Zarządzanie użytkownikami i grupami w systemie linux.

Użytkownicy i grupy mogą być w Linuksie lokalni i domenowi (oparte o LDAP).

Każdy użytkownik i każda grupa mają w systemie swój identyfikator: w przypadku użytkowników to UID (ang. User ID), a w przypadku grup — GID (ang. Group ID).

Każdy użytkownika musi należeć do co najmniej jednej grupy, tzw. grupy podstawowej.

Konto administratora systemu to w Linuksie konto użytkownika root, filozofia tego konta jest odmienna od tej w systemie Windows.

Dwie główne różnice:

Administratorem w Linuksie może być tylko użytkownik root. Można delegować uprawnienia administracyjne na wybranych użytkowników, pozostają oni nadal zwykłymi użytkownikami.

Administrator w Linuksie ma pełne prawa i nikt mu ich nie może odebrać. Jest to zdecydowanie odmienna filozofia od tej w Windows - tam zwykły użytkownik może całkowicie odebrać administratorowi dostęp do swoich plików (administrator może przejąć je na własność uporządkować uprawnienia, jednak zmienia się wtedy stan systemu i nie może to pozostać niezauważone).

Każde rozwiązanie ma swoje wady i zalety.

Katalogi domowe

Każdy użytkownik może mieć swój katalog domowy (profil).

Domyślnie profile tworzone są w katalogu /home i nazwą jest nazwa użytkownika, np. /home/jan, nie ma żadnych ograniczeń, żeby katalogi domowe były w dowolnej lokalizacji i miały dowolną nazwę. Wyjątkiem jest użytkownik root, który katalog domowy ma w folderze /root.

Zalecane jest, aby katalogi domowe znajdowały się na osobnej partycji, na której uruchomiony jest system quota, czyli mechanizm sterowania przydziałem miejsca na dysku dla poszczególnych użytkowników i grup.

Katalog domowy aktualnie zalogowanego użytkownika jest oznaczany przez znak tyldy (~), czyli np. jeśli zalogowany jest jan posiadający katalog domowy /home/jan, to wydanie polecenia cd ~ spowoduje przejście do właśnie tego katalogu.

Hasła

Hasła są newralgicznym elementem każdego systemu informatycznego, są chronione przed nieuprawnionym dostępem.

W Linuksie mamy dwa istotne fakty związane z hasłami:

Hasła są obecnie przechowywane w postaci funkcji skrótu MD5 lub SHA1. Własnością funkcji skrótu jest brak możliwości odzyskania z „szyfrogramu" oryginalnego tekstu (czyli w tym kontekście hasła), stąd też funkcje skrótu są także nazywane jednokierunkowymi.

Hasła są przechowywane w pliku /etc/shadow, który jest dostępny tylko dla użytkownika root.

Sam system nie wystarczy, jeśli użytkownicy poprzez swoją niefrasobliwość doprowadzą do wycieku hasła.

Podstawowe (zwykle oczywiste) zasad postępowania z hasłami.

Hasła powinno być odpowiednio skomplikowane i długie, tj. obecnie przyjmuje się długość co najmniej 8 znaków z różnych kategorii (małe, wielki litery, cyfry, kilka znaków specjalnych, interpunkcyjnych). System można skonfigurować tak, aby wymuszał odpowiednią złożoność hasła.

W przypadku konieczności zapisania hasła, odpowiednio je chronimy, np. portfel, sejf. Przyklejanie kartki pod klawiaturą, z racji powszechności stosowania nie jest dobrym pomysłem. Dobrym zwyczajem jest przechowywanie hasła w odpowiednio „zalakowanej" kopercie w sejfie, do którego dostęp mają osoby z dyrekcji lub zarządu. Bardzo niedobrym zwyczajem jest, aby hasło znała tylko jedna osoba, firma czy instytucja może mieć niepotrzebne komplikacje.

Jeśli hasło musi zostać przekazane osobom nieuprawnionym (czasami jest to konieczność, np. administrator jest niedostępny, a pilnie trzeba wykonać jakąś czynność administracyjną), hasło powinno zostać zmienione przez odpowiednią osobę najszybciej, kiedy to tylko możliwe.

