

# System Linux – wprowadzenie

Jacek Fiok ([jfiok@jfiok.org](mailto:jfiok@jfiok.org); [www.jfiok.org](http://www.jfiok.org))

27.02.2003r.

Czas: 2 x 1h.

Plan:

1. Wprowadzenie do systemu linux. Założenia, budowa systemu.
  2. Tryb tekstowy, terminal, podstawowe komendy. praca zdalna (ssh).
  3. Nowoczesny tryb graficzny, powszechnie uznane i używane programy.  
-- na przykładzie dystrybucji Red Hat Linux 8.0.
- 

## 1. Słowo wstępne

Linux to system operacyjny. Tak jak Windows, MacOS, itd.  
Mówi się, że należy on rodziny systemów UNIX.

### 1.1. Historia (w telegraficznym skrócie)

- System UNIX wywodzi się z Bell Labs (AT&T), jego korzenie sięgają lat 60-ych (1969: UNICS).
- Historia jest dosyć długa. Kwestie prawno-licencyjne: zakaz sprzedawania systemu bezpośrednio. AT&T rozdawało więc licencje. Firmy dodawały ulepszenia i wypuszczały swoje “wersje” pod różnymi nazwami.
- Najważniejszy wniosek: aktualnie istnieją dwie główne “gałęzie” systemu: “System V” oraz “BSD”. Drobne różnice głównie w konfiguracji, wszystkie najważniejsze “funkcje” są wspólne.

Systemy UNIX były typowo systemami komercyjnymi dla bardzo drogich, dużych maszyn (w latach 70-80 nie było powszechności komputerów osobistych!).

Lata 80 (i początek 90): upowszechnienie się komputerów, TCP/IP i Internetu już mniej więcej takiego, jaki znamy dziś.

1991: Linus Torvalds, student z Finlandii, początkowo w zasadzie

hobbistycznie i dla własnych potrzeb - tworzy swój **nowy** system operacyjny, wzorując się na systemie Minix. System zostaje nazwany Linux. Jest rozdawany w Internecie za darmo. Wkrótce Linusowi zaczynają pomagać tysiące osób z całego świata.

### **Podsumowanie:**

- Linux należy do rodziny systemów UNIX.
- Sam Linux jako taki nie jest własnością żadnej firmy. Jest darmowy. Legalny. Nie trzeba żadnych licencji.

Z Linuksem wiąże się **ruch GNU**, na temat którego można by wygłosić oddzielny wykład. Najważniejsze dla Kowalskiego: Linux to nie tylko **freeware**, ale również: dostępność (pełen wgląd, możliwość modyfikacji) do kodu źródłowego – zaufanie, bezpieczeństwo (żadnych dowcipnych “bocznych furtek” do komputera, co **zdarzało się** w systemach komercyjnych).

## **1.2. Budowa, główne cechy i założenia systemu**

**System operacyjny** (luźna definicja) - program, który pośredniczy w komunikacji użytkownika ze sprzętem, wykonując przy okazji wiele zadań pomocniczych. “Superprogram”, pierwszy, nadzorujący całość.

**System operacyjny = jądro („kernel”) + narzędzia.**

Typowo, s.o. zajmuje się takimi rzeczami, jak:

- urządzenia fizyczne
- system plików
- uruchamianie programy
- zarządzanie zasobami
- użytkownicy, bezpieczeństwo
- komunikacja z użytkownikiem i światem (tryb tekstowy, okienka, internet).

Są **dwa główne tryby pracy systemu**: (tylko jeden!) tryb jądra („kernel-mode”) i tryb użytkownika („userspace mode”).

Dzięki **modularności** możemy np. nie łądować/wykonywać kodu obsługi urządzeń, których nie posiadamy w komputerze – **oszczędność zasobów**.

Umożliwia to również: **zmianę ustawień**, a nawet **napisanie od nowa**, skompilowanie i załadowanie nowej funkcji **bez restartu komputera** – **elastyczność**.

Jądro jest **wielozadaniowe i wieloużytkownikowe**.

Główną „jednostką” w systemie, zazwyczaj równoważną wykonywaniu jakiegoś programu (“zadania”), jest **proces** („3 użytkowników ma uruchomionych 10 procesów”).

**Wielozadaniowość z wywłaszczaniem** (preemptive multitasking) – możliwość „zabrania” procesowi dostępu do zasobów (procesora) na rzecz innego: procesu, obsługi zakończonej transmisji danych itd.

Jądro rozdziela zasoby między potrzebujące ich procesy według ściśle zdefiniowanych **mechanizmów szeregowania** (scheduling).

