

- Wykłady
  - Organizacja
  - Materiały (<http://galaxy.agh.edu.pl/~boryczko/UNIX>)
  - Zaliczenie
- Laboratoria
  - Podział na grupy
    - K. Boryczko wt. 12:50, 14:40, 16:15
    - D. Król śr. 14:40, 16:15
    - M. Nabożny pt. 8:00
    - Ł. Faber pt. 9:35, 11:15, 12:50, 14:40
  - Zaliczenie
- Wolne wnioski i postulaty



# Wstęp do systemu UNIX

Cz. I:

Użytkownicy w systemie UNIX

# Definicje (I)

- System operacyjny jest programem, który działa jako pośrednik między użytkownikiem a sprzętem komputerowym.
- Cele:
  - Wydajna eksploatacja sprzętu
  - „Wygoda” w użytkowaniu sprzętu

# Definicje (2)

- System operacyjny uważany jest za zarządcę zasobów (resource allocator)
  - Zasoby (bierne): procesor, pamięć operacyjna, pamięć masowa, porty, ....
  - Strona czynna systemu: procesy
- Sposób zarządzania zasobami stał się podstawą klasyfikacji

# Historia t. I - początki

- 1965 – Bell, General Electric Company, MIT w ramach projektu MAC podejmują wysiłek stworzenia nowego systemu operacyjnego (MULTICS)
- 1969 – projekt upada. K. Thompson i D. Ritchie uruchamiają pierwszą wersję UNIX na PDP-7
- 1971 – UNIX na PDP-11. Obróbka tekstu w wydziale patentowym Bell. Języki B i C.
- 1973 – UNIX przepisany do języka C. Liczba instalacji 25.
- 1974 – pierwszy artykuł promujący UNIX.
- 1977 – Liczba instalacji 500. Nowe aplikacje.
- 1977-1982 – kilkanaście nowych wersji Systemu III
- 1983 – AT&T oficjalnie wspiera UNIX System V.
- 1983 – powstaje Free Software Foundation (FSF). Jej celem jest stworzenie systemu uniksowego wolnego od kodu z AT&T (Richard Stallman).
- 1984 – liczba instalacji dochodzi do 100.000.

# Historia t. 2 – c.d.

- 1988 – firmy Sun oraz AT&T inicjalizują porozumienie Unix International oraz powstanie System V Release 4 (SVR4) łączącego zalety Systemu V oraz BSD. Powstaje Solaris. Konkurenci zawiązują Open Software Foundation (OSF) wspierającą system OSF/I bazujący na BSD. Opublikowano specyfikację POSIX.1 – specyfikacja Uniksa.
- 1992.12.22 – prawa do Uniksa od AT&T kupuje firma Novell.
- 1993 – ukazuje się ostatnia wersja Systemu V SVR4.2MP. Novell przekazuje prawa do przydzielania marki Unix oraz regulowania Single Unix Specification (SUS) organizacji X/Open.
- 1995 Novell sprzedaje kod firmie Santa Cruz Operation (SCO), a dawne Bell Laboratories firmie Hewlett Packard.
- 1996 – z połączenia OSF oraz X/Open powstaje The Open Group, która przyznaje prawo do posługiwania się nazwą Unix.
- 1997 – druga wersja dokumentu SUS.

# Historia t. 3 - BSD

- Ok. 1977 – Computer Systems Research Group (CSRG) tworzy Wersję 1 BSD, a w roku 1978 Wersję 2.
- 1979 – VAX/BSD – współczesna organizacja pamięci wirtualnej.
- 1980 – projekt otrzymuje dofinansowanie DARPA. Powstają wersje 4 i 4.1BSD. Dla systemu zostaje stworzony Berkeley Fast File System (FFS) oraz obsługa protokołów TCP/IP oraz IPC.
- 1983 – powstaje wersja 4.2BSD. Na jej podstawie AT&T włącza obsługę sieci i pamięci wirtualnej do Systemu V.
- 1986 – pojawia się wersja 4.3BSD.
- 1993 – dostępny od tego roku system 4.4BSD oraz wolne systemy 386BSD, NetBSD, FreeBSD stają się ofiarą procesu o nieprawne korzystanie z kodu AT&T. W wyniku ugody powstaje wersja 4.4BSD Lite pozbawiona spornego kodu. Proces zahamował rozwój wersji BSD i związanych z nią projektów (386BSD upadła). Potem ukazuje się 4.4BSD Lite 2 – podstawa dla projektów FreeBSD, NetBSD, OpenBSD i BSD/OS oraz nigdy nie ukończonego Rhapsody firmy Apple Computer.



# Historia t. 4 - pozostali

- Nazwa Unix przez długi okres powstawania systemu była zastrzeżoną nazwą handlową. Stąd producenci sprzętu komputerowego dostarczają własne implementacje systemu Unix pod charakterystycznymi nazwami (AIX, HP-UX). Żadna z nich nie jest czystą wersją Systemu V lub BSD.
- 1991 – Linus Torwalds rozpoczął pracę nad jądrem systemu operacyjnego o nazwie Linux, które w połączeniu z narzędziami GNU (GNU's Not Unix – akronim rekurencyjny) stworzyło funkcjonalnie pełny, uniksopodobny system operacyjny. Obecnie zarejestrowanych jest ok. 600 dystrybucji. Na popularność wpłynęło:
  - zaproszenie do jego tworzenia szerokiej społeczności użytkowników,
  - długa lista architektur komputerowych (w tym niszowych), na które jest dostępny.
- Obecnie wielu producentów systemów uniksowych ma w swojej ofercie własne odmiany Linuksa lub aktywnie uczestniczy w rozwijaniu opartych na nim technologiach (IBM, SGI, Sun).