Hasła trzeba co pewien czas zmieniać. Warto wykazać się rozsądkiem: jeśli sieć jest za dobrze skonfigurowanym firewallem z maskaradą, osoby w sieci darzą się dużym zaufaniem, nie ma potrzeby stosować bardzo restrykcyjnej polityki haseł - zmiana raz w roku powinna wystarczyć, złożoność haseł też nie musi być na bardzo wysokim poziomie. Jeśli mamy do czynienia z przypadkiem przeciwnym, gdzie sieć integruje np. 480 oddziałów w Europie i ludzie się nie znają, jest powszechnie stosowany zdalny dostęp, wtedy niezbędne jest zastosowanie bardziej restrykcyjnej polityki haseł.

Powłoka

Powłoka to program, który będzie domyślnie dostępny po uruchomieniu wiersza poleceń.
W szczególności, jeśli nie korzystamy z systemu graficznego, będzie to program wiersza poleceń dostępny po zalogowaniu. Najczęściej spotykaną powłoką jest bash, inne to np. tcsh, csh, sh.

Pliki poczty

W systemie domyślnie instalowane jest oprogramowanie do lokalnego dostarczania poczty.
Skrzynki pocztowe są wtedy najczęściej zlokalizowane z katalogu /var/mail, a w niektórych dystrybucjach w katalogu /var/spool/mail. W sytuacji, gdy użytkownik korzysta z programu pocztowego, w którym tworzy własne skrzynki pocztowe, są one wtedy zwykle umieszczane w określonym katalogu w profilu użytkownika, np. ~/Mail.

Pliki konfiguracyjne

Główne pliki konfiguracyjne w systemie Linux, które stanowią bazę użytkowników i grup to:

- /etc/passwd — zawiera podstawowe dane użytkowników,
- /etc/shadow — zawiera informacje o hasłach,
- /etc/group — zawiera informacje o grupach.

Format tych plików jest następujący. Każdy wiersz dotyczy pojedynczego użytkownika, hasła lub grupy, natomiast wiersze składają z kolumn rozdzielonych znakiem dwukropka.

Schemat jednego wiersza w pliku /etc/passwd:

- konto — nazwa użytkownika, zgodnie z konwencją bez wielkich liter,
- hasło — kiedyś było tu hasło, obecnie jest znak x,
- UID — ID użytkownika,
- GID — ID grupy podstawowej,
- opis — tzw. pole GECOS, informacje o koncie ustawiane przez polecenie chfn; zwykle w formacie listy przecinkowej zawierającej imię i nazwisko, nr pokoju, telefon służbowy, telefon domowy, inne,
- katalog — ścieżka do katalogu domowego,
- powłoka — domyślny program wiersza poleceń.

Przykładowy fragment pliku:

dhcpd:x:103:65534:DHCP server daemon:/var/lib/dhcp:/bin/false

ftp:x:40:49:FTP account:/srv/ftp:/bin/bash

games:x:12:100:Games account:/var/games:/bin/bash

gdm:x:50:105:Gnome Display Manager daemon:/var/lib/gdm:/bin/false

haldaemon:x:101:102:User for haldaemon:/var/run/hal:/bin/false

ldap:x:76:70:User for OpenLDAP:/var/lib/ldap:/bin/bash

ntp:x:74:103:NTP daemon:/var/lib/ntp:/bin/false

postfix:x:51:51:Postfix Daemon:/var/spool/postfix:/bin/false

root:x:0:0:root:/root:/bin/bash

wwwrun:x:30:8:WWW daemon apache:/var/lib/wwwrun:/bin/false

jan:x:1001:100:Jan Kowal,340,71333,71444,Brak:/home/jan:/bin/bash

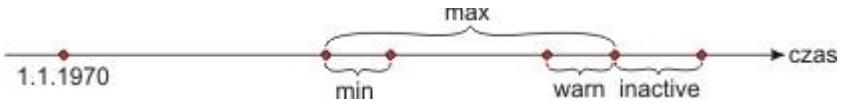
Co to za dziwne konta? Otóż są to konta usług. Są one utworzone ze względów bezpieczeństwa.

Na przykład DHCPD. Usługa DHCP do swojego działania potrzebuje bardzo niewielu danych.
Jeśli byśmy serwer DHCP uruchamiali na tożsamości użytkownika root, to w przypadku włamania ktoś ma potencjalnie dostęp do tego, do czego ma dostęp root, czyli do wszystkiego. W przypadku, gdy usługa działa na tożsamości użytkownika dhcpd, to po włamaniu ktoś przejmuje to co ma użytkownik dhcpd, czyli bardzo niewiele.

