

Core Philosophy

NixOS is a Linux distribution built on the Nix package manager, emphasizing declarative configuration, reproducibility, and atomic upgrades/rollbacks.

- Declarative: The entire system state is defined in a single file (`/etc/nixos/configuration.nix`).
- Immutable: Packages are stored in a read-only `/nix/store` based on their hash.
- Generation-based: Every change creates a new system "generation" that you can roll back to.

System Configuration

- File location: `/etc/nixos/configuration.nix`.
- Apply Changes: `sudo nixos-rebuild switch`.
- Test Changes (without saving to boot menu): `sudo nixos-rebuild test`.
- Build and set as default for next boot: `sudo nixos-rebuild boot`.

Package Management (Nix)

- Search: `nix-env -qaP [query]` or use the web search at search.nixos.org.
- Temporary Install: `nix-shell -p [package]` (Run a shell with the package available).
- Modern Search/Shell (Flakes): `nix shell nixpkgs#[package]`.
- List Generations: `nix-env --list-generations`.
- Garbage Collection (Free space): `sudo nix-collect-garbage -d`.

User and System Maintenance

- Update Channels: `sudo nix-channel --update`.
- Rollback:
 - Inside system: `sudo nixos-rebuild switch --rollback`.
 - At Boot: Select a previous generation from the boot menu.
- User-specific packages: Use `home-manager` (Recommended over `nix-env`).

Configuration Example

```
{ config, pkgs, ... }:

{
  imports = [ ./hardware-configuration.nix ];

  # System packages
  environment.systemPackages = with pkgs; [
    vim git wget firefox
  ];

  # Networking
  networking.hostName = "nixos-pc";
  networking.networkmanager.enable = true;

  # Services
  services.openssh.enable = true;
  services.xserver.enable = true;

  services.xserver.displayManager.gdm.enable = true;

  services.xserver.desktopManager.gnome.enable = true;

  system.stateVersion = "23.11";
}
```

Flakes (Modern Nix)

Flakes provide a standardized way to manage dependencies and reproducibility.

- Create: `nix flake init`.
- Update: `nix flake update`.
- Apply: `sudo nixos-rebuild switch --flake .#[hostname]`.

Pro Tips

Generational Safety

Never be afraid of a system update. If it breaks, simply reboot and select the previous working version from the boot menu.

nix-shell for Development

Use `nix-shell` or `direnv` with `nix-direnv` to automatically load project dependencies (compilers, libraries) whenever you enter a directory.

home-manager

Separate your user configuration (dotfiles, user apps) from the system configuration using Home Manager for better organization and portability.