# Zalety

- Napisany w języku wysokiego poziomu (łatwo przenaszalny między platformami)
- Prosty interfejs dostosowany do potrzeb użytkownika
- Hierarchiczny system plików
- Spójny format plików (strumień bajtów)
- Prosty i spójny interfejs z urządzeniami zewnętrznymi
- Wielodostępny i wieloprocessowy
- Zasłania przed użytkownikiem architekturę sprzętu

# Użytkownicy

- W świecie zewnętrznym rozróżniani w/g nazwy. W systemie w/g numeru identyfikacyjnego.
- Nazwa użytkownika jest niepowtarzalna. W systemie może istnieć wielu użytkowników o tym samym numerze.
- Liczba użytkowników w systemie ograniczona do wartości  $2^{16}$ .

# Podstawowy plik konfiguracyjny ( /etc/passwd)

- Plik tekstowy.
- Zawartość może odczytać każdy użytkownik.
- Informacja zapisana jest w 7-miu kolumnach oddzielonych znakiem „:”.

```
[bory@elrond ~]$ ls -l /etc/passwd
```

```
-rw-r--r--  1 root root 1812 Jul  4 01:00 /etc/passwd
```

# /etc/passwd – kolumna I

- Nazwa użytkownika (login name).
- Litery, cyfry, niektóre znaki specjalne.
- Wielkość liter rozróżnialna.
- Długość do 8-32 znaków.

```
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
.....
mariusz:x:501:503:./home/mariusz:/bin/bash
zbik:x:502:504:./home/zbik:/bin/bash
vrobel:x:503:505:./home/vrobel:/bin/bash
ksw1:x:504:506:./home/KSW/ksw1:/bin/bash
```

# /etc/passwd – kolumna 2

- Dawniej – zakodowana postać hasła.
- Obecnie – niektóre systemy przechowują informacje o stanie hasła.
- Postać zakodowana hasła przeniesiona do:
  - Plik ./shadow
  - Plik ./security/user

```
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
.....
mariusz:x:501:503::/home/mariusz:/bin/bash
zbik:x:502:504::/home/zbik:/bin/bash
vrobel:x:503:505::/home/vrobel:/bin/bash
ksw1:x:504:506::/home/KSW/ksw1:/bin/bash
```

# /etc/passwd – kolumna 3

- Numer identyfikacyjny użytkownika (UID).
- Może się powtarzać (tworzenie kontekstów).
- wg niego system rozpoznaje użytkownika i działa na prawach własności
- Użytkownicy „zwykli” zaczynają się od pewnej wartości
- 0 – root – najistotniejszy w systemie

```
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
.....
mariusz:x:501:503::/home/mariusz:/bin/bash
zbik:x:502:504::/home/zbik:/bin/bash
vrobel:x:503:505::/home/vrobel:/bin/bash
ksw1:x:504:506::/home/KSW/ksw1:/bin/bash
```

# /etc/passwd – kolumna 4

- Numer grupy podstawowej użytkownika (GID).
- Użytkownicy mogą należeć do wielu grup, ale jedna jest grupą podstawową – właściciel grupowy.
- Numery grup są niepowtarzalne.
- Przynależność do grupy decyduje o uprawnieniach.
- Opis grup w pliku /etc/group.

```
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
.....
mariusz:x:501:503:./home/mariusz:/bin/bash
zbik:x:502:504:./home/zbik:/bin/bash
vrobel:x:503:505:./home/vrobel:/bin/bash
ksw1:x:504:506:./home/KSW/ksw1:/bin/bash
```



# /etc/passwd – kolumna 5

- Opis użytkownika.
- Zawartość zależy od administratora systemu.
- Niektóre systemy dzielą ją na 4 podkolumny oddzielone przecinkami (tel. domowy, służbowy, nr pokoju, imię i nazwisko)
- Zmiana zawartości – komenda *chfn*

```
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
.....
mariusz:x:501:503::/home/mariusz:/bin/bash
zbik:x:502:504::/home/zbik:/bin/bash
vrobel:x:503:505::/home/vrobel:/bin/bash
ksw1:x:504:506::/home/KSW/ksw1:/bin/bash
```

# /etc/passwd – kolumna 6

- Bezwzględna ścieżka dostępu do katalogu domowego użytkownika (HOME).
- Jeśli podczas podłączania użytkownika do systemu nie występuje – użytkownik nie zostanie podłączony lub znajdzie się w /.

```
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
.....
mariusz:x:501:503:~/home/mariusz:/bin/bash
zbik:x:502:504:~/home/zbik:/bin/bash
vrobel:x:503:505:~/home/vrobel:/bin/bash
ksw1:x:504:506:~/home/KSW/ksw1:/bin/bash
```

# /etc/passwd – kolumna 7

- Bezwzględna ścieżka dostępu do podstawowego interpretera użytkownika (SHELL).
- Przy zmianie (chsh) sprawdzane, czy występuje jego definicja w pliku /etc/shells.
- nologin – najprostsze blokowanie dostępu użytkownika

```
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
.....
mariusz:x:501:503:./home/mariusz:/bin/bash
zbik:x:502:504:./home/zbik:/bin/bash
vrobel:x:503:505:./home/vrobel:/bin/bash
ksw1:x:504:506:./home/KSW/ksw1:/bin/bash
```