Schemat pojedynczego wiersza pliku /etc/shadow zawiera informacje o hasłach użytkowników:

- nazwa użytkownika,
- zakodowane hasło, zwykle poprzez funkcję SHA1 lub MD5,
- liczba dni od 1.1.1970 do daty ostatniej zmiany hasła,
- liczba dni od ostatniej zmiany hasła, podczas których nie można hasła zmienić (okres min),
- liczba dni od ostatniej zmiany hasła, po których zmiana hasła jest wymagana; inaczej: okres ważności hasła) (okres max),
- liczba dni przed wygaśnięciem hasła, podczas których użytkownik będzie informowany o konieczności zmiany hasła (okres warn),
- liczba dni licząc od ostatniego dnia, kiedy można było zmienić hasło, podczas których użytkownik może jeszcze zmienić hasło; inaczej: hasło straciło już ważność, czyli nie można się za jego pomocą zalogować, ale można za jego pomocą zmienić je na nowe hasło (okres inactive)
- po tych dniach konto zostanie zablokowane,
- po zablokowaniu niezbędny jest kontakt z administratorem systemu,
- liczba dni od 1.1.1970 do momentu (dnia), którym konto zostało/zostanie zablokowane, zarezerwowane.

Aby lepiej zrozumieć powyższe zależności, warto zerknąć na poniższą ilustrację.



Należy pamiętać, że zawartość pliku /etc/shadow jest czymś statycznym. To co się zmienia w całym układzie to upływający czas i to, jak system zareaguje zależy właśnie od bieżącej chwili, czyli od tego, gdzie chwila bieżąca „wpadnie” na powyższym wykresie.

Po co jest wartość min. Jak w jednej z firm pracownicy radzą sobie z koniecznością cyklicznej zmiany hasła. W każdej firmie jest zwykle tak, że hasło trzeba zmieniać co pewien okres czasu (np. 3 miesiące) i trzeba się przy tym stosować do pewnych reguł. Jedną z tych reguł jest zwykle to, że hasło musi być różne od 6 ostatnich haseł. Jak sobie radzą pracownicy? Zmieniają hasło 7 razy, przy czym ostatnia zmiana jest na hasło sprzed zmiany, gdyż system pomięta tylko 6 ostatnich haseł. Oczywiście reguła historii haseł traci sens. Chyba, że... Chyba, że określimy okres min, wtedy już powyższy numer nie przejdzie.

Schemat pojedynczego wiersza pliku /etc/group, który zawiera informacje o grupach:

- nazwa grupy
- hasło grupy
- GID
- lista użytkowników w grupie (np. anna, jan)

Użytkownicy z danej grupie określani są poprzez nazwy, a nie identyfikatory.
W niektórych dystrybucjach można się spotkać z rozwiązaniem, w którym zamiast pliku group są dwa pliki group i gshadow. Ich rola jest wtedy analogiczna do roli plików passwd i shadow.

Narzędzia do obsługi kont - polecenia konsolowe:

useradd

Polecenie to służy do dodania użytkownika do systemu, także do zmiany domyślnych parametrów, według których nowi użytkownicy będą utworzeni.

Składnie użycia tego polecenia. Pierwsza składnia pozwala na dodanie użytkownika, druga wyświetla wartości domyślnych parametrów, a trzecia pozwala na ustawienie tych parametrów domyślnych, które to parametry są przechowywane w pliku /etc/default/useradd.

useradd [-c komentarz] [-d katalog] [-m] [-g gid] [-G grupa,...]

[-e yyyy-mm-dd] [-f liczba] [-s powłoka] [-u uid] nazwauzytkownika

useradd --show-defaults

useradd --save-defaults [-d katalog] [-f liczba] [-g gid] [-G grupa,...] [-s powłoka]

Znaczenie parametrów.

