

domenkozar / configuration.nix



Created 12 years ago • Report abuse

[Code](#) [Revisions 1](#) [Stars 16](#) [Forks 1](#)

configuration.nix

```
1 { config, pkgs, ... }:
2
3 {
4   require = [
5     <nixos/modules/programs/virtualbox.nix>
6     ./hardware.nix
7     ./monitors.nix
8     # ./mailpile.nix
9   ];
10
11   hardware.pulseaudio.enable = true;
12
13   networking.hostName = "borovnicke.kdoma.si";
14   networking.firewall = {
15     enable = false;
16     allowedTCPPorts = [
17       #80 # nginx
18       9102 # bacula-fd
19     ];
20   };
21   networking.networkmanager.enable = true;
22
23   time.timeZone = "Europe/Ljubljana";
24
25   #nix.trustedBinaryCaches = [ "http://hydra.nixos.org/" ];
26   # build packages in chroot
27   nix.useChroot = true;
28
29   environment.etc."synergy-server.conf" = { text = ''
30     section: screens
31       borovnicke:
32         ananas:
33     end
34
35     section: links
36       borovnicke:
37         right = ananas
38       ananas:
39         left = borovnicke
40     end
41   ''; };
42
```

```
43 #export NIX_PATH=/etc/nixos/borovnicke
44 environment.variables.EDITOR = pkgs.lib.mkOverride 0 "vim";
45 environment.variables.NIXOS_CONFIG = "/etc/nixos/borovnicke/configuration.nix";
46 programs.bash.enableCompletion = true;
47
48 environment.interactiveShellInit = ''
49     export PATH="$PATH:$HOME/bin"
50
51     parse_git_branch() {
52         git branch 2> /dev/null | sed -e '/^[^*]/d' -e 's/* \(.*/on \1/'
53     }
54     echo_tmux_title() {
55         printf "\033k`whoami`@\`pwd`: \033\`\"
56     }
57     PS1="$PS1[\e[0;33;49m]\$(parse_git_branch)[\e[0;0m]\$(echo_tmux_title)\n$ "
58
59     # append history instead of overwrite
60     shopt -s histappend
61
62     # big history, record everything
63     export HISTCONTROL=ignoredups:erasedups # no duplicate entries
64     export HISTSIZE=300000
65     export HISTFILESIZE=200000
66     export PROMPT_COMMAND="history -a; history -c; history -r; $PROMPT_COMMAND"
67
68     # check the window size after each command and, if necessary,
69     # update the values of LINES and COLUMNS.
70     shopt -s checkwinsize
71
72     #tmux attach || tmux new
73 '';
74
75 environment.systemPackages = with pkgs; [
76     # generally useful tools
77     links2
78     wgetpaste
79     pwgen
80     curl
81     vimHugeX
82     gitFull
83     gitAndTools.tig
84     subversion
85     mercurial
86     autojump # doesnt keep history
87     anki
88     sqlite
89     sshfsFuse
90     sshuttle
91     fuse
92     ncmpcpp
93     mplayer2
94     jwhois
```

```
95     dos2unix
96     gnumake
97     libevent
98     vifm
99     patchelf
100    bittorrentSync
101    reaverwps
102    unetbootin
103    ike
104    gcc
105    pythonPackages.tarman
106    sysstat
107    nixops
108    speedtest_cli
109    mosh
110    ngrok
111
112    # sys tools
113    rmlint
114    extundelete
115    hdparm
116    aircrackng
117    tmux
118    gnupg
119    wget
120    nmap
121    ncdu
122    telnet
123    unzip
124    unrar
125    psmisc
126
127    # X apps
128    dmenu
129    i3status
130    i3lock
131    i3minator
132    # TODO: make it a floating window so it's not annoying
133    p7zip
134    gmpc
135    pavucontrol
136    xarchive # TODO: add to xfce4 package
137    pa_applet
138    dunst # notifications
139    xfce.thunar_archive_plugin
140    arandr
141
142    wireshark
143    transmission_gtk
144    gnucash
145    gimp_2_8
146    calibre
```

```
147     skype
148     libreoffice
149     vlc
150     evince
151     wine
152
153     networkmanagerapplet
154
155     # TODO: changes hash all the time, didnt find an url that includes version name
156     #(pkgs.lib.overrideDerivation pkgs.teamviewer8 (attrs: {
157         #src = fetchurl { url = "http://download.teamviewer.com/download/teamviewer_linux
158             #sha256 = "1i1hi6s4nggcndfpp6rzzr98cidh72gwx74kb70j84ib1l6lkb9zk"
159     #}))
160
161     # browsers
162     firefoxWrapper
163     chromiumWrapper
164
165     # games
166     spring
167     springLobby
168     teeworlds
169     #xonotic
170
171     # postgresql
172     pgadmin
173
174     # development
175     nodePackages.jshint
176     #nodePackages.ungit
177     phantomjs
178
179     # iphone
180     libimobiledevice
181     usbmuxd
182
183     # python stuff
184     pypy
185     cython
186     python27Full
187     python26Full
188     python33
189     python32
190     pythonPackages.virtualenv
191     pythonPackages.flake8
192     pythonPackages.pillow
193     pythonPackages.mrbob
194     pythonPackages.bpython
195
196     (pkgs.haskellPackages.ghcWithPackages (self : [
197         self.json
198         self.curl
```

```
199     ]))
200 ];
201
202 services = {
203     nixosManual.showManual = true;
204     locate.enable = true;
205     ntp.enable = true;
206     printing.enable = true;
207     postgresql.enable = true;
208     postgresql.package = pkgs.postgresql92;
209
210     synergy.server = {
211         enable = true;
212         autoStart = false;
213         screenName = "borovnicke";
214         address = "0.0.0.0";
215     };
216
217     bacula-fd = {
218         enable = true;
219         name = "borovnicke-fd";
220         director.ananas-dir.password = "foo";
221     };
222
223     #rabbitmq = {
224         #enable = true;
225     };
226
227     redis = {
228         enable = true;
229     };
230
231     #dd-agent = {
232         # enable = true;
233         # api_key = "";
234     };
235     #collectd.enable = true;
236
237     prey = {
238         enable = true;
239         apiKey = "aaa";
240         deviceKey = "aaa";
241     };
242
243     redshift = {
244         enable = true;
245         latitude = "46.055556";
246         longitude = "14.508333";
247         temperature.day = 5700;
248         temperature.night = 4600;
249     };
250
```

```
251     xserver = {
252         enable = true;
253         synaptics = {
254             enable = true;
255             twoFingerScroll = true;
256             additionalOptions = ''
257                 Option "TapButton2" "3"
258             '';
259         };
260         xkbModel = "thinkpad60";
261         startGnuPGAgent = true;
262         startOpenSSHAgent = false;
263         windowManager.default = "i3";
264         windowManager.i3.enable = true;
265         desktopManager.default = "xfce";
266         desktopManager.xfce.enable = true;
267         #desktopManager.default = "gnome3";
268         #desktopManager.gnome3.enable = true;
269         displayManager.auto.user = "ielectric";
270         displayManager.auto.enable = true;
271         vaapiDrivers = [ pkgs.vaapiIntel ];
272     };
273     mesa.videoDrivers = [ "intel" ];
274 };
275
276 nixpkgs.config = {
277
278     firefox = {
279         jre = false;
280         enableGoogleTalkPlugin = true;
281         enableAdobeFlash = true;
282     };
283
284     chromium = {
285         jre = false;
286         enableGoogleTalkPlugin = true;
287         enableAdobeFlash = true;
288     };
289
290 };
291
292 #users.mutableUsers = false;
293 users.extraUsers.ielectric = {
294     name = "ielectric";
295     group = "ielectric";
296     uid = 499;
297     #uid = 1001;
298     createHome = true;
299     extraGroups = [ "wheel" "vboxusers" "networkmanager" ];
300     #passwordFile = "/etc/ielectricPassword";
301     home = "/home/ielectric";
302     shell = "/run/current-system/sw/bin/bash";
```

```

303     };
304
305     users.extraGroups.ielectric = {
306         gid = 498;
307     };

