

Instalacja Gentoo 2023 AMD64 na laptopie HP-i7

Położenie: (nie dotyczy)

© 3bird Projects 2023, <http://edukacja.3bird.pl>

Uwaga: pomimo że laptop ma procesor Intel, to należy zainstalować wersję *Gentoo* oznaczoną jako AMD64. Na dysku instalacyjnym LiveCD powinno znaleźć się także (trzeba osobno ściągnąć): **stage3** oraz najnowszy **portage** (ściągnąć go z jakiegoś *mirror*a z folderu *snapshots*). Można także dołączyć **.config** jądra oraz inne „configi” (folder */etc*). Ewentualnie można to wszystko załączyć na pendrive.

W przypadku systemów UEFI, należy w tym momencie zajrzeć do osobnego dokumentu “UEFI” i aktywować odpowiednie opcje w jądrze. Instalacja Gentoo powinna być uruchomiona w stylu UEFI.

Parametry techniczne hp-i7:

Laptop: HP EliteBook 850 G8.

BIOS: HP T76 v. 01.08.11 (03.07.2022).

Motherboard: HP 8846, PCI-Express 3.0 (8.0 GT/s).

Procesor: Intel Core i7-1165G7 (11 generacja, 10nm), 4 rdzenie (8 wątków), **2.80GHz** (do 4,7GHz w trybie turbo), cache 12MB, **Code name:** Tiger Lake-U, Socket 1449 FCBGA. **Instrukcje:** aes, avx, avx2, avx512f, avx512dq, avx512cd, avx512bw, avx512vl, avx512vbmi, f16c, fma3, mmx, mmxext, pclmul, popcnt, rdrand, sha, sse, sse2, sse3, sse4_1, sse4_2, ssse3. **Moduły:** **i915**, **i965**, **proc_thermal**, **intel_pch_thermal**, **intel_lpss**, **va_gl** (VDPAU).

SSD: 2TB Dysk Samsung 970 EVO Plus 3400Mb/s, obudowa M.2 (2280), kontroler NVM Express Intel RST VMD 9A0B, bus: 0, target: 0, lun: 0 (narzędzie do zarządzania: „*nvme-cli*”, „*lsblk -o MODEL*”).

Grafika: LCD 15,6”, matryca IPS (*In-Plane-Switching*¹) o rozdzielczości 3840x2160px (4K), Intel Iris Xe Graphics (wspiera: DirectX, OpenGL, OpenCL), jasność 400 nitów, HDMI 2.0. Moduł **i915**.

RAM: 32GB (2x16GB) DDR4-3200 1600MHz, SK Hynix, HMAA2GS6AJR8N-XN.

Klawiatura: PC/AT PS/2 (101/102 klawisze).

Touchpad: Synaptics HID PointStyk. Obsługiwany przez DesignWare.

Biometria: Synaptics FS7605 Touch Fingerprint Sensor with PurePrint, Windows Hello Face, telemetria Intel Tigerlake. Sterownik w Linux: *hid-i2c*, *i2c_hid*, *hid_multitouch*, *intel_lpss_pci*, *hid-multitouchpad*, *synaptics*.

Dźwięk: głośniki Realtek Audio, mikrofon technologia Intel Smart Sound.

Kamera: HP HD Camera 0.92Mpx, Producent: LuxVisions, sterowniki: SunplusIT (moduł: *uvcvideo*).

Ethernet: przejściówka USB-RJ45 (moduły **r8152**, *cdc_ether*), Realtek RTL8153 Gigabit Ethernet Adapter (wynik polecenia „*lsusb*”).

Wi-Fi: Intel Wi-Fi 6 AX201 160MHz, standard 802.11ax, wirtualna karta NDIS (moduł **iwlmwifi**).

USB: xHCI (moduł: *xhci_hcd*), USB 3.0/3.1 extensible Host, ACPI UCM-UCSI, Thunderbolt 4.0 HSA Component.

Bluetooth: Intel Wireless Bluetooth, moduły Microsoft, RFCOMM Protocol TDI.

