Technologia i rozwiązania

Najlepsze narzędzia w systemie Linux

Wykorzystaj ponad 70 receptur i programuj szybko i skutecznie

James Kent Lewis

[PACKT] open source
PUBLISHING
community experience distilled

Tytuł oryginału: Linux Utilities Cookbook

Tłumaczenie: Krzysztof Rychlicki-Kicior

ISBN: 978-83-246-8980-4

First published in the English language under the title: „Linux Utilities Cookbook”.

Wydawnictwo HELION
ul. Kościuszki 1c, 44-100 GLIWICE
tel. 32 231 22 19, 32 230 98 63
e-mail: helion@helion.pl
WWW: http://helion.pl (księgarnia internetowa, katalog książek)

Drogi Czytelniku!
Jeżeli chcesz ocenić tę książkę, zajrzyj pod adres
http://helion.pl/user/opinie/nanali
Możesz tam wpisać swoje uwagi, spostrzeżenia, recenzję.

Printed in Poland.

Spis treści

5

ROZDZIAŁ

Uprawnienia, dostęp i bezpieczeństwo

W tym rozdziale zajmiemy się:

- tworzeniem kont użytkowników i zarządzaniem nimi — `useradd`,
- obsługą haseł,
- obsługą uprawnień plików,
- konfigurowaniem zapory sieciowej i ustawień rutera,
- obsługą Secure Linux — SELinux,
- korzystaniem z sudo w celu zabezpieczenia systemu,
- katalogiem */tmp*.

Wprowadzenie

Ten rozdział posłuży nam jako krótkie omówienie uprawnień plików w systemie Linux. Dowiesz się także, jak zabezpieczenia są obsługiwane przez system haseł. Poza tym pokazujemy, jak w bezpieczny sposób skonfigurować zaporę sieciową i ruter, a także wspominamy o SELinux i poleceniu `sudo`.

Tworzenie kont użytkowników i zarządzanie nimi — useradd

W ramach tej porady dodamy konto użytkownika, korzystając z programu `useradd`.

Przygotuj się

Poniższe polecenia nie powinny spowodować problemów w Twoim systemie, niemniej jednak będzie Ci potrzebny dostęp do konta administratora (*roota*).

> W większości dystrybucji systemu Linux są dostępne dwie wersje tego polecenia — `useradd` i `adduser`. Nie zawsze wykonują one te same czynności, dlatego sprawdź podręcznik `man` (i/lub polecenie `file`), aby się upewnić, że wykonujesz odpowiednie polecenie. W systemie Fedora `adduser` stanowi dowiązanie symboliczne do polecenia `useradd`, a zatem są one sobie równoważne.

Jak to zrobić

Teraz wykonamy polecenie `useradd`, aby dodać użytkownika, a następnie polecenie `passwd`, aby określić hasło. Więcej informacji na temat polecenia `passwd` znajdziesz w kolejnych poradach:

1. Najpierw utworzymy kopię zapasową pliku */etc/passwd*. Wykonaj poniższe polecenie:

   ```
   cp /etc/passwd /tmp/passwd.orig
   ```

2. Utwórz użytkownika o nazwie *test1*:

   ```
   useradd test1
   ```

3. Powinieneś powrócić od razu do wiersza poleceń. Następnie wykonaj polecenie:

   ```
   su - test1
   ```

4. Znak zachęty powinien ulec zmianie. Wykonaj polecenie `whoami` — zostanie wyświetlony tekst *test1*. Zawsze upewniaj się w ten sposób, że jesteś zalogowany na właściwe konto. Teraz zmienimy hasło:

   ```
   run passwd
   ```

5. Zostanie wyświetlony komunikat o zmianie hasła, np. Zmiana hasła dla użytkownika *test1*, a także pojawi się komunikat z prośbą o wprowadzenie aktualnego hasła. Co to oznacza? O jakie hasło prosi system?

