



Fix Symbola causing errors and failed installs

Tyler Kelley authored 4 months ago

 config.nix 11.13 KiB

```
1  {
2      config,
3      pkgs,
4      host,
5      username,
6      options,
7      ...
8  }:
9  let
10     inherit (import ../variables.nix) keyboardLayout;
11 in
12  {
13     imports = [
14         ./hardware.nix
15         ./users.nix
16         ../../modules/amd-drivers.nix
17         ../../modules/nvidia-drivers.nix
18         ../../modules/nvidia-prime-drivers.nix
19         ../../modules/intel-drivers.nix
20         ../../modules/vm-guest-services.nix
21         ../../modules/local-hardware-clock.nix
22     ];
23
24     boot = {
25         # Kernel
26         kernelPackages = pkgs.linuxPackages_zen;
27         # This is for OBS Virtual Cam Support
28         kernelModules = [ "v4l2loopback" ];
29         extraModulePackages = [ config.boot.kernelPackages.v4l2loopback ];
30         # Needed For Some Steam Games
31         kernel.sysctl = {
32             "vm.max_map_count" = 2147483642;
33         };
34         # Bootloader.
35         loader.systemd-boot.enable = true;
36         loader.efi.canTouchEfiVariables = true;
37         # Make /tmp a tmpfs
38         tmp = {
39             useTmpfs = false;
40             tmpfsSize = "30%";
41         };
42         # Appimage Support
43         binfmt.registrations.appimage = {
44             wrapInterpreterInShell = false;
45             interpreter = "${pkgs.appimage-run}/bin/appimage-run";
46             recognitionType = "magic";
47             offset = 0;
48             mask = '\xff\xff\xff\xff\x00\x00\x00\x00\xff\xff\xff';
49             magicOrExtension = '\x7fELF....AI\x02';
50         };
51         plymouth.enable = true;
52     };
53
54     # Styling Options
55     stylix = {
56         enable = true;
57         image = ../../config/wallpapers/beautifulmountainscape.jpg;
58         # base16Scheme = {
59             # base00 = "232136";
60             # base01 = "2a273f";
61             # base02 = "393552";
62             # base03 = "6e6a86";
63             # base04 = "908caa";
64             # base05 = "e0def4";
65             # base06 = "e0def4";
66             # base07 = "56526e";
67             # base08 = "eb6f92";
68             # base09 = "f6c177";
69             # base0A = "ea9a97";
70             # base0B = "3e8fb0";
```

```
71 # base0C = "9ccfd8";
72 # base0D = "c4a7e7";
73 # base0E = "f6c177";
74 # base0F = "56526e";
75 # };
76 polarity = "dark";
77 opacity.terminal = 0.8;
78 cursor.package = pkgs.bibata-cursors;
79 cursor.name = "Bibata-Modern-Ice";
80 cursor.size = 24;
81 fonts = {
82     monospace = {
83         package = pkgs.nerdfonts.override { fonts = [ "JetBrainsMono" ]; };
84         name = "JetBrainsMono Nerd Font Mono";
85     };
86     sansSerif = {
87         package = pkgs.montserrat;
88         name = "Montserrat";
89     };
90     serif = {
91         package = pkgs.montserrat;
92         name = "Montserrat";
93     };
94     sizes = {
95         applications = 12;
96         terminal = 15;
97         desktop = 11;
98         popups = 12;
99     };
100 };
101 };
102
103 # Extra Module Options
104 drivers.amdgpu.enable = true;
105 drivers.nvidia.enable = false;
106 drivers.nvidia-prime = {
107     enable = false;
108     intelBusID = "";
109     nvidiaBusID = "";
110 };
111 drivers.intel.enable = false;
112 vm.guest-services.enable = false;
113 local.hardware-clock.enable = false;
114
115 # Enable networking
116 networking.networkmanager.enable = true;
117 networking.hostName = host;
118 networking.timeServers = options.networking.timeServers.default ++ [ "pool.ntp.org" ];
119
120 # Set your time zone.
121 time.timeZone = "America/Chicago";
122
123 # Select internationalisation properties.