- -c tekst — wprowadza opis użytkownika, czyli wartość pola GECOS w pliku passwd.
- -d katalog — określa katalog domowy na potencjalnie inny niż domyślny. Zwracam uwagę, że jest to TYLKO określenie wpisu w pliku passwd, tzn., że jeśli chcemy, aby ten katalog został utworzony należy użyć dodatkowej opcji -m.
- -m — określa, czy katalog domowy ma zostać utworzony (zgodnie z lokalizację określoną parametrach domyślnych lub poprzez opcję -d). Przy tworzeniu katalogu kopiowana jest zawartość katalogu /etc/shel, chociaż można poprzez dodatkową opcję (-k) określić inny katalog „źródłowy”, dzięki czemu możemy mieć kilka schematów początkowej konfiguracji dla różnego typu użytkowników.
- -f liczba — określa liczbę dni „nieaktywności” (wartość pola inactive z pliku shadow)
- -g gid — określa grupę podstawową dla użytkownika
- -G grupa … — określa grupy dodatkowe
- -e yyyy-mm-dd — określa, kiedy konto zostanie/zostało zablokowane; ważne jest wprowadzenie daty w przedstawionym formacie
- -s powłoka — określa powłokę
- -u uid — pozwala jawnie podać identyfikator użytkownika

userdel

Polecenie do usunięcia użytkownika. Ma tylko jeden opcjonalny parametr -r, który określa, czy usunąć także katalog domowy wraz z zawartością. Składnia:

userdel [-r] nazwauzytkownika

usermod

Polecenie to służy do modyfikacji konto użytkownika. Składnia:

usermod [-c komentarz] [-d katalog [-m]] [-f liczba] [-g gid] [-G grupa...]

[-s powłoka] [-u uid] [-l login] nazwa_uzytkownika

Chociaż większość opcji jest taka sama jak w przypadku polecenie useradd, warto zwrócić uwagę na dwie a nich:

- -d katalog [-m] — sama opcja -d zmienia wpis w pliku passwd, jednak w połączniu w opcją -m tworzony jest nowy katalog domowy i przenoszona jest zawartość poprzedniego katalogu domowego do nowej lokalizacji (poprzednia lokalizacja jest usuwana)
- -l login — określa nową nazwę użytkownika; warto zauważyć, że nie jest to tylko wymiana nazwy w pliku passwd, np. w pliku grup nazwa również aktualizowana

groupadd

Polecenie do utworzenia grupy. Składnia:

groupadd [-g gid] nazwa

Opcja -g pozwala na jawne wskazanie identyfikatora nowotworzonej grupy.

groupdel

Polecenie do usuwania grupy. Składnia i zarazem jedyne użycie:

groupdel nazwagrupy

groupmod

Polecenie do modyfikacji konta grupy. Składnia:

groupadd [-g gid] [-n nowanazwa] nazwa

Podanie opcji -g pozwala na zmianę identyfikatora, a podanie opcji -n pozwala na zmianę nazwy grupy.

passwd

Polecenie ma dużo różnych zastosowań.

Ustawienie hasła dla konta użytkownika

- Składnia polecenia:
- passwd [nazwauzytkownika]
- Dla zwykłego użytkownika dostępna jest składnia bez parametru, która pozwala na zmianę hasła aktualnie zalogowane użytkownika. Użytkownik root może dodatkowo podać jako parametr użytkownika, którego chce ustawić hasła. Jeszcze jedną różnicą jest to, że zwykły użytkownik musi

w procesie zmiany hasła wykazać się znajomością dotychczasowego hasła, natomiast użytkownik root po prostu ustawia hasło bez konieczności podawania dotychczasowego.

Ustawienie informacji o użytkowniku

Składnia polecenia:

- passwd -f [nazwaużytkownika]
- Polecenie pozwala na ustawienie wartości pola GECOS w pliku passwd. Dostępność opcji nazwauzytkownika jest analogiczna jak w poprzednim użyciu. Format danych jest z kolei taki jak w pliku /etc/passwd. Ten sam efekt co powyższym poleceniem można także uzyskać poleceniem chfn

Sterowanie dostępnością konta

Składnia polecenia:

- passwd {-l | -u | -d | -e} nazwauzytkownika

Znaczenie opcji:

- -l — blokowanie konta,
- -u — odblokowanie konta,
- -d — usunięcie hasła,
- -e — ustawienie konta tak, żeby użytkownik musiał zmienić hasło przy kolejnym logowaniu.

Ta składnia jest dostępna tylko dla administratora. Polecenie operuje na pliku shadow i wykonanie polecenia z każdą opcją powoduje pewną modyfikację tego pliku.