6. Tak naprawdę nie znam odpowiedzi na to pytanie, a strony podręcznika `man` tej kwestii nie rozwiązują. Ten krok jest zawsze pomijany, co jest moim zdaniem dość dziwne. Istnieją sposoby na wykonanie tego kroku dzięki wykorzystaniu funkcji crypto i innych skomplikowanych procedur. Ja na co dzień stosuję nieco inne podejście:

7. Wciśnij kombinację klawiszy *Ctrl+C*, aby opuścić polecenie `passwd`, a następnie wykonaj polecenie `exit`, aby powrócić do konta administratora. Otwórz do edycji plik */etc/passwd* i przejdź do dolnego wiersza. W poniższym wierszu wartość liczbowa może być inna, ale poza tym nie powinno być problemów ze znalezieniem wiersza podobnego do następującego:

   ```
   test1:x:1003:1003::/home/test1:/bin/bash
   ```

8. Usuń x, dzięki czemu wiersz przyjmie następującą postać:

   ```
   test1::1003:1003::/home/test1:/bin/bash
   ```

9. Zapisz plik i wyjdź z edytora. Jeśli otrzymasz błąd uprawnień, upewnij się, że jesteś na koncie administratora.
10. Wykonaj polecenie `su`, aby ponownie przejść na konto użytkownika *test1*:

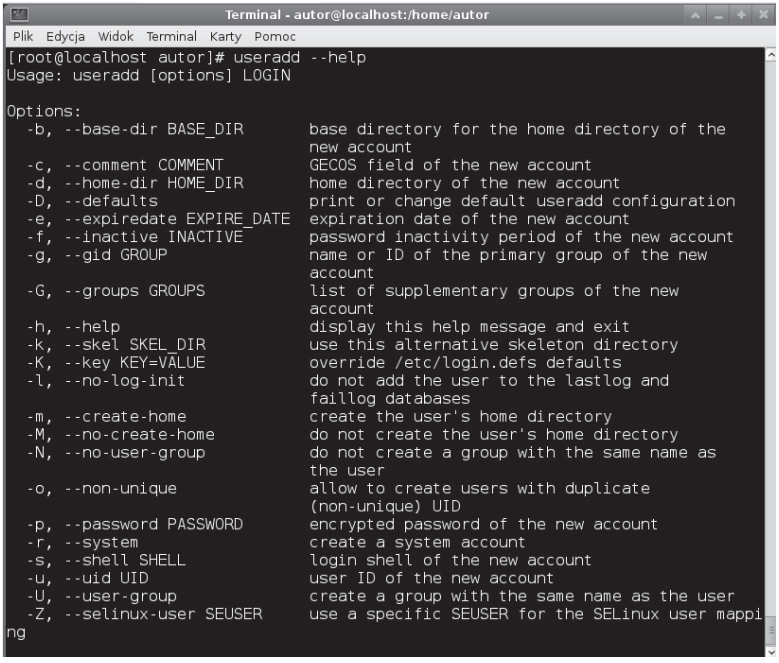
    ```
    su - test1
    ```

11. Wykonaj polecenie `passwd`.
12. Teraz jest znacznie lepiej — system nie prosi nas o hasło. Można teraz utworzyć nowe hasło, ponieważ posiadanie niezabezpieczonych kont w systemie nie jest zbyt mądre. Jeśli chcesz zachować to konto na przyszłość, radzę zapisać to hasło w bezpiecznym miejscu, a najlepiej przechować w jakimś zaszyfrowanym pliku.
13. Po dwukrotnym wprowadzeniu nowego hasła powinieneś otrzymać komunikat podobny do poniższego:

    ```
    passwd: all authentication tokens updated successfully
    ```

Teraz dysponujemy nowym kontem użytkownika. Zwróć uwagę, że użytkownik może wykonywać większość operacji z poziomu terminala, o ile tylko ma odpowiednie uprawnienia. W zależności od dystrybucji systemu Linux zwykły użytkownik może nie mieć dostępu do wszystkich zasobów (np. systemu dźwięku).