Kontrolery: PCI Express (moduł *pcieport*), HID V2, HID PCI Minidriver for ISS, urządzenie I2C HID, Intel Dynamic Tuning, Intel Host Bridge/DRAM Registers, Intel Integrated Sensor Solution, Intel LPC Controller, Intel Serial IO GPIO / I2O, Intel SMBus / UMBus (driver: *i801_smbus*, kernel: **i2c_i801**), Intel SPI (flash), ISS Dynamics Bus Enumerator, TPM 2.0. **Info:** Można użyć **lspci**, aby zobaczyć wykryte urządzenia lub **lspci -k**, aby zobaczyć także moduły obsługujące urządzenia. Bardzo dobrym poleceniem jest także **lsusb** oraz **hwinfo**, **lshw** (lub **/usr/sbin/gtk-lshw** w trybie graficznym).

System: przydzielone Windows 11 Pro 22H2, numer seryjny 5CG22548Q8, aktywowany za pomocą licencji cyfrowej (klucz prezentowany za pomocą darmowego programu *ShowKeyPlus* ze sklepu Microsoft: 8P3GG-6NFFG-FXF67-XV8X9-66P2T).

Instalacja

Włączenie UEFI: **F10** (nacisnąć po pojawieniu się logo HP Wolf Security).

Menu bootowania: **F9** (nacisnąć po pojawieniu się logo HP Wolf Security). Domyślne bootowanie z

¹ Matryca IPS jest podświetlana, a więc nie przepalają się w niej piksele jak w aktywnych matrycach TFT (tranzystory i kondensatory). Ma także szybszy czas reakcji i oferuje wyraźny obraz przy większym koncie widzenia.

DVD lub pendrive'a jest wyłączone. Aby było to możliwe (w trybie *UEFI-Only*), należy wyłączyć opcję „*Secure Boot Control*” (trzeba będzie wpisać kod potwierdzenia). Jeśli to nie pomoże:

- albo włączyć opcję „*Launch CSM*” (*Compatibility Support Module*), co umożliwi bootowanie starych systemów z MBR na laptopach z UEFI (niezalecane!);
- albo stworzyć pendrive'a bootowalnego z *LiveCD (Hybrid ISO)*; nie należy używać „*Minimal Installation CD*”, gdyż ta wersja instalatora przeznaczona jest na systemy z BIOS/MBR (możliwe jest także użycie dystrybucji „*SystemRescueCd*” opartej na Gentoo i przystosowanej do UEFI). Pendrive bootowalny z UEFI-Only można stworzyć w windowsowym programie *Rufus* (jeśli nie wystąpi w nim opcja UEFI-Only, to znaczy że obraz *.iso jest przeznaczony na BIOS/MBR).

boot: gentoo

Keymap selection: pl

Zakładanie partycji na dysku

Uwaga: Zmiana nazwy i typu partycji nie usuwa danych, jeśli tylko rozmiar partycji pozostaje bez zmian. Należy zwrócić uwagę, iż zalecane jest umieszczenie partycji SWAP oraz /var (także /srv) na dysku HDD (ze względu na dużą ilość zapisów, co nie jest korzystne dla dysku SSD). Przy dużej ilości RAM, system i tak nie będzie często korzystał ze SWAP.

livecd root # **gdisk /dev/nvme0n1** (operacje będą wykonywane na dysku NVM Express)

Command (? for help): **p** (pokaż partycje)

Command (? for help): **d** (usuwanie starych partycji)

Command (? for help): **n** (tworzenie nowych partycji, nadajemy im nazwy i typy)

Command (? for help): **p**

Wynik ustawień:

Number	Start (sector)	End (sector)	Size	Code	Name
1	2048	411647	200.0 MiB	EF00	EFI system
2	411640	821247	200.0 MiB	8300	Linux filesystem (/boot)
3	821248	135038975	64.0 GiB	8200	Linux swap
4	135038976	1183614975	500.0 GiB	8304	Linux x86-64 root (/)
5	1183614976	3907028991	1.3 TiB	8302	Linux /home

Uwaga: Według oficjalnych zaleceń, partycja /var (jeśli już ją tworzymy) powinna mieć rozmiar około 8GB. W praktyce jednak, powinno to być minimum 20GB (np. sam pakiet „rust” potrzebuje 9GB na kompilację).

Command (? for help): **w**

livecd root # **shutdown -r now**

Konfiguracja sieci

Nacisnąć F9 i ponownie wybrać bootowanie z pendrive'a / DVD:

boot: gentoo

Keymap selection: pl

Po podłączeniu przelotki USB-ETH:

livecd root # **ip addr**

1: **lo:** <LOOPBACK,UP,LOWER_UP> mtu 65536...

inet 127.0.0.1/8 brd 127.255.255.255 scope host lo...