**Asynchroniczne wejście-wyjście** (asynchronous I/O) – operacje odczytu/zapisu na urządzeniach nie blokują pracy innych procesów (np. systemy Windows95/98 są na pograniczu zawieszenia, kiedy czytamy/piszemy na dyskietkę).

**Ochrona pamięci** - jeden proces nie może uszkodzić innego procesu, ani tym bardziej systemu.

**Demony (daemons).**

Wiemy już o trybie jądra i trybie użytkownika.

Inny podział: istnieją programy „zwykłe”, które **wykonują jakąś pracę i kończą swoje działanie** (rysując okienka, czy być może zapisując gdzieś jakiś wynik), oraz takie, które **działają w systemie przez cały czas, zwykle oferując pewne usługi**. Programy te nazywamy demonami.

Typowe demony to np. serwer WWW, czy FTP.

Istnieją również demony poziomu jądra (np. związane z pamięcią wirtualną).

### **System plików:**

- pliki
- katalogi
- pliki specjalne (tak w Linuksie “widoczne” są urządzenia)
- “dołączanie” innych systemów plików

Ścieżki dostępu są postaci typu /katalog/katalog2/plik.txt.

Nie ma “dysków” (C:, D:). Cały system plików “zaczyna się” od “/”, a fizyczne urządzenia są “dołączane” (mount) w dowolnych miejscach -- np. /windows (a typowo: /mnt/[literka dysku znana z windowsów]).

C:\windows\plik.txt  
/mnt/c/windows/plik.txt

(uwaga: Linux bez problemu czyta i pisze na dyski windowsowe; w drugą stronę jest trudniej – przynajmniej z gołymi windowsami się nie da)

Ścieżki mogą być bezwzględne (gdy zaczynają się od “/”), lub względne – zaczynając od katalogu, w którym się aktualnie znajdujemy.

**Jeszcze jedna uwaga:** rozpowszechnianiem (darmowego!) Linuksa zajmuje się wiele firm (ma marginesie - pytanie, z czego one żyją? np. ze sprzedaży za pieniądze wersji z bogatszą dokumentacją, pomocy technicznej, wdrożeń...). Takie rozprowadzane zestawy (czyli jądro i wybrany przez firmę zbiór programów) nazywamy **dystrybucjami**.

## **2. Praca z systemem: tryb tekstowy, terminal, podstawowe komendy.**

Kiedy dostajemy się do Linuksa “z zewnątrz”, odbywa się to zazwyczaj w trybie tekstowym – pracujemy z komendami tekstowymi (co przypomina nieco DOS). Mówimy, że **otwieramy terminal**.

Mówimy też, że jest to **praca zdalna**. Programy są fizycznie uruchamiane na komputerze z Linuksem. My, np. na Windowsach w okienku ssh, wprowadzamy komendy. Komendy są przesyłane przez sieć, wykonywane i ostatecznie widzimy efekty ich działania (wypisywany tekst) znowu w okienku terminala.

Oczywiście, można tak pracować również przy “prawdziwym” komputerze - “terminal lokalny”.

### **Logowanie**

- Aplikacje: telnet, ssh (telnet: niezaszyfrowane hasła).
- (telnet: jest w Windowsach; ssh – trzeba ściągnąć; polecany program: putty – [www.chiark.greenend.org.uk/~sgtatham/putty](http://www.chiark.greenend.org.uk/~sgtatham/putty))
- Użytkownik = {identyfikator, hasło}

### **Wylogowanie**

- exit.

### **Ogólna zasada wyglądu poleceń:**

znak zachęty\$ program [argumenty...]

**Opcje programów** najczęściej podaje się ze znakiem - “-opcja”.

**Wbudowana pomoc** do programów – typowo opcja “--help”, czasem “-help” lub “-h”.

**Podręcznik/pomoc systemowa** (“manual pages”): **man [polecenie]**.

Pomoc do programu obsługi pomocy – **man man**.

## Absolutnie podstawowe komendy:

**ls** – wyświetla zawartość katalogu (dos: DIR)

**ls -l** – zawartość ze szczegółami.

**cd** – zmienia “katalog bieżący” (prawie tak jak w dos).

Każdy użytkownik ma swój “katalog domowy” (prawie jak “moje dokumenty”). Np. /home/users/jfiok. Forma skrótowa: ~ lub ~użytkownik.

Samo “cd” przechodzi do ~. Wyświetlenie katalogu bieżącego: **pwd**.

Tworzenie i usuwanie katalogów: **mkdir**, **rmdir** (dos: md, rd).

Kopiowanie plików: **cp** [co] [gdzie]; przenoszenie: **mv** [co] [gdzie].

Usuwanie: **rm** [co].