# Idea podziału na grupy

- Łatwe „rozkładanie” praw w systemie
- Ułatwienie zarządzania użytkownikami
- Podniesienie bezpieczeństwa systemu

```
[bory@elrond ~]$ ls -l /etc/group  
-rw-r--r-- 1 root root 674 Jul  4 01:00 /etc/group
```

# /etc/group

- Plik tekstowy
- Jedna linia opisuje jedną grupę
- Każda linia podzielona jest na 4 pola oddzielone znakiem „:”

```
root:x:0:root
bin:x:1:root,bin,daemon
daemon:x:2:root,bin,daemon
sys:x:3:root,bin,adm
adm:x:4:root,adm,daemon
.....
mariusz:x:503:
zbik:x:504:
ksw:x:506:
```

# /etc/group – kolumna I

- Nazwa grupy
- Unikalna w systemie
- Linux po dodaniu nowego użytkownika defaultowo tworzy grupę o nazwie takiej jak nowododawany użytkownik (ustawienia w plikach: */etc/default/useradd* oraz */etc/login.defs* )

```
root:x:0:root
bin:x:1:root,bin,daemon
daemon:x:2:root,bin,daemon
sys:x:3:root,bin,adm
adm:x:4:root,adm,daemon
.....
mariusz:x:503:
zbik:x:504:
ksw:x:506:
```

# /etc/group – kolumna 2

- Pole historyczne.
- Dawniej hasło grupowe konieczne do administrowania grupą (dodawanie, usuwanie członków).
- Obecnie informacja o stanie hasła. Hasło właściwe w:
- ./gshadow
- ./security/group

```
root:x:0:root
bin:x:1:root,bin,daemon
daemon:x:2:root,bin,daemon
sys:x:3:root,bin,adm
adm:x:4:root,adm,daemon
.....
mariusz:x:503:
zbik:x:504:
ksw:x:506:
```



# /etc/group – kolumna 3

- Numer grupy w systemie
- W obrębie systemu niepowtarzalny

```
root:x:0:root
bin:x:1:root,bin,daemon
daemon:x:2:root,bin,daemon
sys:x:3:root,bin,adm
adm:x:4:root,adm,daemon
.....
mariusz:x:503:
zbik:x:504:
ksw:x:506:
```

# /etc/group – kolumna 4

- Lista członków grupy
- Nazwy użytkowników w systemie (login name) oddzielone przecinkami

```
root:x:0:root
bin:x:1:root,bin,daemon
daemon:x:2:root,bin,daemon
sys:x:3:root,bin,adm
adm:x:4:root,adm,daemon
.....
mariusz:x:503:
zbik:x:504:
ksw:x:506:
```

# /etc/shadow

- Plik tekstowy
- Budowa linikowa – jedna linijka-jeden użytkownik
- Każda linijka ma 9 pól oddzielonych „:”
- Prawa dostępu umożliwiają edycję tylko użytkownikowi root

```
[bory@elrond ~]$ ls -l /etc/shadow
```

```
-r----- 1 root root 1366 Nov 27 22:13 /etc/shadow
```

# /etc/shadow – kolumna I

- Nazwa użytkownika w systemie (login name)

**root:\$1\$mkALRPUE\$dR.jY4OKQ0hv6pdJRxmMU.:12348:0:99999:7:::**

**bin:\*:12348:0:99999:7:::**

**daemon:\*:12348:0:99999:7:::**

**.....**

**ksw1:\$1\$6.QNeIxX\$Up0Ylp.FbFfhb./RJ16H31:12731:0:99999:7:::**

**ksw2:\$1\$W4Xxob3d\$St7cwB80UqgDwzXZ6OSRA0:12731:0:99999:7:::**

# /etc/shadow – kolumna 2

- Postać zakodowana hasła
- \* oznacza iż żadne hasło nie jest poprawne
- !! hasło nie zostało ustawione
- \$!\$ kodowanie funkcją haszującą md5 (stała długość, zależy od postaci jawnej, sól...)
- Xiaoyun Wang, „How to Break MD5 and Other Hash Function”, 27 lipiec 2008
- SHA-256, SHA-512

**root:\$1\$mkALRPUE\$dR.jY4OKQ0hv6pdJRxmMU.:12348:0:99999:7:::**

**bin:\*:12348:0:99999:7:::**

**daemon:\*:12348:0:99999:7:::**

**.....**

**ksw1:\$1\$6.QNeIxX\$Up0Ylp.FbFfhb./RJ16H31:12731:0:99999:7:::**

**ksw2:\$1\$W4Xxob3d\$St7cwB80UqgDwzXZ6OSRA0:12731:0:99999:7:::**

# /etc/shadow – kolumna 3

- Liczba dni, licząc od 1 stycznia 1970r., kiedy hasło było ostatni raz zmienione
- Wykorzystywane do wymuszania częstości zmiany hasła

**root:\$1\$mkALRPUE\$dR.jY4OKQ0hv6pdJRxmMU.:12348:0:99999:7:::**

**bin:\*:12348:0:99999:7:::**

**daemon:\*:12348:0:99999:7:::**

**.....**

**ksw1:\$1\$6.QNeIxX\$Up0Ylp.FbFfhb./RJ16H31:12731:0:99999:7:::**

**ksw2:\$1\$W4Xxob3d\$St7cwB80UqgDwzXZ6OSRA0:12731:0:99999:7:::**

# /etc/shadow – kolumna 4

- Liczba dni przed upływem których zmiana hasła nie jest możliwa
- Wykorzystywane wraz z listą historii haseł