2: **wlp0s20f3:** <NO-CARRIER, BROADCAST, MULTICAST,UP> mtu 1500...

link/ether b0:3c:dc:b4:d9:26

3: **enp0s20f0u1:** <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500...

link/ether 9c:eb:e8:af:82:d2...

inet **192.168.7.36** brd 192.168.7.255...

Jeśli ethernet nie został automatycznie wykryty, próbujemy zrobić to ręcznie (po instalacji i tak to zmienimy):

livecd root # **ifconfig enp0s20f0u1 192.168.7.53 broadcast 192.168.7.255 netmask 255.255.255.0 up**

livecd root # **ifconfig wlp0s20f3 192.168.7.52 broadcast 192.168.7.255 netmask 255.255.255.0 up**

livecd root # **route add default gw 192.168.7.1**

Wynik „route”:

Destination	Gateway	Genmask	Flags	Metric	Ref	Use	Iface
192.168.7.0	*	255.255.255.0	U	0	0	0	enp0s20f0u1
loopback	localhost	255.0.0.0	UG	0	0	0	lo
default	192.168.7.1	0.0.0.0	UG	0	0	0	enp0s20f0u1

Uwaga: Polecenie „ifconfig” pochodzi z pakietu „net-tools”.

```
lived root # nano -w /etc/resolv.conf
domain 3bird
nameserver 1.1.1.1
nameserver 1.0.0.1
lived root # date
lived root # date 082012332018 (mm dd ggmm yyyy)
```

Przygotowanie dysków

```
lived root # mkdir /mnt/usb_sda1
lived root # mkdosfs -F 32 -n EFI-BOOT /dev/nvme0n1p1 (z pakietu dosfstools z flagą „compat”)
lived root # mkswap /dev/nvme0n1p3
lived root # swapon /dev/nvme0n1p3
lived root # mkreiserfs /dev/nvme0n1p2 (/boot)
lived root # mkreiserfs /dev/nvme0n1p4 (/)
lived root # mkreiserfs /dev/nvme0n1p5 (/home)
lived root # mount -o noatime /dev/nvme0n1p4 /mnt/gentoo
lived root # mkdir /mnt/gentoo/boot (jeśli nie ma)
lived root # mount -o noatime /dev/nvme0n1p2 /mnt/gentoo/boot
Info: Kopiujemy stage3 i portage z DVD lub z USB (czyli: mount -t vfat /dev/sda1 /mnt/usb; jeśli nie ma na płycie lub na pendrivie, można ściągnąć z Internetu za pomocą wget):
lived root # cp /mnt/usb/stage*.tar.xz /root
lived root # wget http://ftp.vectranet.pl/gentoo/snapshots/portage-latest.tar.bz2
lived root # cd /mnt/gentoo
lived gentoo # tar xpf /root/stage*.tar.xz --numeric-owner
lived gentoo # tar -xvjf /root/portage*.bz2 -C /mnt/gentoo/usr
lived gentoo # umount /mnt/usb_sda1
```

