
    };
  };
}
```

systemd unity

```
{
  inputs,
  lib,
  config,
  pkgs,
  ...
}: {
  systemd.user.timers.foo = {
    Unit = { Description = "Foo timer"; };

    Timer = {
      OnCalendar = "*-*-* *:00:00";
      Unit = "foo.service";
    };

    Install = { WantedBy = [ "timers.target" ]; };
  };
}
```


NixVim

```
{ pkgs, lib, ... }:  
let  
  nixvim = import (builtins.fetchGit {  
    url = "https://github.com/nix-community/nixvim";  
    ref = "nixos-24.05";  
  });  
in {  
  imports = [ nixvim.homeManagerModules.nixvim ];  
  
  programs.nixvim = {  
    enable = true;  
    enableMan = true;  
  
    colorschemes.tokyonight.enable = true;  
  };  
}
```

Plasma Manager

```
{ pkgs, ... }:  
{  
  imports = [ <plasma-manager/modules> ];  
  programs.plasma = {  
    enable = true;  
    workspace = {  
      clickItemTo = "select";  
      lookAndFeel = "org.kde.breezedark.desktop";  
      cursor.theme = "Bibata-Modern-Ice";  
      iconTheme = "Papirus-Dark";  
      wallpaper = "${pkgs.kdePackages.plasma-workspace-wallpapers}/share/wallpapers/Patak/contents/images/1080x1920.png";  
    };  
    hotkeys.commands."launch-konsole" = {  
      name = "Launch Konsole";  
      key = "Meta+Alt+K";  
      command = "konsole";  
    };  
  };  
};  
}
```

Reference

- <https://nix-community.github.io/home-manager/>
- <https://nix-community.github.io/nixvim/>
- <https://nix-community.github.io/plasma-manager/>
- <https://home-manager-options.extranix.com/>
- <https://github.com/Misterio77/nix-starter-configs>
- <https://github.com/jakubvokoun/nixos-config>

Demo

- NixOS + standalone Home Manager
- nixvim