

# Kurs administrowania systemem Linux

## Zajęcia nr 7: Podstawowe czynności administracyjne w Linuksie

Instytut Informatyki Uniwersytetu Wrocławskiego

10 kwietnia 2025

# Nazwy symboliczne i odpowiadające im numery

- Komputery posługują się wyłącznie liczbami (1, 2, 4, 8, 16-bajtowymi).
- Ludzie wolą nazwy symboliczne (napisy, często długie).
- Popularne przestrzenie nazw:
  - Nazwy hostów (np. `www.ii.uni.wroc.pl`).
  - Nazwy protokołów sieciowych (różnych warstw, np. `ip`, `icmp`, `udp`).
  - Nazwy serwisów (portów, np. `ssh`, `domain`, `http`).
  - Nazwy użytkowników (np. `root`).
  - Grupy użytkowników (np. `staff`, `adm`).
- Różne rodzaje serwisów określają relacje między nazwami symbolicznymi i numerami.
- W Linuksie dostępem do nich zarządza *Name Service Switch* (GNU C Library).

# Name Service Switch (NSS)

## Rodzaje serwisów

**files** Pliki tekstowe, zwykle w katalogu `/etc`.

**db** Bazy danych Berkeley DB, zwykle w `/var/db`. Szyszy dostęp, niż do plików testowych.

**nis** Network Information Service.

**nisplus** NIS+.

**dns** Domain Name Service (tylko dla nazw hostów).

Jest też kilka innych, zależnie od konfiguracji, np. **compat** lub **ldap**.

# Name Service Switch (NSS)

serwis	zawartość	funkcja	plik w /etc
hosts	nazwy hostów i adresy IP	gethostbyname(3)	hosts
services	nazwy i numery portów sieciowych	getservent(3)	services
protocols	nazwy i numery protokołów sieciowych	getprotoent(3)	protocols
networks	nazwy sieci	getnetent(3)	networks
ethers	adresy MAC		ethers
aliases	aliasy pocztowe	getaliasent(3)	aliases
publickey	Secure RPC dla NFS i NIS+		publickey
rpc	nazwy i numery RPC	getrpcbyname(3)	rpc
passwd	informacje o użytkownikach	getpwent(3)	passwd
shadow	hasła użytkowników	getspnam(3)	shadow
group	grupy podstawowe użytkowników	getgrent(3)	group
initgroups	grupy dodatkowe użytkowników	getgrouplist(3)	group
netgroup	grupy użytkowników w sieci		netgroup

## Plik nsswitch.conf(5)

### nsswitch.conf

```
passwd:      compat nisplus
group:       compat nisplus
shadow:      compat nisplus
gshadow:     files nisplus
hosts:       files dns
networks:    files
protocols:   db files
services:    db files
rpc:         db files
netgroup:    nis
```

- Zob. też `nss(5)`.
- Wiele programów ma opcję `-n`, która wyłącza usługę NSS.
- Odpytywanie: `getent(1)`. Por. `getent hosts localhost` oraz np. `dig localhost`.

# Użytkownicy i grupy

- Baza informacji o użytkownikach (lokalnych): `/etc/passwd`
- Baza haseł: `/etc/shadow` (tylko dla roota)
- Baza informacji o grupach użytkowników: `/etc/group`
- Logi: `/var/log/{w,b}tmp`, `/var/run/utmp`, zob. `utmp(5)`, `utmpdump(1)`.
- Informacje o mnie: `id(1)`, `whoami(1)`, `logname(1)`, `groups(1)`.
- Informacje o innych: `w(1)`, `who(1)`, `pinky(1)` (d. `finger(1)`), `users(1)`, `last(1)`.
- Zob. też `who am i`, `who mom likes` itp.
- Wiele grup zezwalających na dostęp do urządzeń: `cdrom`, `floppy`, `dialout`, `bluetooth`, `audio`, `video`, `wireshark`, `kvm`, `plugdev`, `netdev` i in.
- ... i wykonywanie czynności: `staff`, `operator`, `adm` itd.
- Zwykle instalator traktuje pierwszego konfigurowanego użytkownika jako szczególnie uprawnionego.
- System weryfikacji uprawnień jest dosyć szczelny. Warto tworzyć i używać konta w celu separacji dostępu do danych (por. `lp`, `mail`, `irc`, `nobody` itd.). Oczywiście piaskownice są bardziej szczelne.