**root:\$1\$mkALRPUE\$dR.jY4OKQ0hv6pdJRxmMU.:12348:0:99999:7:::**

**bin:\*:12348:0:99999:7:::**

**daemon:\*:12348:0:99999:7:::**

**.....**

**ksw1:\$1\$6.QNeIxX\$Up0Ylp.FbFfhb./RJ16H31:12731:0:99999:7:::**

**ksw2:\$1\$W4Xxob3d\$St7cwB80UqgDwzXZ6OSRA0:12731:0:99999:7:::**



# /etc/shadow – kolumna 5

- Liczba dni po upływie których konieczna jest zmiana hasła
- Wykorzystywane do wymuszania częstości zmiany hasła

```
root:$1$mkALRPUE$dR.jY4OKQ0hv6pdJRxmMU.:12348:0:99999:7:::  
bin:*:12348:0:99999:7:::  
daemon:*:12348:0:99999:7:::  
.....  
ksw1:$1$6.QNeIxX$Up0Ylp.FbFfhb./RJ16H31:12731:0:99999:7:::  
ksw2:$1$W4Xxob3d$St7cwB80UqgDwzXZ6OSRA0:12731:0:99999:7:::
```

# /etc/shadow – kolumna 6

- Liczba dni, jaka dzieli hasło od przedawnienia, kiedy użytkownik będzie o tym fakcie powiadomiony.
- Informacja porządkowa

```
root:$1$mkALRPUE$dR.jY4OKQ0hv6pdJRxmMU.:12348:0:99999:7:::  
bin:*:12348:0:99999:7:::  
daemon:*:12348:0:99999:7:::  
.....  
ksw1:$1$6.QNeIxx$Up0Ylp.FbFfhb./RJ16H31:12731:0:99999:7:::  
ksw2:$1$W4Xxob3d$St7cwB80UqgDwzXZ6OSRA0:12731:0:99999:7:::
```

# /etc/shadow – kolumna 7

- Liczba dni, po wygaśnięciu hasła, kiedy konto jest blokowane (interwencja administratora)

```
root:$1$mkALRPUE$dR.jY4OKQ0hv6pdJRxmMU.:12348:0:99999:7:::  
bin:*:12348:0:99999:7:::  
daemon:*:12348:0:99999:7:::  
.....  
ksw1:$1$6.QNeIxx$Up0Ylp.FbFfhb./RJ16H31:12731:0:99999:7:::  
ksw2:$1$W4Xxob3d$St7cwB80UqgDwzXZ6OSRA0:12731:0:99999:7:::
```

# /etc/shadow – kolumna 8

- Liczba dni, licząc od 1 stycznia 1970r., po upływie których konto jest blokowane. Wykorzystywane przy zakładaniu kont tymczasowych.
- Kolumna 9 – pole zarezerwowane

**root:\$1\$mkALRPUE\$dR.jY4OKQ0hv6pdJRxmMU.:12348:0:99999:7:::**

**bin:\*:12348:0:99999:7:::**

**daemon:\*:12348:0:99999:7:::**

**.....**

**ksw1:\$1\$6.QNeIxX\$Up0Ylp.FbFfhb./RJ16H31:12731:0:99999:7:::**

**ksw2:\$1\$W4Xxob3d\$St7cwB80UqgDwzXZ6OSRA0:12731:0:99999:7:::**

# /etc/gshadow

Przechowuje hasło grupowe oraz informacje o administratorach grupy

- kolumna 1 – nazwa grupy w systemie
- kolumna 2 – hasło grupowe
- kolumna 3 – lista administratorów grupy
- kolumna 4 – lista członków grupy

**root:::root**

**bin:::root,bin,daemon**

.....

**ksw:2YtkbrJqHJqEw:bory:**

# BSD

- `/etc/passwd` – dla zgodności
- `/etc/master.passwd`

```
root:$I$qrIT0nO$t/IneYLk4DASrLgrKRSS.0:0:0::0:0:Charlie &:/root:/bin/csh
toor:*:0:0::0:0:Bourne-again Superuser:/root:
daemon:*:1:1::0:0:Owner of many system processes:/root:/usr/sbin/nologin
operator:*:2:5::0:0:System &:/usr/sbin/nologin
.....
polkit:*:562:562::0:0:PolicyKit Daemon User:/nonexistent:/sbin/nologin
wacek:$I$UjAvonVC$IcY8vhICMJD/zGk/cbEcmI:1001:1001::1235865600:0:Waclaw
Kowalski:/home/wacek:/bin/sh
```

- `/etc/pwd.db` – baza danych (indeksowana dla zwiększenia wydajności)

# /etc/security/user

default:

```
admin = false
login = true
su = true
daemon = true
rlogin = true
sugroups = ALL
admgroups =
ttys = ALL
auth1 = SYSTEM
auth2 = NONE
tpath = nosak
umask = 022
expires = 0
SYSTEM = "compat"
```