Oto zrzut ekranu z polecenia `useradd --help`, które wykonałem w systemie Fedora 17:



Zobacz również

Polecenie `useradd` może być używane do różnych innych celów. Można m.in. modyfikować sposób działania istniejących kont lub zdefiniować termin jego wygaśnięcia. Istnieje możliwość przyznania szerokich uprawnień — niemal tak dużych jak uprawnienia administratora. Zajrzyj na stronę podręcznika `man` lub wywołaj polecenie useradd z parametrem `--help` w celu uzyskania bardziej szczegółowych informacji.

Obsługa haseł

O poleceniu `passwd` wspomniałem w poprzedniej poradzie. Używa się go do zmiany tokenów uwierzytelniania użytkownika. Do wykonania poniższych poleceń będzie potrzebny dostęp do konta administratora. Skorzystamy także z konta *test1*, utworzonego w poprzedniej poradzie.

Jak to zrobić

1. Przejdź do konta użytkownika *test1* z poziomu konta zwykłego użytkownika:

   ```
   su - test1
   ```

2. Wprowadź hasło. Operacja powinna przebiec bez problemów.
3. Teraz zablokujemy to konto. Powróć do konta administratora i wykonaj polecenie:

   ```
   passwd -l test1
   ```

4. Z poziomu konta zwykłego użytkownika wykonaj ponownie polecenie `su - test1`. Operacja powinna zakończyć się niepowodzeniem.
5. Powróć do konta administratora i odblokuj konto, korzystając z polecenia `passwd -u test1`. Zaloguj się ponownie i sprawdź, czy wszystko działa.
6. Teraz wygasimy konto *test1*. W ten sposób zmusimy użytkownika do utworzenia nowego hasła. Wykonaj poniższe polecenie jako *root*:

   ```
   passwd -e test1
   ```

7. Z poziomu zwykłego użytkownika zaloguj się ponownie na konto `test1`, korzystając z polecenia `su - test1`. Wprowadź swoje hasło.
8. Zostaniesz poproszony o utworzenie nowego hasła. Bądź uważny — najpierw musisz wprowadzić stare (aktualne) hasło, a następnie dwa razy nowe.
9. Warto zapamiętać, że hasło możemy także usunąć, wykonując polecenie `passwd -d test1`. Jest to znacznie prostsze niż bezpośrednia modyfikacja pliku */etc/passwd*, wykonana w poprzedniej poradzie.

Zobacz również

Modyfikacja konta użytkownika obejmuje wiele różnych aspektów. Można do nich zaliczyć czas pozostały do wygaśnięcia konta użytkownika, a także moment, w którym należy rozpocząć informowanie użytkownika o zmianie hasła. Więcej informacji znajdziesz w podręczniku `man`.

> **Co nieco o hasłach**
>
> Dawno temu użytkownicy wybierali proste hasła i nigdy ich nie zmieniali. Nie musieliśmy co chwilę ich zmieniać i mogliśmy korzystać z nich wszędzie, dlatego nie było potrzeby zapisywania haseł. Niestety, ta sytuacja uległa zmianie. Obecnie hasła muszą zawierać kombinacje liter, cyfr, a nawet znaków specjalnych. Co więcej, muszą one być dość długie, ponieważ zasady tworzenia haseł w jednym systemie mogą nie obowiązywać w innym. W związku z tym sugeruję stosować różne hasła dla różnych kont i zapisywać je w bezpiecznym miejscu. Prawdopodobnie będzie konieczna także regularna zmiana tych haseł.

Obsługa uprawnień plików

Linux jest systemem umożliwiającym obsługę wielu użytkowników, dlatego każdy plik ma określone uprawnienia dostępu i przypisanego właściciela. W ten sposób możemy ustrzec się przed niepowołanym dostępem do zasobów, do których dany użytkownik nie ma uprawnień (niezależnie od tego, czy chce zrobić to przypadkiem, czy też celowo). Administrator (*root*) ma zwykle dostęp do wszystkich plików w systemie operacyjnym.