## Plik `/etc/passwd` (zob. `passwd(5)`)

Każdy wpis zajmuje jeden wiersz, 7 pól oddzielonych znakiem „:”

- 1 nazwa użytkownika (*login name*)
- 2 zaszyfrowane hasło, znak x (por. `/etc/shadow`) lub puste
- 3 numer użytkownika (w Linuksie  $\geq 1000$  dla zwykłych użytkowników)
- 4 numer grupy głównej użytkownika (por. `/etc/group`)
- 5 pole GECOS (komentarz)
- 6 katalog domowy użytkownika
- 7 powłoka startowa użytkownika (opcjonalnie, por. `chsh(1)`)

Pole GECOS (General Electric Comprehensive Operating Supervisor 1962), 5 pól oddzielonych przecinkami (por. `chfn(1)` i `login.defs(5)`).

- 1 imię i nazwisko lub nazwa programu (f)
- 2 numer pokoju (r)
- 3 numer służbowego telefonu (w)
- 4 numer prywatnego telefonu (h)
- 5 dodatkowe informacje kontaktowe (o)

9 pól w formacie /etc/passwd. Czasy w sekundach epoki Uniksa.

- 1 nazwa użytkownika (*login name*)
- 2 zaszyfrowane hasło (ew. poprzedzone ! lub \*) lub puste
- 3 data ostatniej zmiany hasła
- 4 minimalny wiek hasła do zmiany
- 5 maksymalny wiek hasła do zmiany (< poprz., zmiana zablokowana)
- 6 okres ostrzegania o konieczności zmiany hasła
- 7 okres możliwości zalogowania z wymuszeniem zmiany hasła po wygaśnięciu jego ważności
- 8 data wygaśnięcia konta (jeśli 0, tj. 1/1/1970, konto zablokowane)
- 9 pole zarezerwowane

Dodatkowo pliki:

- /etc/{passwd-,shadow-,group-,gshadow-,subuid-,subgid-} — zawartość plików sprzed ostatniej zmiany
- /var/backups/{passwd,shadow,group,gshadow}.bak  
— periodyczne kopie zapasowe (zob. /etc/cron.daily/passwd)



## Grupy

- Plik `/etc/group` — 3 pola: nazwa grupy, hasło, lista użytkowników.
- Hasła do grup zwykle w `/etc/gshadow`. Wówczas także możliwość zdefiniowania administratorów grup.
- Można być członkiem grupy lub mieć hasło do grupy.
- Polecenie `newgrp(1)`.
- Polecenia `su(1)` i `sg(1)`.

## Podużytkownicy i podgrupy

- Pliki `/etc/{subuid,subgid}`
- Potrzebne np. przy uruchamianiu kontenerów nieuprzywilejowanych.

## Jak nie zapomnieć hasła?



# Jak zmienić zapomniane hasło roota?

## Zwykle działa

- Uruchom system ratunkowy, np. z pendrive'a.
- Zamontuj *rootfs* systemu ratowanego np. w `/target/`.
- Pierwszy wiersz `/target/etc/passwd` zmień na `root::0:0:root:/root:/bin/bash`
- Uruchom system ratowany.
- Zaloguj się na konto root podając puste hasło.
- Ustaw nowe hasło roota poleceniem `passwd(1)`.

## Warianty

- Usunąć hasło z `/etc/shadow`.
- W-chroot-ować się w system ratowany i wykonać polecenie `passwd(1)`.
- Komplikacje: hasła do BIOS-u, dysku itp.