# /etc/security/user

```
SYSTEM = "compat"  
logintimes =  
pwdwarntime = 0  
account_locked = false  
loginretries = 0  
histexpire = 0  
histsize = 0  
minage = 0  
maxage = 0  
maxexpired = -1  
minalpha = 0  
minother = 0  
minlen = 0  
mindiff = 0  
maxrepeats = 8
```

# /etc/security/user

```
dictionlist =  
pwdchecks =  
dce_export = false
```

root:

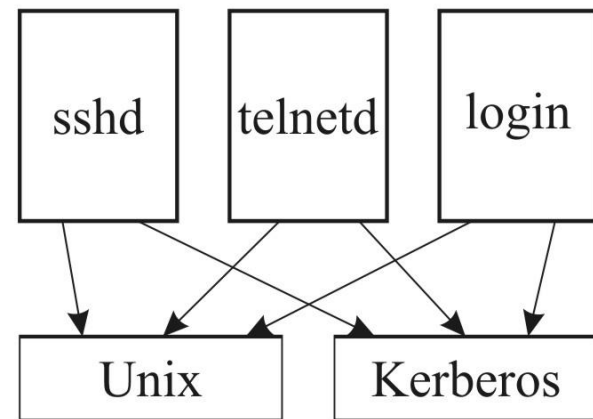
```
admin = true  
SYSTEM = "compat"  
loginretries = 0  
account_locked = false  
ttys = ALL  
rlogin = false  
login = true
```

# PAM

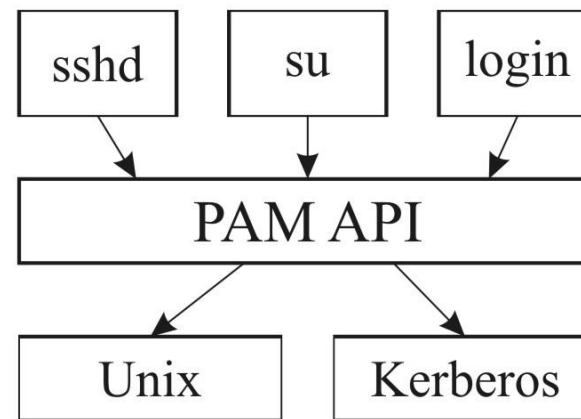
- Pluggable Authentication Modules (PAM)
- Zaproponowany przez firmę SUN, wprowadzony przez inne firmy oraz pojawił się w wielu dystrybucjach systemu Linux.
- Dostępność kodu źródłowego oraz jasne i proste reguły tworzenia modułów przekładające się bezpośrednio na elastyczną politykę bezpieczeństwa zadecydowały o jego rosnącej popularności.
- Biegła jego znajomość wymaga długiej praktyki.

# PAM - architektura

a)



b)



Aplikacje  
*Interfejsy*  
Konfiguracja  
*Interfejsy*  
Mechanizmy

# Pliki i katalogi PAM

- W systemach SunOS/SOLARIS
  - `/etc/pam.conf` – opis sposobu użycia modułów.
  - `/lib/security` (`/lib64/security/`) – moduły.
- Systemy linuksowe
  - `/etc/pam.conf` – opis użycia modułów. Obecnie jeśli istnieje katalog `/etc/pam.d` to on zawiera opis sposobu użycia modułów, a plik `/etc/pam.conf` jest ignorowany. Nazwa pliku odpowiada nazwie usługi lub aplikacji.
  - `/lib/security` (`/lib64/security/`) – moduły.

# Plik konfiguracyjny (setup)

```
#%PAM-1.0
```

```
auth      sufficient  pam_rootok.so
```

```
auth      include     system-auth
```

```
account   required    pam_permit.so
```

```
session   required    pam_permit.so
```

- Plik tekstowy o budowie linijkowej.
- Komentarz od znaku # do końca linii.
- Jedna linia zawiera opis sposobu wywołania modułu oraz dalszego postępowania zależnego od sposobu zakończenia wykonania modułu.
- Wykonywane sekwencyjnie. Tworzy sekwencje lub stos wywołań modułów.

# Budowa ogólna pliku konfiguracyjnego (I)

[nazwa usługi] rodzaj\_zadania znacznik uruchamiany\_moduł argumenty

- Nazwa usługi (lub aplikacji) pojawia się jeśli konfiguracja jest oparta na pliku /etc/pam.conf. Pliki konfiguracyjne w katalogu /etc/pam.d nie wymagają jej.
- System PAM wyróżnia 4 rodzaje (typy) zadania:
  - **auth** – autentykacja, sprawdzenie nazwy i hasła użytkownika i na tej podstawie przydzielenie lub odrzucenie dostępu.
  - **account** – zarządzanie atrybutami konta (hasła, blokowanie konta).
  - **password** – ogólne zasady zarządzania hasłem (postać, złożoność, historia).
  - **session** – zarządzanie sesją dla jej konfiguracji przed uzyskaniem przez użytkownika dostępu do zasobów systemu.



