

# Simplifying NixOS-Related Commands

To simplify NixOS-related commands, I utilize [just](#), which proves to be very convenient.

Alternatively, you can also use similar tools like Makefile or [cargo-make](#) for this purpose.

Here, I will provide my approach as a reference.

Below is an example of how my Justfile looks:

The latest Justfile I'm using: [ryan4yin/nix-config/Justfile](#)

```

1  # just is a command runner, Justfile is very similar to Makefile, but sim
2
3  #####
4  #
5  # Nix commands related to the local machine
6  #
7  #####
8
9  deploy:
10     nixos-rebuild switch --flake . --use-remote-sudo
11
12  debug:
13     nixos-rebuild switch --flake . --use-remote-sudo --show-trace --verbose
14
15  up:
16     nix flake update
17
18  # Update specific input
19  # usage: make upp i=home-manager
20  upp:
21     nix flake update $(i)
22
23  history:
24     nix profile history --profile /nix/var/nix/profiles/system
25
26  repl:
27     nix repl -f flake:nixpkgs
28
29  clean:
30     # remove all generations older than 7 days
31

```

```

32     sudo nix profile wipe-history --profile /nix/var/nix/profiles/system --older-
33
34 gc:
35     # garbage collect all unused nix store entries
36     sudo nix-collect-garbage --delete-old
37
38 #####
39 #
40 # Idols, Commands related to my remote distributed building cluster
41 #
42 #####
43
44 add-idols-ssh-key:
45     ssh-add ~/.ssh/ai-idols
46
47 aqua: add-idols-ssh-key
48     nixos-rebuild --flake .#aquamarine --target-host aquamarine --build-host aquam
49
50 aqua-debug: add-idols-ssh-key
51     nixos-rebuild --flake .#aquamarine --target-host aquamarine --build-host aquam
52
53 ruby: add-idols-ssh-key
54     nixos-rebuild --flake .#ruby --target-host ruby --build-host ruby switch --use
55
56 ruby-debug: add-idols-ssh-key
57     nixos-rebuild --flake .#ruby --target-host ruby --build-host ruby switch --use
58
59 kana: add-idols-ssh-key
60     nixos-rebuild --flake .#kana --target-host kana --build-host kana switch --use
61
62 kana-debug: add-idols-ssh-key
63     nixos-rebuild --flake .#kana --target-host kana --build-host kana switch --use
64
65 idols: aqua ruby kana
66
67 idols-debug: aqua-debug ruby-debug kana-debug

```

By Save this [Justfile](#) to the root directory of your Nix flake. Then, I can use [just deploy](#) to deploy the configuration to my local machine, and [just idols](#) to deploy the configuration to all my remote servers.

This approach simplifies the execution of NixOS commands by abstracting them behind target names in the Justfile, providing a more user-friendly and convenient experience.

---

Loading comments...