124 i18n.defaultLocale = "en_US.UTF-8";
125
126 i18n.extraLocaleSettings = {
127     LC_ADDRESS = "en_US.UTF-8";
128     LC_IDENTIFICATION = "en_US.UTF-8";
129     LC_MEASUREMENT = "en_US.UTF-8";
130     LC_MONETARY = "en_US.UTF-8";
131     LC_NAME = "en_US.UTF-8";
132     LC_NUMERIC = "en_US.UTF-8";
133     LC_PAPER = "en_US.UTF-8";
134     LC_TELEPHONE = "en_US.UTF-8";
135     LC_TIME = "en_US.UTF-8";
136 };
137
138 programs = {
139     firefox.enable = false;
140     starship = {
```

```
141 enable = true;
142 settings = {
143   add_newline = false;
144   buf = {
145     symbol = " ";
146   };
147   c = {
148     symbol = " ";
149   };
150   directory = {
151     read_only = " ";
152   };
153   docker_context = {
154     symbol = " ";
155   };
156   fossil_branch = {
157     symbol = " ";
158   };
159   git_branch = {
160     symbol = " ";
161   };
162   golang = {
163     symbol = " ";
164   };
165   hg_branch = {
166     symbol = " ";
167   };
168   hostname = {
169     ssh_symbol = " ";
170   };
171   lua = {
172     symbol = " ";
173   };
174   memory_usage = {
175     symbol = " ";
176   };
177   meson = {
178     symbol = " ";
179   };
180   nim = {
181     symbol = " ";
182   };
183   nix_shell = {
184     symbol = " ";
185   };
186   nodejs = {
187     symbol = " ";
188   };
189   ocaml = {
190     symbol = " ";
191   };
192   package = {
193     symbol = " ";
194   };
195   python = {
196     symbol = " ";
197   };
198   rust = {
199     symbol = " ";
200   };
201   swift = {
202     symbol = " ";
203   };
204   zig = {
205     symbol = " ";
206   };
207 };
208 };
209 dconf.enable = true;
210 seahorse.enable = true;
```

```
211 fuse.userAllowOther = true;
212 mtr.enable = true;
213 gnupg.agent = {
214     enable = true;
215     enableSSHSupport = true;
216 };
217 virt-manager.enable = true;
218 steam = {
219     enable = true;
220     gamescopeSession.enable = true;
221     remotePlay.openFirewall = true;
222     dedicatedServer.openFirewall = true;
223 };
224 thunar = {
225     enable = true;
226     plugins = with pkgs.xfce; [
227         thunar-archive-plugin
228         thunar-volman
229     ];
230 };
231 };
232
233 nixpkgs.config.allowUnfree = true;
234
235 users = {
236     mutableUsers = true;
237 };
238
239 environment.systemPackages = with pkgs; [
240     vim
241     wget
242     killall
243     eza
244     git
245     cmatrix
246     lolcat
247     htop
248     brave
249     libvirt
250     lxqt.lxqt-policykit
251     lm_sensors
252     unzip
253     unrar
254     libnotify
255     v4l-utils
256     ydotool
257     duf
258     ncdu
259     wl-clipboard
260     pciutils
261     ffmpeg
262     socat
263     cowsay
264     ripgrep
265     lshw
266     bat
267     pkg-config
268     meson
269     hyprpicker
270     ninja
271     brightnessctl
272     virt-viewer
273     swappy
274     appimage-run
275     networkmanagerapplet
276     yad
277     inxi
278     playerctl
279     nh
280     nixfmt-rfc-style
```

```
281 discord
282 libvirt
283 swww
284 grim
285 slurp
286 file-roller
287 swaynotificationcenter
288 imv
289 mpv
290 gimp
291 pavucontrol
292 tree
293 spotify
294 neovide
295 greetd.tuigreet
296 ];
297
298 fonts = {
299   packages = with pkgs; [
300     noto-fonts-emoji
301     noto-fonts-cjk
302     font-awesome
303     # Commenting Symbola out to fix install this will need to be fixed or an alternative found.
304     # symbola
305     material-icons
306   ];
307 };
308
309 environment.variables = {
310   ZANEYOS_VERSION = "2.2";
311   ZANEYOS = "true";
312 };
313
314 # Extra Portal Configuration
315 xdg.portal = {
316   enable = true;
317   wlr.enable = true;
318   extraPortals = [
319     pkgs.xdg-desktop-portal-gtk
320     pkgs.xdg-desktop-portal
321   ];
322   configPackages = [
323     pkgs.xdg-desktop-portal-gtk
324     pkgs.xdg-desktop-portal-hyprland
325     pkgs.xdg-desktop-portal
326   ];
327 };
328
329 # Services to start
330 services = {
331   xserver = {
332     enable = false;
333     xkb = {
334       layout = "${keyboardLayout}";
335       variant = "";
336     };
337   };
338   greetd = {
339     enable = true;
340     vt = 3;
341     settings = {
342       default_session = {
343         # Wayland Desktop Manager is installed only for user ryan via home-manager!