# Budowa ogólna pliku konfiguracyjnego (2)

[nazwa usługi] rodzaj\_zadania znacznik uruchamiany\_moduł argumenty

- Znacznik (kontrolny) określa w jaki sposób system PAM ma zareagować na zakończenie z sukcesem lub porażką konkretnego modułu. Podstawowe znaczniki to:
  - **required** – wykonanie modułu zakończone sukcesem jest konieczne do pomyślnego zakończenia wykonania modułów danego typu. Informacja przekazywana jest do aplikacji po zakończeniu wykonywania modułów danego typu.
  - **requisite** – jak required ale informacja do aplikacji jest zwracana po wykonaniu modułu danego.
  - **sufficient** – zakończone sukcesem wykonanie modułu wystarcza, aby uznać za wykonanie zakończone sukcesem modułów tego typu.
  - **optional** – moduł nie jest uważany za krytyczny. PAM ignoruje taki moduł w momencie określania czy działanie modułów tego typu uznać za zakończone sukcesem czy porażką.
  - **include** – wymusza uwzględnienie wszystkich linii opisujących ten sam rodzaj zadania zapisanych w pliku, którego nazwa pojawiła się jako argument znacznika. Jeśli wartościowanie kolejnej linii zostało określone jako *done* lub *die* to kolejne linie nie będą wartościowane.
  - **substack** - jak include, ale jeśli wartościowanie kolejnej linii zostało określone jako *done* lub *die* to kolejne linie będą wartościowane.

# Budowa ogólna pliku konfiguracyjnego (3)

[nazwa usługi] rodzaj\_zadania znacznik uruchamiany\_moduł argumenty

- Znacznik kontrolny może zostać zapisany w bardziej złożonej, nowej formie:

wartosc1=akcja1 wartosc2=akcja2 .....wartoscn=akcjn

- Jako *akcjn* może pojawić się liczba całkowita mówiąca ile kolejnych linii przeskoczyć w wartościowaniu co umożliwia tworzenie ścieżek wykonań.
- Jako wartość może pojawić się również jeden z napisów:
  - **ignore** - jeśli wartość ta została użyta w sekwencji (stosie) wywołania modułów to wartość zwrócona przez bieżący moduł ma nie być uwzględniana w wartościowaniu całej sekwencji.
  - **bad** – oznacza, że wartość zwrócona przez moduł może posłużyć do określenia przyczyny zakończenia niepowodzeniem jego wykonywania. Po takim zakończeniu wykonywane są kolejne moduły z sekwencji.
  - **die** – jak bad, ale nie są wykonywane kolejne moduły z sekwencji.
  - **ok** – wartość zwrócona przez bieżący moduł określa sposób wykonania wszystkich modułów z sekwencji.

# Budowa ogólna pliku konfiguracyjnego (4)

- Kolejne wartości:
- **done** – działanie jak ok, ale wykonywanie modułów kończy się na bieżącym module a wartość zwrócona aplikacji jest wartością zwróconą przez ten moduł.
- **reset** – zeruje stos wartości zwróconych przez uprzednio wykonane moduły.
- Istnieją odpowiedniki dla: **required**, **requisite**, **sufficient**, **optional**.

[nazwa usługi] rodzaj\_zadania znacznik uruchamiany\_moduł argumenty

- Kolejne kolumny zawierają ścieżki dostępu do uruchamianych modułów. Jeśli jest to ścieżka względna, to moduły są uruchamiane z katalogu */lib/security* (lub */lib64/security*).
- Ostatnia kolumna to opcjonalne argumenty uruchomienia modułu.

# Przykład konfiguracji – lista historii haseł

- Utworzenie listy historii haseł wymaga modyfikacji pliku */etc/pam.d/system-auth*. Moduł odpowiedzialny to *pam\_unix*.

- Linie:

```
password    sufficient    pam_unix.so sha512 shadow nullok try_first_pass use_authtok
```

- Zastępujemy linią:

```
password    sufficient    pam_unix.so sha512 shadow nullok try_first_pass use_authtok remember=5
```

- Hasła w postaci zaszyfrowanej będą zapisywane w pliku */etc/security/opasswd*.
- Jeśli plik ten nie istnieje, to należy go utworzyć. Właścicielem indywidualnym musi być użytkownik *root*, a grupowym grupa *root*.
- Wg przykładowej konfiguracji pamiętanych będzie ostatnich pięć haseł.

# Przykład konfiguracji – blokowanie konta po nieudanych próbach logowania

- Blokowanie konta po określonej liczbie nieudanych prób logowania wymaga modyfikacji pliku */etc/pam.d/system\_auth*. Moduł odpowiedzialny to *pam-tally*.

- Dodajemy linię:

`auth required pam_tally.so onerr=fail no_magic_root`

Moduł został wywołany z dwoma opcjami:

1. `onerr=fail` – do pliku dziennika będą zapisywane zdarzenia zakończone niepowodzeniem,
2. `no_magic_root` – wymusza restrykcje również dla użytkownika *root*.

- Oraz linię:

`account required pam_tally.so deny=5 no_magic_root reset`

Opcje wywołania modułu *pam\_tally* to:

- `deny=5` – jej wartość to liczba nieudanych logowań po których zostanie zablokowane konto.
- Poprawne podłączenie się do systemu zeruje licznik połączeń niepoprawnych.
- Lista niepoprawnych logowań w pliku */var/log/faillog*.
- Zarządzanie poleceniem: *faillog*

# Ograniczanie zasobów systemu

- Podstawowy plik konfiguracyjny to: `/etc/security/limits.conf`
- Plik tekstowy, budowa linikowa. Jedna linia opisuje ograniczenie danego zasobu dla użytkownika lub grupy użytkowników.
- Czytany sekwencyjnie, stąd możliwość wyjątków.
- Plik `/etc/security/limits.conf` jest plikiem konfiguracyjnym modułu `pam_limits`. Stąd aby ograniczenia były efektywne, konieczne jest wywołanie modułu `pam_limits` w plikach PAM opisujących sposób autentykacji użytkownika. Przykładowo:

system-auth:	session	required	pam_limits.so
sudo:	session	required	pam_limits.so
password-auth:	session	required	pam_limits.so
fingerprint-auth:	session	required	pam_limits.so