Przygotowanie systemu

```
lived gentoo # mount -t proc proc /mnt/gentoo/proc
lived gentoo # mount --rbind /sys /mnt/gentoo/sys
lived gentoo # mount -o bind /dev /mnt/gentoo/dev
lived gentoo # cp /etc/resolv.conf /mnt/gentoo/etc/resolv.conf
lived gentoo # chroot /mnt/gentoo /bin/bash
lived / # mkdir /var/db/repos/gentoo (jeśli nie ma)
lived / # env-update (generowanie zmiennych systemowych, tworzenie nowego środowiska)
lived / # source /etc/profile (wprowadzanie zmiennych systemowych do pamięci)
lived / # cat /proc/cpuinfo
lived / # gcc -c -Q -march=native --help=target | grep march (sprawdzamy jaki parametr 'march' wykrywa gcc; w moim przypadku: tigerlake)
lived / # nano /etc/portage/make.conf (ustawienie USE flags)
lived / # eselect profile list (sprawdzamy nasz bierzący profil; czy jest na pewno tym, czego chcemy)
lived / # emerge cpuid2cpuflags
lived / # cpuid2cpuflags (sprawdzamy flagi obsługiwane przez nasz procesor i poprawiamy plik /etc/portage/make.conf)
lived / # ln -sf /usr/share/zoneinfo/Europe/Warsaw /etc/localtime
lived / # echo „Europe/Warsaw” > /etc/timezone
lived / # nano /etc/fstab
# <fs> <mountpoint> <type> <opts> <dump/pass>
/dev/nvme0n1p1 /boot/efi vfat noauto,noatime 1 2
```

```

/dev/nvme0n1p2 /boot reiserfs notail,noauto 1 2
/dev/nvme0n1p3 none swap sw 0 0
/dev/nvme0n1p4 / reiserfs notail,noatime 0 0
/dev/nvme0n1p5 /home reiserfs notail 0 0
/dev/cdrom /mnt/cdrom auto noauto,user 0 0
/dev/sda /mnt/backup_sda reiserfs noauto,notail,rw,user 0 0
/dev/sda1 /mnt/backup_sda1 reiserfs noauto,notail,rw,user 0 0
/dev/sdb /mnt/backup_sdb reiserfs noauto,notail,rw,user 0 0
/dev/sda /mnt/usb_sda vfat noauto,rw,user 0 0
/dev/sda1 /mnt/usb_sda1 vfat noauto,rw,user 0 0
/dev/sda2 /mnt/usb_sda2 vfat noauto,rw,user 0 0
/dev/sdb /mnt/usb_sdb vfat noauto,rw,user 0 0
/dev/sdb1 /mnt/usb_sdb1 vfat noauto,rw,user 0 0
# Do obsługi systemu exFAT należy zainstalować: fuse-exfat, exfat-utils. Czasami należy dodać usera do grupy fuse.
# Aktywacja opcji user_allow_other pozwala zwyklemu uzytkownikowi montować dysk z linii poleceń przy użyciu opcji al-
low_other.
# Opcje dla exfat: users,umask=002,uid=1000,gid=1001 (jest to grupa "fuse")
/dev/sda1 /mnt/usb_sda1_exfat exfat-fuse noauto,rw,noatime,allow_other,users,fsmask=137,dmask=027,
uid=1000,gid=100 0 0
# Do obsługi systemu NTFS należy zainstalować: sys-fs/fuse, ntfs3g.
/dev/sda1 /mnt/usb_sda1_ntfs ntfs noauto,rw,user,noatime,utf8,fsmask=137,dmask=027,uid=1000,gid=100 0 0
proc /proc proc defaults 0 0
shm /dev/shm tmpfs nodev,nosuid,noexec 0 0
sysfs /sys sysfs nodev,nosuid,noexec 0 0

```

livecd / # In -snf /proc/self/mounts /etc/mtab (plik /etc/mtab wychodzi z użycia)

livecd / # eselect locale list | more

livecd / # eselect locale set 616 (czyli pl_PL.utf8; jeśli nie ma wszystkich locali, wrócić do tego na końcu)

livecd / # ls /usr/portage/sys-kernel

livecd / # emerge sys-kernel/gentoo-sources (instalacja wybranego źródła)
kernel-6.1.6

Kompilacja kernela ręcznie

livecd / # eselect kernel set 1

livecd / # cd /usr/src/linux

livecd / # make menuconfig

Info: Dyski twarde typu SCSI tworzą urządzenia /dev/sda* (a nie /dev/hda*). Muszą być na stałe wbudowane w jądro (nie mogą być kompilowane jako moduły). Generalnie istnieją sterowniki wysokiego poziomu (SCSI Disk) i sterowniki niskiego poziomu (SATA chipset).

Info: Nie jest wymagany initrd, jeśli filesystem jest wkompileowany w kernel.

Info: Sterowniki karty sieciowej powinny być zaznaczone jako moduły!