Można używać **wzorców** (np. \*; nie trzeba, jak w dos-ie “\*.\*)”).

---

## Prawa dostępu do plików

```
-rw-rw-r-- 1 jfiok jfiok 5606 sty 31 01:05 notatki.txt
```

1. kolumna - typ pliku; dalej – 3x3 znaki:

- prawa do: (czytania, pisania, wykonywania)
- dla (właściciela, jego grupy, pozostałych użytkowników)

W systemie istnieje jeden “super-użytkownik”, administrator. Jego konto nazywa się root. “Znak zachęty” dla roota typowo kończy się na # zamiast \$. Tylko root ma prawo np. instalować nowe programy, czy zmieniać konfigurację urządzeń. Warta zauważenia pozytywna konsekwencja: “zwykły” użytkownik nie zniszczy (być może nawet przypadkiem!) całego systemu.

---

## Komendy - cd.

- wypisanie zawartości pliku: **cat** [plik]
- stworzenie nowego pliku: **touch** [plik]
- najprostszy edytor: **pico** (dla ludzi o mocniejszych nerwach: **vi**)
- zmiana hasła: **passwd**

- **finger** – pokazuje, kto jest zalogowany na danym komputerze
- **finger [użytkownik]** – pokazuje informacje o danym użytkowniku (nazwisko, ostatni czas zalogowania).

[przerwa]

### Inne polecenia, nieco bardziej “rozrywkowe”:

- informacje o systemie:
  - **uname, uname -a**: nazwa systemu (u nas: Linux, inaczej będzie w Unixach komercyjnych) plus inne podstawowe rzeczy (typowo: wersja jądra).
  - **uptime**: czas działania systemu od ostatniego restartu, plus informacja o ostatnim średnim obciążeniu maszyny (“load average”).
  - **cat /proc/cpuinfo** – wyświetlenie tego pliku specjalnego daje informacje o procesorze (procesorach). **Można się przyjrzeć całemu katalogowi /proc.**
  - **ps, ps -ax**: lista procesów działających w systemie. ps – procesy nasze na naszym terminalu; ps -ax – wszystkie procesy.
- **talk (oraz write)**: rozmowa z innym użytkownikiem zalogowanym na danej maszynie.

### 3. Nowoczesny tryb graficzny. Najpopularniejsze programy.

Znajomość trybu tekstowego i komend jest przydatna do rozumienia systemu. Również do pracy zdalnej. Pierwsze lata istnienia systemu - to tylko tryb tekstowy. Ale od dawna już Linux to nie tylko skomplikowane komendy i “białe literki na czarnym tle”.

#### X Window System („system okien X”)

Obsługę trybu graficznego zapewnia nam program „XFree86”. Jest to darmowa wersja zdefiniowanego w sposób bardzo ścisły i znanego w systemach UNIX systemu do pracy okienkowej „X11”.

Często mówimy potocznie na to środowisko „X Windows” (choć nie jest to poprawna formalnie nazwa).

System X Window jest zbudowany:

- modularnie. Istnieją różne moduły do obsługi różnych funkcji, np. moduł obsługi danej karty graficznej, moduł obsługi OpenGL (biblioteka animacji 3D).
- w architekturze klient-serwer.

Dzięki takiej architekturze możemy np. uruchomić program na jednym komputerze, a wyświetlać wynik jego działania na innym.

Typowe nowoczesne dystrybucje instalują się w ten sposób, że po włączeniu komputera jesteśmy proszeni o podanie nazwy użytkownika i hasła od razu w trybie graficznym.

Sam program X odpowiada za bardzo niewiele rzeczy. Po zalogowaniu się uruchamiany jest **program obsługi okien** (window manager). To ten program decyduje o wyglądzie pulpitu i kształcie/zachowaniu ikonki. Istnieje wiele takich programów. Najpopularniejsze: KDE oraz GNOME.

W naszej prezentacji posłużymy się popularną dystrybucją Red Hat Linux ([www.redhat.com](http://www.redhat.com)). Aktualna wersja: 8.0. Domyślny program obsługi okien w tej dystrybucji to GNOME.

### **Instalacja programów w Linuksie:**

1. "ręcznie" - ze źródeł, z binariów, przez instalator.
2. archiwa dystrybucji. system pakietów – RedHat: RPM. zależności. istnienie systemów półautomatycznych (RedHat: red-carpet; Debian: apt-get; PLD: apt-get, poldek).
3. problem bibliotek ("DLL hell"). W Linuksie może być zainstalowane kilka wersji danej biblioteki naraz.

### **Demonstracja wyglądu i zachowania środowiska graficznego:**

Explorer. Ikonki. Pasek zadań. Menu start.