# Plik `/etc/security/limits.conf` (I)

- Każda linia ma następującą składnię:  
zakres typ\_ograniczenia zasób wartość
- Zakres może być wyspecyfikowany jako:
  - Nazwa użytkownika.
  - Nazwa grupy poprzedzona znakiem @.
  - Znak \* dla domyślnego zakresu (każdego).
  - Znak % dla ograniczenia maksymalnej liczby połączeń użytkowników z danej grupy.
  - Zakres numerów użytkowników w formacie *początkowy:końcowy*.
  - Zakres numerów grup użytkowników w formacie *@początkowy:@końcowy*.



# Plik `/etc/security/limits.conf` (2)

zakres typ\_ograniczenia zasób wartość

- Typ ograniczenia przyjmuje następujące wartości symboliczne:
  - **soft** – ograniczenie miękkie, możliwe do przekroczenia. Jest traktowane jako domyślne w systemie. Zazwyczaj wymaga zdefiniowania ograniczenia twardego.
  - **hard** – ograniczenie nałożone przez administratora systemu, niemożliwe do przekroczenia przez zwykłego użytkownika.
  - - umożliwia wprowadzenie obu ograniczeń jednocześnie.

# Plik `/etc/security/limits.conf` (3.1)

zakres typ\_ograniczenia zasób wartość

## Możliwe zasoby to:

- **core** – maksymalny rozmiar pliku core (kB).
- **data** – maksymalny rozmiar segmentu danych procesu (kB).
- **filesize** – maksymalny rozmiar pliku (kB).
- **memlock** – maksymalny rozmiar pamięci zaalokowanej (kB).
- **nofile** – maksymalna liczba otwartych plików.
- **rss** maksymalny rozmiar pamięci rss (od 2.4.30 ignorowany)(kB).
- **stack** – maksymalny rozmiar segmentu stosu (kB).
- **cpu** – czas wykorzystania procesora przez proces (min).
- **nproc** – maksymalna liczba uruchomionych procesów.
- **as** – maksymalny adres przestrzeni pamięci (kB).
- **maxlogins** – maksymalna liczba połączeń do systemu. Nie dotyczy użytkownika o UID=0.

# Plik `/etc/security/limits.conf` (3.2)

zakres typ\_ograniczenia zasób wartość

## Możliwe zasoby (cd):

- **maxsyslogins** – maksymalna, sumaryczna liczba połączeń do systemu.
- **maxpriority** – maksymalna wartość priorytetu dla zadań użytkownika.
- **locks** – maksymalna liczba „zamkniętych” plików (od 2.6).
- **sigpending** – maksymalna długość kolejki sygnałów dla procesu.
- **msqueue** – maksymalna długość kolejki komunikatów (od 2.6) (kB).
- **nice** – maksymalna wartość parametru NICE procesu (z przedziału [-20:19]).
- **rtprio** – maksymalna wartość priorytetu dla zadań czasu rzeczywistego.

# Plik `/etc/security/limits.conf` (4)

zakres typ\_ograniczenia zasób wartość

## Wartości – zasady ogólne:

- Ustawienia dla użytkowników indywidualnych mają pierwszeństwo przed ustawieniami dla grup użytkowników.
- Wartość limitu musi być możliwa do ustawienia ze względu na zasoby systemu. Jeśli jest to niemożliwe znacznik `required` powoduje iż połączenie nie jest możliwe.
- Każdy zasób, którego wartości mogą być **-1**, **unlimited** lub **infinity** nie jest w przypadku ich użycia ograniczany. Wyjątek: *priority* i *nice*.
- Błędy konfiguracyjne raportuje moduł *pam\_limits*.

# Plik /etc/security/limits.conf (5)

## przykład

@students	-	maxlogins	7
@students	hard	nproc	100
@students	hard	rss	10240
@students	hard	cpu	10
@workers	-	maxlogins	7
@workers	hard	nproc	250
@workers	hard	rss	100240
@workers	hard	cpu	20
bory	-	maxlogins	30

# adduser (useradd) (I)

- Dodawanie użytkownika do systemu
- Opcje przyjmują wartości defaultowe (pliki */etc/default/useradd*, */etc/login.defs*, katalog */etc/skel*)

```
usage: adduser [-u uid [-o]] [-g group] [-G group,...]
              [-d home] [-s shell] [-c comment] [-m [-k template]]
              [-f inactive] [-e expire ] [-p passwd] [-M] [-n] [-r] name
adduser      -D [-g group] [-b base] [-s shell]
              [-f inactive] [-e expire ]
```



# adduser (useradd) (2)

- Plik */etc/default/useradd*

```
# useradd defaults file
GROUP=100
HOME=/home
INACTIVE=-1
EXPIRE=
SHELL=/bin/bash
SKEL=/etc/skel
CREATE_MAIL_SPOOL=yes
```

- Plik */etc/login.defs*
  - Zakresy UID oraz GID
  - Wartość domyślnej umaski
  - Metoda szyfrowania hasła



# passwd

- Zwykły użytkownik – zmiana hasła
- root – ustawienia parametrów hasła

Usage: passwd [OPTION...] <accountName>

-k, --keep-tokens      keep non-expired authentication tokens  
-d, --delete            delete the password for the named account (root only)