Należy uaktywnić opcje:

- General Setup / Auditing support
- Bus options / [*] PCI Express ASPM control
- Bus options / DesignWare PCI Core Support / [*] Platform bus based DesignWare PCIe Controller (obsługa touchpada)
- Device Drivers / Character Devices / [*] Legacy (BSD) PTY support (tworzy terminale, tj. Master → /dev/ptyX oraz Slave → /dev/ttyX)
- Device Drivers / Generic Drivers Options / [*] Maintain a devtmpfs filesystem to mount at /dev
- Device Drivers / SCSI Device support / <*> SCSI disk support (skoro dysk twardy jest typu SCSI, to nie można tego kompilować jako moduły!; tylko CD-ROM może być jako moduł)
- Device Drivers / Input device support / Mice / <M> PS/2 mouse (potrzebne dla niektórych touchpadów, np. FocalTech)
- Device Drivers / Network device support / <*> Ethernet driver support / [*] Realtek devices / <M> Realtek 8169
- Device Drivers / Network device support / <M> USB Network Adapters / <M> Realtek RTL8152 (sterowniki dla adaptera USB-Ethernet firmy Asus)
- Device Drivers / Network device support / <M> USB Network Adapters / <M> Multi-purpose USB Networking Framework / <M> CDC Ethernet support
- Device Drivers / Network device support / <M> USB Network Adapters / <M> Multi-purpose USB Networking Framework / <M> Davicom DM96xx based USB 10/100 ethernet devices (sterowniki dla niebieskiego adaptera USB-Ethernet)
- Device Drivers / Network device support / <M> USB Network Adapters / <M> Multi-purpo-

se USB Networking Framework / <M> CoreChip-sz SR9700 based USB 1.1 10/100 ethernet devices (sterowniki dla [niebieskiego adaptera USB-Ethernet](#))

- Device Drivers / NVME Support / <*> NVM Express block device (chipset dla dysków SSD, możliwe że widoczny będzie jako /dev/nvme0n1p1)
- Device Drivers / I2C support / I2C Hardware Bus Support -->
 - <*> Synopsys Designware Platform (obsługa touchpada)
 - <*> Synopsys Designware PCI
- Enable the block layer / Partition Type / [*] PC BIOS (MSDOS partition table)
- File systems / Pseudo Filesystems / [*] /proc file system support
- File systems / Pseudo Filesystems / [*] Tmpfs
- Networking support / Networking options / <*> Unix domain sockets
- Processor Types and Features / [*] Symmetric multi-processing support
- Protokół I2C (potrzebny dla sensorów sprzętowych, m. in. dla v4l)

Moduły (wynik lsmod): cfg80211, cdc_ether, usbnet, x86_pkg_temp_thermal, btusb, btrtl, btbcm, bluetooth, rkill, thermal, i915, acpi_cpufreq, i2c_i801, wmi, aesni_intel, tg3, libphy, r8169, r8152, fuse, ahci, pcieport, i801_smbus, uvcvideo, intel_vsec.

Nie należy uaktywniać opcji: MSI_WMI, HP_WMI.

```
livecd / # make && make modules_install
```

```
livecd / # mkdir /boot/efi
```

```
livecd / # mount /boot/efi
```

```
livecd / # mkdir /boot/efi/EFI/gentoo
```

```
livecd / # cp System.map /boot
```

```
livecd / # cp .config /boot/config-6.1.6-2023-01-18
```

```
livecd / # cp arch/x86/boot/bzImage /boot/kernel-6.1.6-2023-01-18
```

(Uwaga: katalog x86_64 jest linkiem do katalogu x86 i nie należy go kopiować; jądro kopiujemy tutaj tylko dla celów archiwalnych, gdyż i tak zostanie użyte jądro skopiowane do folderu EFI)

```
livecd / # cp arch/x86/boot/bzImage /boot/efi/EFI/gentoo/gentoo.efi
```

```
livecd / # cp arch/x86/boot/bzImage /boot/efi/EFI/Boot/bootx64.efi
```

(kopiujemy także tutaj, jako domyślny system fallback)

Uwaga: Konfiguracja obecnego działającego jądra dostępna jest po wydaniu polecenia: `cat /proc/config | more`. Plik konfiguracyjny to `/usr/src/linux/.config`. Jeśli wystąpią problemy podczas tworzenia folderów w EFI lub podczas kopiowania tam plików („Nie można przydzielić pamięci”) - należy ponownie sformatować partycję EFI jako fat32.