Konfiguracja parametrów hasła

Składnia polecenia:

- passwd [-n min] [-x max] [-w warn] [-i inactive] nazwauzytkownika
- Znaczenie opcji jest takie samo, jak to umówione w kontekście pliku shadow. Proszę zwrócić uwagę, że te opcje dokładnie odpowiadają poszczególnym kolumnom pliku shadow, stąd zrozumienie struktury tego pliku sprawia, że czytelnik bez problemu poradzi sobie z użyciem tego polecenia.

Pobieranie informacji o kontach

Składnia polecenia:

- passwd -S [-a | nazwauzytkownika]

Znaczenie opcji

- nazwauzytkownika — wyświetlenie informacji o koncie użytkownika nazwauzytkownika
- -a — wyświetlenie informacji o wszystkich kontach

Dla zwykłego użytkownika dostępne jest polecenie tylko z opcją -S, które wydrukuje informacje o aktualnie zalogowanym użytkowniku, natomiast pozostałe opcje są, oczywiście, dostępne dla użytkownika root.

Ponieważ wynik działania tego polecenia może nie być oczywisty, krótko go umówimy. Przykładowy wydruk:

$ passwd -S gucio

gucio PS 10/19/2008 0 99999 7 -1

Pierwsza kolumna to oczywiście nazwa konta. Kolejna to stan konta. Mamy następujące wartości:

- PS — użytkownik ma hasło i może się logować
- LK — konto jest zablokowane
- NP — użytkownik nie ma hasła

Dalsze kolumny ponownie odpowiadają kolumnom z pliku shadow: data ostatniej zmiany hasła i okresy min, max, warn oraz inactive.

gpasswd

Polecenie pozwalające utworzyć lub usunąć hasło dla konta grupy. Składnia:

gpasswd [-r] grupa

Podanie opcji -r spowoduje usunięcie hasła, wywołanie polecenia bez tej opcji – ustawienie hasła. Na pytanie do czego może służyć hasło dla grupy odpowiemy sobie przy okazji systemu plików, ponieważ, jak się można domyślać, będzie ono pozwalało na sterowanie dostępem do zasobów.

su

Polecenie to służy do zmiany identyfikatora użytkownika i grupy w konsoli, czyli inaczej mówiąc umożliwia „przelogowanie” w konsoli tekstowej. Tak naprawdę w bieżącej konsoli uruchamiana jest nowa konsola, w której funkcjonujemy na nowej tożsamości. Po wylogowaniu wracamy do poprzedniej konsoli, w tym także tożsamości. Składnia:

su [-c polecenie] [-] [uzytkownik]

Znaczenie opcji:

- - — sprawia, że powłoka będzie powłoką logowania, czyli np. zostaną przywrócone zmienne środowisko; inaczej mówiąc konsola będzie wyglądać tak, jakby użytkownik się w niej zalogował, a nie wykonał su,
- -c polecenie — zamiast przełączyć się na użytkownika, wykonamy jako ten użytkownik polecenie, dokładniej: zostanie uruchomiona nowa konsola z nową tożsamością, w tej konsoli zostanie wykonane polecenie, po czym po wykonaniu polecenia, ta nowa konsola zostanie zamknięta i powrócimy do poprzedniej konsoli (tożsamości).

Jak zauważamy, parametr użytkownik jest opcjonalny. Otóż niepodanie tego parametru oznacza próbą przelogowania się na użytkownika root. Warto też odnotować, że użytkownik root może się „przełączać” na dowolnego użytkownika bez konieczności podawania hasła (pamiętamy, że root może wszystko).

sudo

Jest to polecenie, za pomocą którego możemy delegować uprawnienia administracyjne na innych użytkowników. Polecenie to jednak jest ogólniejsze: służy do tego, że wykonywać polecenia na innej tożsamości (część funkcjonalności tego polecenia oferuje polecenie su). Polecenie to ma kilka składni:

sudo [-u użytkownik] polecenie

sudoedit [-u użytkownik] ścieżka-do-pliku

sudo { -v | -k | -l }

Pierwsza składnia służy do uruchomienia danego polecenia na innej tożsamości. Ponownie, niepodanie użytkownika oznacza próbę wykonania polecenia na tożsamości użytkownika root.