344         user = username;
345         # .wayland-session is a script generated by home-manager, which links to the current wayland compositor(sway) and
346         # with such a vendor-no-locking script, we can switch to another wayland compositor without modifying greetd
347         # command = "$HOME/.wayland-session"; # start a wayland session directly without a login manager
348         command = "${pkgs.greetd.tuigreet}/bin/tuigreet --time --cmd Hyprland"; # start Hyprland with a TUI login manager
349       };
350     };
351   };
352 }
```

```
351 };
352 smartd = {
353     enable = false;
354     autodetect = true;
355 };
356 libinput.enable = true;
357 fstrim.enable = true;
358 gvfs.enable = true;
359 openssh.enable = true;
360 flatpak.enable = false;
361 printing = {
362     enable = true;
363     drivers = [
364         # pkgs.hplipWithPlugin
365     ];
366 };
367 gnome.gnome-keyring.enable = true;
368 avahi = {
369     enable = true;
370     nssmdns4 = true;
371     openFirewall = true;
372 };
373 ipp-usb.enable = true;
374 syncthing = {
375     enable = false;
376     user = "${username}";
377     dataDir = "/home/${username}";
378     configDir = "/home/${username}/.config/syncthing";
379 };
380 pipewire = {
381     enable = true;
382     alsa.enable = true;
383     alsa.support32Bit = true;
384     pulse.enable = true;
385 };
386 rpcbind.enable = false;
387 nfs.server.enable = false;
388 };
389 systemd.services.flatpak-repo = {
390     path = [ pkgs.flatpak ];
391     script = ''
392         flatpak remote-add --if-not-exists flathub https://flathub.org/repo/flathub.flatpakrepo
393     '';
394 };
395 hardware.sane = {
396     enable = true;
397     extraBackends = [ pkgs.sane-airscan ];
398     disabledDefaultBackends = [ "escl" ];
399 };
400
401 # Extra Logitech Support
402 hardware.logitech.wireless.enable = false;
403 hardware.logitech.wireless.enableGraphical = false;
404
405 # Bluetooth Support
406 hardware.bluetooth.enable = true;
407 hardware.bluetooth.powerOnBoot = true;
408 services.bluedevil.enable = true;
409
410 # Enable sound with pipewire.
411 hardware.pulseaudio.enable = false;
412
413 # Security / Polkit
414 security.rtkit.enable = true;
415 security.polkit.enable = true;
416 security.polkit.extraConfig = ''
417     polkit.addRule(function(action, subject) {
418         if (
419             subject.isInGroup("users")
420             && (
```

```
421         action.id == "org.freedesktop.login1.reboot" ||
422         action.id == "org.freedesktop.login1.reboot-multiple-sessions" ||
423         action.id == "org.freedesktop.login1.power-off" ||
424         action.id == "org.freedesktop.login1.power-off-multiple-sessions"
425     )
426 )
427 {
428     return polkit.Result.YES;
429 }
430 })
431 '';
432 security.pam.services.swaylock = {
433     text = ''
434         auth include login
435     '';
436 };
437
438 # Optimization settings and garbage collection automation
439 nix = {
440     settings = {
441         auto-optimise-store = true;
442         experimental-features = [
443             "nix-command"
444             "flakes"
445         ];
446         substituters = [ "https://hyprland.cachix.org" ];
447         trusted-public-keys = [ "hyprland.cachix.org-1:a7pgxzMz7+chwVL3/pzj6jIBMioiJM7ypFP8PwtkuGc=" ];
448     };
449     gc = {
450         automatic = true;
451         dates = "weekly";
452         options = "--delete-older-than 7d";
453     };
454 };
455
456 # Virtualization / Containers
457 virtualisation.libvirtd.enable = true;
458 virtualisation.podman = {
459     enable = true;
460     dockerCompat = true;
461     defaultNetwork.settings.dns_enabled = true;
462 };
463
464 # OpenGL
465 hardware.graphics = {
466     enable = true;
467 };
468
469 console.keyMap = "${keyboardLayout}";
470
471 # Open ports in the firewall.
472 # networking.firewall.allowedTCPPorts = [ ... ];
473 # networking.firewall.allowedUDPPorts = [ ... ];
474 # Or disable the firewall altogether.
475 # networking.firewall.enable = false;
476
477 # This value determines the NixOS release from which the default
478 # settings for stateful data, like file locations and database versions
479 # on your system were taken. It's perfectly fine and recommended to leave
480 # this value at the release version of the first install of this system.
481 # Before changing this value read the documentation for this option
482 # (e.g. man configuration.nix or on https://nixos.org/nixos/options.html).
483 system.stateVersion = "23.11"; # Did you read the comment?
484 }
485
```