Po skopiowaniu jądra, tworzymy wpis w menu UEFI:

```
livecd / # emerge efibootmgr
```

```
livecd / # efibootmgr -c -d /dev/nvme0n1 -p 1 -L "Gentoo" -I "\EFI\gentoo\gentoo.efi"
```

(przypisujemy tutaj partycję zawierającą UEFI; to jest małe „L”; więcej informacji w dokumencie UEFI)

Ustawienia końcowe

```
livecd / # emerge acpid metalog reiserfsprogs udev sys-apps/dbus app-misc/mc
```

Info: W przypadku wpadnięcia w błędne koło zależności, należy zainstalować jakiś pakiet bez flag, a potem już normalnie, z flagami.

```
livecd / # rc-update add metalog boot
```

```
livecd / # rc-update add udev sysinit
```

```
livecd / # rc-update add acpid default
```

```
livecd / # rc-update add dbus default
```

```
livecd / # rc-update add hostname default
```

```
livecd / # rc-update add netmount default
```

```
livecd / # rc-update add modules boot
```

Info: Sprawdzić czy na poziomie boot są pakiety → root, procfs, mtab, fsck, swap, hwclock

```
livecd / # passwd
```

```
livecd / # nano /etc/conf.d/hostname (wpisać krótką nazwę komputera: „hp-i7”)
```

```
livecd / # nano /etc/hosts (wpisać IP maszyn w sieci)
```

```
127.0.0.1          localhost.localdomain    localhost
::1               localhost.localdomain    localhost
192.168.7.2       hp-i7.3bird              hp-i7
192.168.7.3       hp-i7.3bird              hp-i7
192.168.7.6       hp-i7-vm-suse.3bird      hp-i7-vm-suse
```

```
192.168.7.9      epson.3bird      epson
192.168.7.10   panasonic.3bird  panasonic
192.168.7.11   3bird-extender.3bird  3bird-extender
```

Wpisy na potrzeby wirtualnych domen Apache:

```
# 192.168.7.2      serwis1.teb      serwis1
# 192.168.7.2      serwis2.teb      serwis2
# 192.168.7.6      serwis1-vm.3bird  serwis1-vm
# 192.168.7.6      serwis2-vm.3bird  serwis2-vm
```

Wpisy na potrzeby TEB:

```
# 192.168.17.3    hp-i7.3bird      hp-i7
```

livecd / # **cat /etc/resolv.conf**

```
domain 3bird
nameserver 1.1.1.1
nameserver 1.0.0.1
```

Uwaga: Jeśli siecią będzie zarządzać *netmount* (patrz poniżej), to i tak będzie on nadpisywać plik */etc/resolv.conf* przy każdym uruchomieniu sieci zależnie od interfejsu (każdy interfejs może mieć własne odmienne DNS-y). Plik ten jest także nadpisywany przez serwis *dhclient* (jeśli istnieje).

livecd / # **nano /etc/conf.d/netmount**

Sprawdzamy, czy istnieje zahaszowana wartość (nie aktywować):

```
# rc_need="net.enp0s20f0u1"
# rc_need="net.wlp0s20f3"
```

livecd / # **nano /etc/rc.conf**

Ustawić:

```
UNICODE="yes"
rc_hotplug="*" (aby po wpięciu wtyczki eth był uruchamiany interfejs)
extra_net_fs_list="cifs"
```

livecd / # **nano /etc/conf.d/hwclock** (w *kernerze* musi być opcja/moduł „*rtcN*”)

```
clock=UTC
systohc=YES
hctosys=NO
```

livecd / # **USE="distinct-l -ru-g" emerge terminus-font** (rodzina pięknych polskich czcionek konsolowych, niestety największy rozmiar to tylko 32px)

livecd / # **nano /etc/conf.d/consolefont** (CONSOLEFONT="ter-232b", czyli rozmiar 32, największa, pogrubiona, kodowana jako iso-8859-2; poniżej wersje czcionki „*terminus*”)

Nazwa	Mapowanie	Strony kodowe
ter-1*	iso01, iso15, cp1252	ISO8859-1, ISO8859-15, Windows-1252
ter-2*	iso02, cp1250	ISO8859-2, Windows-1250
ter-7*	iso07, cp1253	ISO8859-7, Windows-1253
ter-9*	iso09, cp1254	ISO8859-9, Windows-1254
ter-c*	cp1251, iso05	Windows-1251, ISO8859-5
ter-d*	iso13, cp1257	ISO8859-13, Windows-1257
ter-g*	iso16	ISO8859-16
ter-i*	cp437	IBM-437
ter-k*	koi8r	KOI8-R
ter-m*	mik	Bulgarian-MIK
ter-p*	pt154	Paratype-PT154
ter-u*	koi8u	KOI8-U
ter-v*	wszystkie powyżej i więcej	wszystkie powyżej i więcej (8 kolorów)