Druga składnia służy do edycji plików na innej tożsamości, przy czym scenariusz jest następujący:

Plik do edycji jest kopiowany (ewentualnie tworzony, jeśli nie istnieje) do katalogu tymczasowego (zwykle /var/tmp), przy czym jego właścicielem i grupą jest tożsamość, na której ma być dokonana edycja.

Następnie uruchamiany jest edytor do edycji tego tymczasowego pliku (brany jest edytor ze zmiennych EDITOR lub VISUAL, a jeśli zmienne nie są ustawione, z listy edytorów określonych w pliku konfiguracyjnym sudoers).

Po zakończeniu edycji oryginalny plik jest zastępowany tym zmodyfikowanym. Jeśli ta zamiana jest niemożliwa, zmodyfikowany plik pozostaje w katalogu tymczasowym, a użytkownik dostaje stosowne ostrzeżenie.

Czym to się tak naprawdę różni (poza różnymi szczegółami), od odpalenia polecenia sudo z poleceniem edycji? W pliku konfiguracyjnym określamy co, kto i na czym może uruchomić, w szczególności, jeśli ktoś chciałby edytować za pomocą edytora vi musielibyśmy dać mu takie uprawnienie. A co, jeśli ktoś panicznie boi się edytora vi? No właśnie, żeby wszystkim dogodzić musielibyśmy każdemu dać prawa do każdego edytora. Korzystając z sudoedit jest to prostsze, ponieważ w konfiguracji dajemy prawa do sudoedit, a to który edytor będzie uruchomiony zależy od zmiennej EDITOR, co już może sobie użytkownik ustawić wedle uznania.

Przypuśćmy, że jesteśmy zalogowani jako jan. W momencie, gdy chcemy wykonać useradd jako root, zostaniemy poproszeni o podanie hasła. Jednak, gdy zaraz potem wykonamy useradd ponownie, o podanie hasła nie zostaniemy poproszeni. Stanie się tak, ponieważ po wykonaniu sudo po raz pierwszy, system zapamięta, że miało to miejsce i przez pewien czas (domyślnie 5 minut) nie będzie ponownie żądał hasła. Co więcej, jeśli w okresie tych 5 minut wykonamy useradd ponownie, te 5 minut będą się liczyć na nowo (od momentu tego drugiego wykonania useradd). Generalnie kolejnym poleceniem nie musi być useradd - istotnym jest, żeby kolejne polecenia było wykonane na tożsamości użytkownika root.

W trzeciej mamy następujące znaczenie opcji:

- -v - powoduje przedłużenie zapamiętania hasła dla użytkownika root (podbicie timestampu), czyli jakby udaje wykonanie polecenia sudo na koncie root (dotyczy tylko konta root),
- -k - powoduje natychmiastowe „zapomnienie” hasła (bardzo przydatne, gdy musimy okazjonalnie wykonać polecenie na cudzym komputerze, po czym chcemy przejść do innego komputera -

niewykonanie sudo -k może zachęcić użytkownika wykonania poleceń na tożsamości użytkownika root (dotyczy tylko konta root.

- -l - podanie tej spowoduje wyświetlenie uprawnień danego użytkownika (dostępne po podaniu hasła konta użytkownika root).

Plikiem konfiguracyjnym polecenia sudo jest plik /etc/sudoers. Plik ten ma bardzo rozbudowaną składnię i spore możliwości. Poniżej kilka przykładów przybliżających, co można tam wstawić:

Aliasy. Przykłady:

User_Alias WEBMASTERS = jan, monika

Host_Alias SERVERS = dc, www, ftp, mail, ns

Host_Alias LAB31 = 10.2.31.0/24

Cmnd_Alias KILL = /usr/bin/kill

Cmnd_Alias HALT = /usr/sbin/halt

Cmnd_Alias PRINTING = /usr/sbin/lpc, /usr/bin/lprm

Cmnd_Alias USERCMNDS = /usr/sbin/useradd, /usr/sbin/userdel, (itd.)