Przygotuj się

Najpierw omówimy podstawowe uprawnienia dostępu do pliku. W tym przykładzie zakładam, że maska pliku jest ustawiona na wartość `0022`. Wykonaj polecenie `umask`, aby sprawdzić tę informację.

Przeanalizujmy efekt działania polecenia `ls -la` w katalogu, w którym znajduje się mój skrypt do wykonania kopii zapasowej, czyli *b*:

```
-rwxr-xr-x. 1 gosc1 root 559 Mar 28 12:43 b
```

Analizując od lewej strony, najpierw dowiadujemy się, z jakim rodzajem pliku mamy do czynienia. Obecność znaku `-` informuje, że jest to zwykły plik. Litera `d` obecna w tym miejscu oznaczałaby katalog, a `l` — łącze (ang. *link*). Kolejne trzy zbiory trzyliterowe określają uprawnienia dostępu do pliku, które można podawać albo w sposób symboliczny, albo liczbowy. Skorzystamy z trybu liczbowego (ósemkowego).

Pierwsza trójka, `rwx`, określa ustawienia dostępu dla właściciela (`gosc1`). Kolejne trzy znaki (`r-x`) stanowią ustawienia dostępu dla grupy (`root`). Ostatni zbiór to ustawienia dla pozostałych użytkowników. Litera `r` oznacza, że plik można odczytać, `w` — zapisać, a `x` — wykonać.

Polecenie `chmod` przyjmuje ciągi złożone z od jednej do czterech cyfr ósemkowych. Jeśli pierwszej cyfry nie ma, przyjmuje się, że na początku jest umieszczane wiodące zero. Pierwsza cyfra określa ID użytkownika, ID grupy lub tzw. lepki bit (ang. *sticky bit*). Druga cyfra określa uprawnienia użytkownika, a trzecia — uprawnienia dla wszystkich.

Teraz zmienimy uprawnienia pliku tymczasowego i zobaczymy, jaki będzie efekt.

Jak to zrobić

1. Przejdź do katalogu */tmp*:

   ```
   cd /tmp
   ```

2. Jeśli plik *f1* istnieje, usuń go:

   ```
   rm f1
   ```

3. Korzystając z konta gościa (w moim przypadku jest to konto *autor*), utwórz plik tymczasowy:

   ```
   ls > f1
   ```

4. Wykonaj poniższe polecenie:

   ```
   ls -al f1
   ```

5. Powinien zostać wyświetlony tekst podobny do poniższego:

   ```
   -rw-rw-r--. 1 autor autor 131 Mar 29 10:35 f1
   ```

6. Przedstawione uprawnienia stanowią uprawnienia domyślne przydzielane przez polecenie `umask`. W ten sposób stwierdzamy, że właściciel i grupa mają uprawnienia do odczytu i zapisu, a inni mogą jedynie odczytać plik.

7. W jaki sposób dokonamy zmiany? Korzystając z polecenia `chmod`. Załóżmy, że nasz plik jest skryptem i chcielibyśmy uczynić go wykonywalnym. Wykonaj poniższe polecenie:

   ```
   chmod 775 f1
   ```

8. Wykonaj polecenie `ls -la f1`. Efekt powinien być podobny do poniższego:

   ```
   -rwxrwxr-x. 1 autor autor 131 Mar 29 10:35 f1
   ```

9. Pojawiające się w trzech miejscach litery `x` oznaczają, że każdy użytkownik może wykonać ten plik. Wykonajmy jeszcze dwa polecenia: `chmod 000 f1`, a następnie `ls -la f1`. Zostanie przedstawiony następujący efekt:

   ```
   ----------. 1 autor autor 131 Mar 29 10:35 f1
   ```

No proszę! Czy to oznacza, że z tym plikiem nie da się już nic zrobić? Na szczęście nie — właściciel wciąż może zmienić uprawnienia. A skoro już o tym mowa, to aby zmienić właściciela pliku, należy skorzystać z polecenia chown. Zazwyczaj jest ono wykonywane z poziomu konta administratora.

