MatthewCroughan /
nixcfg

<> **Code**   ⊙ Issues   ⑂ Pull requests   2   ▶ Actions   ⊞ Projects   ⊘ Security   ⊯ In

nixcfg / hosts / t480 / **configuration.nix** ⧉   ···

🧑 **MatthewCroughan** stop using tmp.useTmpfs on some machines   d66d3a7 · 2 years ago   🕘

157 lines (144 loc) · 3.96 KB

```nix
1    { config, pkgs, inputs, ... }:
2    {
3      imports = with inputs.self.nixosModules; [
4        ./disks.nix
5        ./hardware-configuration.nix
6        users-matthewcroughan
7        profiles-tailscale
8        profiles-sway
9        profiles-fail2ban
10   #    profiles-steam
11        profiles-wireless
12        profiles-pipewire
13        profiles-avahi
14        mixins-obs
15        mixins-v4l2loopback
16        mixins-common
17        mixins-i3status
18        mixins-fonts
19        mixins-bluetooth
20        mixins-vaapi-intel-hybrid-codec
21        mixins-zram
22        editor-nvim
23      ];
24
25      nixpkgs.overlays = [
26        (self: super: {
27          mpv-unwrapped = super.mpv-unwrapped.override { ffmpeg_5 = super.ffmp
28          sway-unwrapped = super.sway-unwrapped.override { stdenv = super.with
29          kitty = super.kitty.override { stdenv = super.withCFlags [ "-funroll
30          nixUnstable = super.nixUnstable.override { stdenv = super.withCFlags
31        })
32      ];
33
34      home-manager = {
35        useGlobalPkgs = true;
36        useUserPackages = true;
37        users = import "${inputs.self}/users";
```

```nix
37    users = import ${inputs.self}/users;
38        extraSpecialArgs = {
39          inherit inputs;
40          headless = false;
41        };
42      };
43
44      users.users.matthew.extraGroups = [ "video" ];
45
46      nix = {
47        # From flake-utils-plus
48        generateNixPathFromInputs = true;
49        generateRegistryFromInputs = true;
50        linkInputs = true;
51      };
52
53      networking = {
54        firewall = {
55          # Syncthing ports
56          allowedTCPPorts = [ 22000 ];
57          allowedUDPPorts = [ 21027 22000 ];
58        };
59        hostName = "t480";
60        useNetworkd = true;
61        wireless = {
62          userControlled.enable = true;
63          enable = true;
64          interfaces = [ "wlp3s0" ];
65        };
66        useDHCP = false;
67        interfaces = {
68          "enp0s31f6".useDHCP = true;
69          "wlp3s0".useDHCP = true;
70        };
71      };
72
73      services = {
74        udev.extraRules = ''
75          # Nreal Air Glasses
76          SUBSYSTEMS=="usb", ATTRS{idVendor}=="3318", ATTRS{idProduct}=="0424'
77          # Gamecube Controller Adapter
78          SUBSYSTEM=="usb", ENV{DEVTYPE}=="usb_device", ATTRS{idVendor}=="057€
79          # Xiaomi Mi 9 Lite
80          SUBSYSTEM=="usb", ENV{DEVTYPE}=="usb_device", ATTRS{idVendor}=="05c€
81          SUBSYSTEM=="usb", ENV{DEVTYPE}=="usb_device", ATTRS{idVendor}=="2717
82        '';
83        thermald.enable = true;
84        tlp = {
85          enable = true;
86          settings = {
87            PCIE_ASPM_ON_BAT = "powersupersave";
88            CPU_SCALING_GOVERNOR_ON_AC = "performance";
89            CPU_SCALING_GOVERNOR_ON_BAT = "powersave";
```

```nix
 90              CPU_MAX_PERF_ON_AC = "100";
 91              CPU_MAX_PERF_ON_BAT = "30";
 92              STOP_CHARGE_THRESH_BAT1 = "95";
 93              STOP_CHARGE_THRESH_BAT0 = "95";
 94            };
 95          };
 96          logind.killUserProcesses = true;
 97        };
 98
 99      boot = {
100        kernelPackages = config.boot.zfs.package.latestCompatibleLinuxPackages
101        kernelParams = [
102          "i915.modeset=1"
103          "i915.fastboot=1"
104          "i915.enable_guc=2"
105          "i915.enable_psr=1"
106          "i915.enable_fbc=1"
107          "i915.enable_dc=2"
108        ];
109        loader = {
110          systemd-boot = {
111            enable = true;
112            configurationLimit = 10;
113          };
114          efi = {
115            canTouchEfiVariables = true;
116          };
117        };
118      };
119
120      i18n.defaultLocale = "en_GB.UTF-8";
121      console = {
```

nixcfg / hosts / t480 / **configuration.nix**   ↑ Top

| Code | Blame | 🗗 Raw ⧉ ⬇ ✏ ▾   ‹›

```nix
126      time.timeZone = "Europe/London";
127      location.provider = "geoclue2";
128
129      hardware = {
130        opengl.enable = true;
131        opengl.extraPackages = with pkgs; [
132          vulkan-loader
133          vulkan-validation-layers
134          vulkan-extension-layer
135          vulkan-tools
136        ];
137        trackpoint = {
138          enable = true;
139          sensitivity = 255;
140        };
141      };
```

```nix
142
143        environment.systemPackages = with pkgs; [
144          wget
145          chromium
146          vim
147          tmux
148          mpv
149          gnumake
150          htop
151          git
152          inputs.agenix.packages.x86_64-linux.default
153        ];
154
155        system.stateVersion = "23.11";
156      }
```