Wartości domyślne. Trzy poziomy określania:

- Defaults
- Defaults:użytkownik
- Defaluts@Host

Przykłady:

Defaults syslog=auth

Defaults passprompt="Witaj %u - podaj swoje hasło:"

Defaults:jan timestamp_timeout=-1

Defaults:enemy timestamp_timeout=0

Defaults targetpw (bardzo ważna opcja)

Nadawanie uprawnień. Ogólna składnia:

kto komputer=(jakokto) polecenia

Przykłady

root ALL=(ALL) ALL

jan localhost=(root) NOPASSWD: KILL

jan cassiopeia=(root) NOPASSWD: /usr/sbin/useradd

jan cassiopeia=(root) PASSWD: /usr/sbin/userdel

jan cassiopeia=(ala,basia) NOPASSWD: /bin/ls

Więcej można się dowiedzieć w manualu: man sudoers.

sg newgrp

Służy do zmiany identikatora bieżącego identyfikatora grupy (GID). Składnia:

sg [-l | -c command] group

Działanie i znaczenie opcji są takie samo jak w przypadku polecenia su, natomiast należy zwrócić uwagę na jakie grupy możemy się „przełączać". Pamiętamy, że użytkownik ma grupę podstawową i grupy dodatkowe. Przełączać się możemy na wszystkie grupy dodatkowe, do których użytkownik należy (bez podawania hasła). Dodatkowo, jeśli jakaś grupa ma ustawione hasło i podczas przełączania to hasło podamy, to również będziemy mogli zmienić tożsamość grupy.

users

Wyświetla listę zalogowanych użytkowników na komputerze. Składnia i użycie zarazem:

users

groups

Wyświetla listę grup, do których należy użytkownik. Składnia:

groups [nazwauzytkownika]

Podanie nazwy użytkownika spowoduje wyświetlenie listy group konta użytkownika, natomiast niepodanie nazwy użytkownika spowoduje wyświetlenie list grup aktualnie zalogowanego użytkownika. Przykład (zalogowany jest jan):

```
1  $ groups
2  users
3  $ groups jan
4  jan : users
5  $ sg root
6  Hasło:
7  $ groups jan
8  jan : users
9  $ groups
```

10 root users

Polecenie robi to samo co id -Gn.

id

Wyświetla informacje o kontach użytkowników. Składnia:

id [-u | -g | -G] [-n] [nazwa]

Podanie nazwy użytkownika spowoduje wyświetlenie informacji o koncie użytkownika, niepodanie nazwy użytkownika — informacje będą dotyczyły zalogowanego użytkownika.

Znaczenie opcji:

- -u — wyświetla ID/nazwę użytkownika
- -g — wyświetla ID/nazwę grupy podstawowej użytkownika
- -G — wyświetla ID/nazwy wszystkich grup, do których należy użytkownik (łącznie z grupą podstawową)
- -n — decyduje, czy będą wyświetlane nazwy czy identyfikatory

last

Wyświetla historię logowań. Składnia:

last [-liczba | -num liczba] [-a] [uzytkownik]

Znaczenie opcji:

- -liczba, -num liczba — liczba wyświetlanych wierszy
- -a — nazwa hosta będzie ostatnią kolumną
- podanie nazwy użytkownika filtruje wpisy i zostawia tylko te dotyczące podanego użytkownika

Pliki, w których dane są przechowywane to /var/log/wtmp i /var/log/btmp, przy czym, żeby rejestrowanie historii logowań miało miejsce, te pliki muszą istnieć (w razie potrzeby należy je utworzyć poleceniem touch).

finger

Program do monitorowania aktywności użytkowników w systemie. Składnia:

finger [-l] [użytkownik]

Podanie nazwy użytkownika spowoduje wyświetlenie informacji o podanym użytkowniku, brak nazwy użytkownika — wyświetlenie informacji o wszystkich zalogowanych użytkownikach. Opcja -l określa rodzaj wydruku (pełny lub skrócony) i zmienia zawartość wydruku.

who

Polecenie podobne do polecenia finger, ma kilka różnych składni:

who [-HT]

who -q

who -r

Pierwsza składnia drukuje listę aktualnie zalogowanych użytkowników. Znaczenie opcji:

- -H — dołącza nagłówek do listy
- -T — dołącza do listy informację, czy do użytkownika można wysyłać komunikaty:
  - + : można
  - - : nie można
  - ? : nie znaleziono terminala

Druga składnia wyświetla informacje o aktualnie zalogowanych użytkownikach w trochę innej formie:

- w pierwszym wierszu drukuję listę zalogowanych użytkowników (oddzielonych znakiem spacji)
- w drugim wierszu podaje liczbę zalogowanych użytkowników

Trzecia składnia podaje bieżący poziom pracy.