Zobacz również

W tej poradzie nie omawiałem bitów `setuid`, `setgid` i `sticky`. Zerknij do podręcznika `man` na temat polecenia `chmod`, aby dowiedzieć się więcej na ich temat. Bit ograniczonego prawa usunięcia (ang. *restricted deletion bit*) jest omówiony w poradzie Katalog */tmp*.

Konfigurowanie zapory sieciowej i ustawień rutera

Zapora sieciowa jest używana do zapobiegnięcia nieautoryzowanemu dostępowi do komputera — przy jednoczesnym dopuszczeniu autoryzowanego ruchu. Polecenie iptables jest wykorzystywane do konfigurowania i przeglądania tabel reguł IPv4 jądra. Szczegółowe omówienie tego narzędzia wykracza poza ramy tej książki, dlatego w tym miejscu omówimy sobie pokrótce jedynie podstawy tego narzędzia.

Polecenie `iptables` korzysta z jednej (lub kilku) tabeli. Każda tabela składa się z pewnej liczby gotowych łańcuchów, ale może także zawierać łańcuchy stworzone przez użytkownika. Łańcuch stanowi zbiór reguł, a każda reguła decyduje o tym, co zrobić z pakietem, który pasuje do jej ustawień. Takie dopasowanie nazywamy celem (ang. *target*).

Jeśli pakiet nie pasuje do danej reguły, zostaje on poddany sprawdzeniu przez następną regułę z łańcucha. Jeśli jednak pakiet pasuje do reguły, może zostać podjęta wobec niego jedna z akcji:

- akceptuj (ang. *ACCEPT*) — pakiet jest przepuszczany,
- odrzuć (ang. *DROP*) — pakiet jest odrzucany,
- zakolejkuj (ang. *QUEUE*) — pakiet jest dodawany do przestrzeni użytkownika,
- powróć (ang. *RETURN*) — dany łańcuch jest przerywany; kolejna wywołana reguła pochodzi z łańcucha wywołującego.

Jak to zrobić

W tym miejscu przedstawimy zaledwie kilka poleceń typu `iptables`. Nie wykonuj ich w swoim systemie — instrukcje te stanowią jedynie przykład użycia:

1. Aby usunąć wszystkie istniejące reguły, wykonaj polecenie:

```
iptables -F
```

2. Aby zablokować konkretny adres IP, wykonaj następujące polecenie:

```
iptables -A INPUT -s 192.168.1.115 -j DROP
```

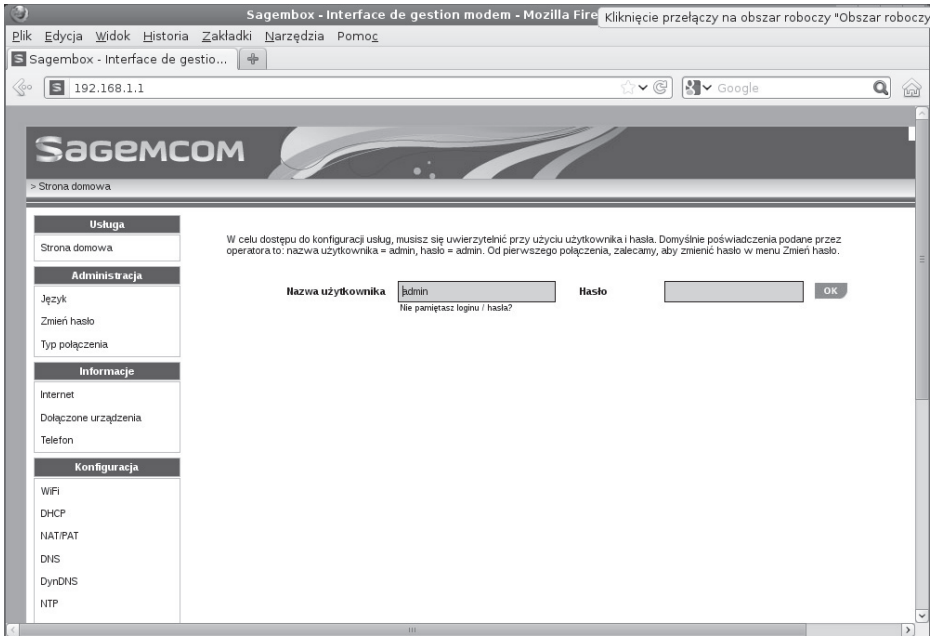
3. Aby zezwolić na dostęp do pętli zwrotnej, skorzystaj z poniższego polecenia:

```
iptables -A INPUT -i lo -j ACCEPT
iptables -A OUTPUT -o lo -j ACCEPT
```