## Morał

- W razie fizycznego dostępu do komputera hasło roota nie jest zabezpieczeniem.
- Rootfs powinien się znajdować na zaszyfrowanej partycji.

## Klasyczne rozwiązania (Unix, Linux)

- Zamiast ręcznie edytować `/etc/passwd` itd. — specjalne programy.
- Bezpieczna edycja plików systemowych: `vipw(8)`, `vigr(8)` (także `visudo(8)`).
- Niskopoziomowe narzędzia `useradd(8)`, `userdel(8)`, `usermod(8)` — kompleksowe zmiany w plikach `/etc` i katalogu `/home`.
- Zakładanie wielu użytkowników na raz: `newusers(8)`.

## W Debianie

- Pakiety: `passwd`, `shadow-utils` i `adduser`.
- Narzędzia Debiana: `adduser(8)`, `deluser(8)`, `addgroup(8)`, `delgroup(8)`.
- Konfiguracja w `adduser(5)`, `deluser(5)`.

# Zarządzanie użytkownikami

- Dodanie użytkownika (Debian): `adduser user`
- Dodanie użytkownika do grupy (Debian): `adduser user group`  
Uwaga: użytkownik `user` będzie należał do tej grupy w sesji logowania rozpoczętej po wykonaniu tego polecenia — trzeba się wylogować i zalogować.
- Zablokowanie użytkownika `user`: `usermod -e 1970-01-01 user`.
- Odblokowanie użytkownika `user`: `usermod -e user`
- Zablokowanie/odblokowanie `hasła` użytkownika `user`: `passwd [-l | -u] user`
- Zmiana hasła użytkownika `user`: `passwd user`
- Zmiana czasów ważności hasła: `chage(1)`
- Wykonanie powłoki jako użytkownik `user`: `su - user`
- Wykonanie programu w podanej grupie: `sg grupa program`

# Ograniczenia dostępu do konta

## Pełna blokada

- **Użytkownik zablokowany** — nie można uruchomić procesu z UID tego użytkownika.

## Ograniczenia

- **Domyślna powłoka zablokowana** (np. `false(1)`, `nologin(1)`) — nie można się zalogować w sytuacjach, w których system wymusza użycie domyślnej powłoki (np. logowanie na konsoli, zdalnie poprzez ssh itp.). Zastosowania:
  - użytkownicy systemowi (np. demony),
  - dostęp do konta poprzez inne protokoły, np. ftp.
- **Domyślna powłoka ograniczona** (np. `rbash(1)`) — można się zalogować, ale zbiór dostępnych poleceń jest ograniczony.
- **Hasło zablokowane** — nie można się uwierzytelnić za pomocą hasła. Zastosowania:
  - użytkownicy systemowi (np. demony),
  - użytkownik uwierzytelnia się w inny sposób (zdalnie bądź lokalnie — certyfikaty, tokeny itp.).
- **Dostęp zdalny zablokowany** — zob. np. `sshd_config(5)`. Użytkownik loguje się lokalnie (terminal, su itp.). Zastosowania: np. konto root.

- `sudo` — selektywne nadawanie uprawnień do wykonywania jako root pojedynczych programów.
- W Debianie pakiet `sudo`.
- Baza danych: plik `/etc/sudoers`, zob. `sudoers(5)`.
- Nie modyfikować zwykłym edytorem! Program `visudo(8)`: brak hazardów czasowych (zakłada locka) i pozostawiania kopii zapasowych. Sprawdza poprawność składniową pliku przy zapisie.
- Także `sudoedit`, `sudo -e` — edycja plików zamiast wykonywania programów.

### Składnia w skrócie

*kto skąd=(jako-kto : z-jaką-grupą) co-wykonać*

- ALL oznacza wzorzec pasujący do wszystkiego.
- Przykład: `jan localhost=(root:staff) /bin/ip`  
— jan może uruchomić ip(8) jako root w grupie staff.
- Używać bezwzględnych ścieżek do programów!