Guziki maksymalizacji-minimalizacji.

Pulpity. Applety (pogoda, słownik, poczta).

Automatyczna obsługa plików typu PDF, PS (Postscript), obrazków.



**Klikanie jest równie łatwe, co pod Windows**, z tą różnicą, że raz zainstalowane nic się nie psuje, nie wieszka i generalnie nie robi “samo” losowo bez pytania.

## **Zamienniki typowych windowsowych programów**

(uwaga: istnieje strona ze spisem zamienników dla Linuksa:

<http://linuxshop.ru/linuxbegin/win-lin-soft-en>)

**WWW: IE => mozilla (i pochodne, np.: galeon).**

Zalety: zakładki; type-ahead-find; odporność na “wirusy”; zgodność ze standardami; nie próbuje “myśleć za użytkownika” -- można zapisać na dysk plik PDF.

Mozilla jest dostępna również pod Windows.

**E-mail: Outlook => Evolution, kmail, mozilla...**

**Albo inne podejście: dostępny zewsząd pine w trybie tekstowym.**

Odporność na “wirusy”. Evolution – prawie identyczny z OE (inna rzecz, czy to dobrze).

Mozilla jest dostępna również pod Windows.

**Office => OpenOffice.**

OpenOffice poprawnie czyta i zapisuje pliki Worda/Excelsa/Powerpointa.

Jest to dobry program. Naprawdę warto zastanowić się nad wydawaniem (niemałych!) pieniędzy na MS Office do typowych biur -- OpenOffice jest również pod Windows.

Jest pewien (niestety, poważny) problem - z zamiennikiem dla Accessa.

**Winamp => XMMS**

**GG/ICQ => istnieją niezależne od oficjalnych realizacje pod Linuksa.**

**Windows Commander => mc** (w trybie tekstowym; za to: łatwo używalny zdalnie, przez terminal)

**Programy do obsługi modemu => kppp** (prostszy niż dial-up networking)

**Programy do obsługi filmów => xine, mplayer.**

**Na zakończenie – mały “bajer”:** ssh forwarding X-ów.

---

## **Dodatek 1. Typowe problemy.**

- **instalacja Linuksa razem z Windowsami na jednym komputerze**  
Nie ma problemu. Instalator RedHata robi to standardowo.
- pewien **problem** pojawia się, jeżeli **windowsy były instalowane PO** Linuksie i zamazały “boot sector” (ściślej: MBR) dysku twardego, skutkiem czego **nie możemy uruchomić Linuksa.**

Proste rozwiązanie:

- MBR to pierwsze 512 bajtów dysku.
- po instalacji Linuksa zrzucić MBR do pliku-kopii.  
`# dd if=/dev/hda of=/root/mbr-copy bs=512 count=1`
- w razie zamazania MBR przez Windowsy:
  - uruchom Linuksa z płyty instalacyjnej
  - otwórz terminal, zamontuj partycję linuksową
  - odtwórz MBR:  
`# dd if=mbr-copy of=/dev/hda bs=512 count=1`

## **Dodatek 2. Obsługa sieci w kilku zdaniach.**

Jądro Linuksa obsługuje sieć, posługując się podstawowym pojęciem **interfejsu sieciowego.**

Interfejs sieciowy, to na przykład połączenie lokalne LAN na karcie sieciowej, połączenie kablem szeregowym z innym komputerem, połączenie modemowe, tunel VPN, itd.

Każdy interfejs jest obsługiwany niezależnie od pozostałych.

Interfejsy, podobnie jak typowe moduły, **można konfigurować, włączać i wyłączać w trakcie działania systemu.**

Na każdy interfejs można „nakładać” kolejne, coraz bardziej zaawansowane funkcje, np. mądrzejsze protokoły, czy polityki sterowania ruchem – jest to pewien „drugi wymiar” modularności.

### **Konfiguracja połączenia modemowego – kppp.**

Portom COM[1-4] odpowiadają pliki specjalne /dev/ttyS[0-3].

Reszta równie prosta, jak w windowsach.

### **Konfiguracja łącza stałego.**

Interfejsy sieciowe nazywają się eth[0,1,2...]; gdy mamy 1 kartę sieciową: eth0. W Redhacie mamy do dyspozycji graficzne narzędzie redhat-config-network. Wystarczy podać w nim parametry sieci (adres IP stały, bądź przez DHCP).

Nieco trudniej ma się sprawa z Neostradą Plus.

Potrzebny jest dodatkowy program do obsługi protokołu sieciowego PPP over Ethernet, np. pppoe. Szczegóły: [www.jfiok.org/neostrada.html](http://www.jfiok.org/neostrada.html).

Jacek Fiok