Teraz zajmiemy się ruterami. Większość z nich zawiera wbudowane zapory sieciowe, które można skonfigurować z poziomu przeglądarki internetowej. Choć taki sposób dostępu nie zastąpi polecenia `iptables`, zazwyczaj jest on łatwiejszy w konfiguracji i pozwala na pracę z dowolnego komputera działającego w Twojej sieci.

Adres strony internetowej domowego rutera to (najczęściej) *192.168.1.1*. Spróbuj otworzyć go w Twojej przeglądarce.

Oto zrzut ekranu ustawień bezpieczeństwa przykładowego rutera:



Prawdopodobnie trzeba będzie wprowadzić identyfikator użytkownika i hasło. Zajrzyj do dokumentacji rutera, aby znaleźć domyślne ustawienia (o ile nie uległy one zmianie). Przejdź na zakładkę *Bezpieczeństwo* (*Security*) lub podobną, aby uzyskać dostęp do omawianych funkcjonalności.

Zobacz również

Na temat polecenia `iptables` można by napisać całą książkę. Wiele informacji znajdziesz w podręczniku `man`, a także w książkach poświęconych zaporom sieciowym. Istnieje także wiele ciekawych stron internetowych związanych z tą tematyką.

Obsługa Secure Linux (SELinux)

W tym podrozdziale zajmiemy się omówieniem usługi Security Enhanced Linux (SELinux). W poradzie Obsługa uprawnień plików omawialiśmy standardowy sposób ochrony systemu w Linuksie. Metoda ta nosi nazwę uznaniowej kontroli dostępu (ang. *Discretionary Access Control* — DAC) i ma pewne ograniczenia. Zwykły użytkownik może udostępnić swoje pliki (przypadkowo lub celowo) innym użytkownikom do odczytu lub zapisu, co w konsekwencji może prowadzić do ujawnienia istotnych informacji. Większy poziom bezpieczeństwa zapewnia obowiązkowa kontrola dostępu (ang. *Mandatory Access Control*) stosowana w SELinux. MAC korzysta z polityki bezpieczeństwa, która obejmuje wszystkie procesy i pliki w systemie. Wszystkie pliki w SELinux mają etykiety, które zawierają informacje związane z bezpieczeństwem.

Przykładowo poniższy listing przedstawia plik objęty kontrolą DAC:

```
ls -la ifcfg-eth0
-rw-r--r--. 1 root root 73 Apr 22  2011 ifcfg-eth0
```

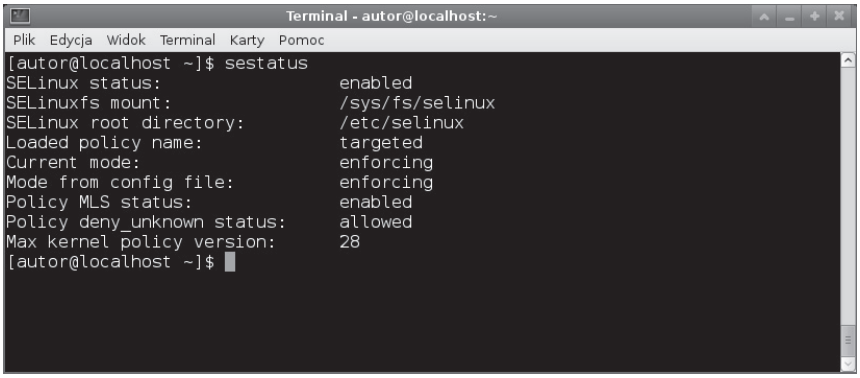
Ten sam plik sprawdzony za pomocą opcji Z (kontekst bezpieczeństwa) wygląda następująco:

```
ls -Z ifcfg-eth0
-rw-r--r--. root root unconfined_u:object_r:default_t:s0ifcfg-eth0
```