-x, --maximum=DAYS    maximum password lifetime (root only)  
-n, --minimum=DAYS    minimum password lifetime (root only)  
-w, --warning=DAYS    number of days warning users receives before  
                         password expiration (root only)  
-i, --inactive=DAYS    number of days after password expiration when an  
                         account becomes disabled (root only)  
-S, --status            report password status on the named account (root  
                         only)  
--stdin                read new tokens from stdin (root only)

# usermod

- Zmiana parametrów użytkownika.

usage: usermod [-u uid [-o]] [-g group] [-G group,...]  
[-d home [-m]] [-s shell] [-c comment] [-l new\_name]  
[-f inactive] [-e expire ] [-p passwd] [-L|-U] name

# gpsswd

- Administrowanie grupą (hasło grupowe, administrator).

gpsswd group

gpsswd -a user group

gpsswd -d user group

gpsswd -R group

gpsswd -r group

gpsswd [-A user,...] [-M user,...] group

# groupadd

- Utworzenie nowej grupy w systemie.

usage: groupadd [-g gid [-o]] [-r] [-f] group

# groupdel

- Usunięcie grupy z systemu.
- Usuwana grupa nie może być podstawową dla żadnego użytkownika w systemie.
- Argument to nazwa lub numer grupy.

usage: groupdel group

# userdel

- Usuwanie użytkownika z systemu.
- Użytkownik nie może być zalogowany.
- Argument to nazwa lub UID.
- -r usunięcia zawartości katalogu osobistego oraz pliku poczty.
- Pamiętać o /tmp, /var.

usage: userdel [-r] name

# Konsystentność zbiorów

- vipw – skrypt: edytor + prosty program do sprawdzania poprawności składni
- pwck - sprawdza zgodność */etc/passwd* z */etc/shadow* pod kątem:
  - liczby pól
  - unikalność nazwy użytkownika
  - poprawność UID i GIDs
  - poprawność grupy podstawowej
  - występowanie katalogu domowego
  - występowanie shella logującego
- -r tylko odczyt

Usage: pwck [-q] [-r] [-s] [passwd [shadow]]



# Konsystentność zbiorów

- grpck – bada pod kątem zgodności */etc/group* z */etc/gshadow*:
  - poprawność liczby pól
  - unikalność nazwy grupy
  - poprawność administratora i listy członków
- -r tylko do odczytu

Usage: grpck [-r] [-s] [group [gshadow]]

# Podłączanie do systemu

- Konsola
- Zmiana kontekstu
- Zdalnie, poprzez sieć
- Z wykorzystaniem terminali znakowych (tty)

# Zmiana kontekstu

- su – switch user

- Składnia:

su [-] [login\_name]

- Użycie „-” zapewnia aktualizację środowiska do środowiska użytkownika docelowego.  
Pominięcie pozostawia bieżące
- Pominięcie użytkownika docelowego przełącza na użytkownika root

# W

- Udziela informacji o użytkownikach aktualnie zalogowanych i wykonywanych przez nich programach.
- Plik: /etc/utmp

```
09:53:41 up 176 days, 2:03, 19 users, load average: 0.15, 0.08, 0.06
USER      TTY      FROM          LOGIN@  IDLE   JCPU   PCPU   WHAT
bory      :0        -              30Aug04 ?      0.00s 35.11s /usr/bin/gnome-
bory      pts/0     :0.0           30Aug04 176days 0.02s 0.02s bash
.....
bory      pts/24    -              15Nov04 2days 3.06s 3.06s bash
bory      pts/10    s4.msi.umn.edu 6:44am 2:18m 0.08s 0.01s less clustering
```

# who

- Udziela informacji o zalogowanych użytkownikach
- Plik `/var/run/utmp`

bory	:0	Aug 30 08:51
bory	pts/0	Aug 30 08:51 (:0.0)
bory	pts/1	Aug 30 08:51 (:0.0)
bory	pts/2	Aug 30 08:51 (:0.0)
bory	pts/3	Nov 28 13:05
bory	pts/4	Nov 28 13:55

# last

- Udostępnia informację o historii logowań użytkowników
- Plik /var/log/wtmp

bory	pts/10	s4.msi.umn.edu	Tue Feb 22 06:44	still logged in
vrobelt	pts/10	r9.ists.pl	Mon Feb 21 20:47 - 20:49	(00:02)
bory	pts/10	wfitj16e.ftj.agh	Mon Feb 21 13:30 - 13:47	(00:17)
vrobelt	pts/10	r9.ists.pl	Thu Feb 17 17:58 - 17:58	(00:00)
zbik	pts/17	atwork.zbik.org	Wed Feb 16 11:25 - 11:38	(00:12)
vrobelt	pts/17	r9.ists.pl	Wed Feb 16 00:02 - 00:05	(00:02)
bory	pts/17	dq42.internetdsl	Tue Feb 15 20:36 - 20:40	(00:03)
zbik	pts/17	atwork.zbik.org	Tue Feb 15 14:45 - 14:53	(00:08)
vrobelt	pts/17	r9.ists.pl	Mon Feb 14 03:58 - 04:28	(00:29)
vrobelt	pts/17	r9.ists.pl	Sun Feb 13 17:33 - 23:03	(05:29)
vrobelt	pts/17	r9.ists.pl	Sat Feb 12 19:38 - 19:43	(00:05)