```

gistfile1.txt

```

1  { config, pkgs, ... }:
2
3  # auto configure monitors when you connect them
4  # xfce 4.12 will fix this, once released.
5
6  # TODO: save configuration state.
7  # https://github.com/wertarbyte/autorandr/blob/master/autorandr
8
9  let
10     autoconnect = pkgs.writeScript "automonitors.sh" ''
11         #!/bin/sh
12         export DISPLAY=:0.0
13         export XAUTHORITY=/home/ielectric/.Xauthority
14         XRANDR=${pkgs.xlibs.xrandr}/bin/xrandr
15         exec > /tmp/monitor 2>&1
16         $XRANDR --auto
17     '';
18 in {
19     powerManagement.resumeCommands = ''
20         ${autoconnect}
21     '';
22     services.udev.extraRules = ''
23         ACTION=="change", SUBSYSTEM=="drm", ENV{HOTPLUG}=="1", RUN+="${autoconnect}"
24     '';
25 }

```

hardware.nix

```

1  { config, pkgs, ... }:
2
3  {
4      boot.loader.grub.device = "/dev/sda";
5      boot.loader.grub.version = 2;
6      boot.initrd.kernelModules = [ "dm-crypt" "ehci_hcd" "ahci" "ums_realtek" "usb_storage" ];
7      boot.initrd.luks.devices = [{ name = "root"; device = "/dev/sda2"; allowDiscards = true };
8      boot.kernelModules = [ "kvm-intel" "tp_smapi" ];
9      boot.extraModulePackages = [ config.boot.kernelPackages.tp_smapi ];
10     boot.extraModprobeConfig = ''
11         #options iwlwifi 11n_disable=1 power_save=0
12     '';
13     # set kernel version
14     boot.kernelPackages = pkgs.linuxPackages_3_11;
15     boot.blacklistedKernelModules = [ "pcspkr" ];
16     #boot.extraKernelParams = [ "pcie_aspm=off" ];

```

```
17
18 hardware.enableAllFirmware = true;
19
20 #services.thinkfan.enable = true;
21
22 nix.maxJobs = 4;
23
24 fileSystems = [
25   { mountPoint = "/"; label = "root"; options = "noatime"; }
26   { mountPoint = "/boot"; label = "boot"; options = "noatime"; }
27   { mountPoint = "/tmp"; device = "tmpfs"; fsType = "tmpfs"; options = "nosuid,nodev,
28 ];
```