Nazwa	Style
ter-*n	normal
ter-*b	bold
ter-*f	framebuffer-bold
ter-x*	tylko dla środowiska graficznego

Uwaga: Pliki czcionek kopiowane są do dwóch miejsc:

- `/usr/share/fonts` – tu są czcionki dla środowiska X
- `/usr/share/consolefonts` – to są czcionki dla konsoli

Aby zobaczyć natychmiastowy efekt zmiany czcionki:

```
livecd / # /etc/init.d/consolefont restart
livecd / # rc-update add consolefont boot
livecd / # nano /etc/conf.d/keymaps (tylko to: KEYMAP="pl", windowkeys="YES")
livecd / # emerge noto dejavu ubuntu-font-family inconsolata xorg-fonts fonts-meta co-
refonts (inne czcionki)
livecd / # umount /mnt/usb
livecd / # umount /boot/efi
livecd / # umount /boot
livecd / # exit
livecd gentoo # exit
livecd # cd /
livecd / # umount /mnt/gentoo/proc
livecd / # umount -l /mnt/gentoo/dev{/shm,/pts,}
livecd / # umount -R /mnt/gentoo
livecd / # reboot
```

Po restarcie

```
# mkdir /mnt/cdrom
# mkdir /mnt/usb_sda1 (i inne)
# emerge -vp xorg-server plasma-meta libreoffice firefox konsole cronie display-mana-
ger sddm xf86-video-intel dhcp (tylko client)...
# useradd -g users -G
disk,wheel,audio,cdrom,cdrw,video,cron,scanner,usb,vboxusers,lp,lpadmin,backlight,kvm,ntfsu-
ser,polkitd,plugdev,users -m -d /home/robert -s /bin/bash -c 'Robert Surma' robert && passwd
robert (niektóre z tych grup jeszcze nie istnieją)
# nano /etc/conf.d/display-manager
CHECKVT=7
DISPLAYMANAGER="sddm"
# rc-update add display-manager default
# rc-update add elogind boot (bez tego nie będzie w SDDM przycisków „Restart” i „Zamknij”)
# env-update && source /etc/profile
# eselect (poustawiać wszystkie opcje)
Info: Za pomocą programu kMenuEdit można poustawiać składniki menu.
```

Naprawa partycji

Jeśli w jakiegoś powodu utraciliśmy dane na partycji (np. w przypadku powiększenia partycji lub dzielenia), część plików można odzyskać.

Wszystkie próby naprawcze lepiej robić na kopii partycji niż na oryginale. Dlatego kopiujemy obraz partycji na zewnętrzny dysk:

```
# dd if=/dev/sda3 conv=sync,noerror > /mnt/backup_sdc/kopia_partycji.img
```

Kojarzymy obraz z urządzenie `loop`:

```
# losetup /dev/loop0 /mnt/backup_sdc/kopia_partycji.img
```

Próbujemy odbudować super-block lub go po prostu sprawdzić:

```
# reiserfsck --rebuild-sb /dev/loop0
```

Przebudowujemy drzewo katalogów (może zająć ze 3 godziny):

```
# reiserfsck --rebuild-tree -S /dev/loop0
```

Sprawdzamy, czy przebudowa i naprawa się udała:

```
# reiserfsck --check /dev/loop0
```

Ewentualnie:

```
# reiserfsck --fix-fixable (naprawia nie naruszając drzewa systemu plików)
```

Montujemy naprawioną partycję:

```
# mount /dev/loop0 /mnt/tymczasowy
```

Dodatkowe informacje

Niektórzy zalecają w stosunku do dysków SSD włączenie w cronie dwa razy dziennie polecenia:

```
# Mins Hours Days Months Day of the week command  
15 1,13 * * * /sbin/fstrim -v /
```

Dotyczy to jednak starszych dysków SSD (nie znam dobrze szczegółów, więc na razie wstrzymuję się).

```
# rc-update del binfmt boot (serwis, który pozwala na uruchamianie innych plików binarnych niż standardowe; usuwam ten serwis, bo wydaje mi się podejrzany, a bez niego wszystko działa)
```

Ostatnia aktualizacja: 31 stycznia 2023.