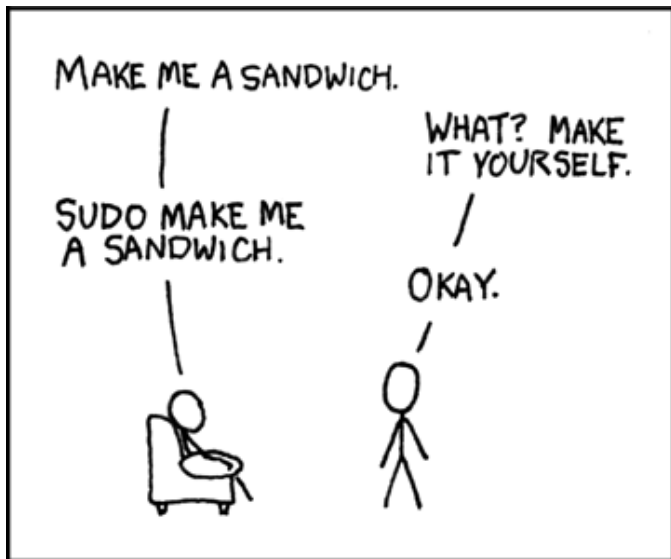
## Wykonanie pojedynczego polecenia jako root

- `su - -c polecenie` — wymaga podania hasła *roota*.
- `sudo polecenie` — wymaga *jednorazowego* podania hasła *użytkownika*.

## Własności sudo

- Użytkownik nie musi znać hasła *roota*.
- Hasło *roota* może w ogóle być wyłączone (por. Ubuntu).
- `sudo [-u user] -i` — uruchomienie powłoki jako użytkownik *user*.  
Lepsze niż `sudo su` lub `sudo /bin/bash`.
- Uwaga: `sudo` przydaje się w skryptach! (zob. także opcję `-n`)
- Pamiętaj o opcji `-k`, `-K`.
- W Ubuntu był *exploit* na `sudo -k`.
- Nie używaj bez potrzeby opcji `:NOPASSWD!`





# Hasło roota?

## Czy blokować?

- Wszystko, co nie jest używane, powinno być zablokowane.
- W niektórych dystrybucjach domyślnie hasło roota jest wyłączone.
- Instalator Debiana pyta, choć sugeruje, żeby pozostawić włączone.
- Można zablokować: `passwd -l`, a jak się nie spodoba — odblokować: `sudo passwd -u`.
- Zawsze można zresetować, jeśli nawet się zapomni.
- Uwaga: jedyne hasła, których *absolutnie nie wolno* zapomnieć, to hasła do kryptografii (zaszyfrowane partycje itp.).

## Krytyka sudo

- Program bardzo duży i skomplikowany.
- Skomplikowany plik konfiguracyjny — ryzyko błędnego skonfigurowania.
- Wykryto poważne podatności, zob. np. Animesh Jain: CVE-2021-3156: Heap-Based Buffer Overflow in Sudo (Baron Samedit).
- W OpenBSD `doas(1)`, zob. Ted Unangst: <https://flak.tedunangst.com/post/doas>.

- Kopalnia wiedzy o systemie.
- Warto je stale przeglądać i analizować.
- Katalog `/var/log/`.
- Większość plików do odczytu dla grupy `adm` — warto dodać siebie do tej grupy, by móc przeglądać logi jako zwykły użytkownik.
- Klasycznie: demon `(r)syslog`, zob. `rsyslog.conf(5)`, `rsyslogd(8)`.
- W systemd: `journalctl(1)`.
- Polecenie `logger(1)`.
- Automatyczne usuwanie starych logów: `logrotate(8)`.
- Warto wydłużyć „czas życia” logów w `/etc/logrotate.conf`, `/etc/logrotate.d/`.
- Programy `ccze(1)`, `clog(1)`, `colortail(1)`, `lwatch(1)` itp.