Ciąg `unconfined_u` oznacza użytkownika, `object_r` — rolę, `default_t` — typ, a `s0` — poziom. To właśnie te informacje pozwalają na podejmowanie przez system decyzji dotyczących bezpieczeństwa. Pamiętaj, że najpierw są sprawdzane reguły DAC — jeśli nie pozwalają one na podjęcie działań, to reguły SELinux w ogóle nie będą sprawdzane.

Przygotuj się

Wykonamy jedynie kilka poleceń na koncie administratora; przejrzymy także niektóre ustawienia. Nie będziemy wprowadzać żadnych zmian w konfiguracji. W tym przykładzie zakładam, że SELinux działa w trybie restrykcyjnym (ang. *enforcing mode*). Aby określić tryb, w którym się znajdujesz, wywołaj polecenie sestatus. Efekt powinien być podobny do tego:



Jak to zrobić

1. Wykonaj polecenie `getenforce`; powinien zostać wyświetlony komunikat o włączeniu trybu restrykcyjnego (ang. *enforcing mode*).
2. Przejrzyj listę powiązań:

   ```
   semanage login -l
   ```

3. Aby przejrzeć konteksty SELinux dla wykonywanych procesów, wykonaj poniższe polecenie:

   ```
   ps -eZ
   ```

4. Aby obejrzeć kontekst użytkownika, wykonaj poniższe polecenie:

   ```
   id -Z
   ```

5. Polecenie `sealert` jest wykorzystywane do przeglądania pełnych komunikatów SELinux w momencie wystąpienia błędów. Sprawdź zawartość pliku */var/log/messages*, aby się przekonać, czy zostały wygenerowane jakiekolwiek błędy. Jeśli tak, możesz wykonać polecenie `sealert -l` dla komunikatu o wybranym numerze, aby dowiedzieć się więcej.
6. Szczegółowa lista wartości logicznych SELinux z opisami jest dostępna za pomocą polecenia:

   ```
   semanage boolean -l
   ```

7. Aby obejrzeć listę bez opisów, wykonaj poniższe polecenie:

   ```
   getsebool -a
   ```

8. Aby sprawdzić, czy pliki i katalogi dysponują odpowiednio uruchomionym kontekstem SELinux, wykonaj poniższe polecenie:

   ```
   matchpathcon
   ```

Zobacz również

Jak już wspomniałem, SELinux jest instalowany domyślnie w większości dystrybucji. Czasami możesz nie zdawać sobie sprawy, że jest on dostępny. Niekiedy może się jednak okazać, że z powodu jego obecności będą wynikać różne problemy. Jeśli spróbujesz zainstalować usługę — np. `vsftpd` — może się okazać, że nie będzie to możliwe z powodu wystąpienia konfliktu z polityką SELinux. Zazwyczaj w takich sytuacjach otrzymasz komunikat o błędzie. Choć komunikat ten może okazać się pomocny w rozwiązaniu problemu, z mojego doświadczenia wynika, że takie porady nie są przydatne. Mimo wykonania opisanych kroków dostęp nadal nie będzie możliwy. W takich sytuacjach przełączam tryb SELinux na tryb zezwalania (ang. *permissive mode*) i kontynuuję wykonywane przeze mnie operacje:

```
setenforce 0
```

Zwróć uwagę, że taki trik zadziała tylko do następnego restartu.

Więcej informacji na temat SELinux znajdziesz w doskonałym przewodniku na stronie internetowej dystrybucji Fedora.

Korzystanie z narzędzia sudo

Czasami (zwłaszcza gdy jesteś administratorem systemu) istnieje konieczność przyznania niektórym użytkownikom większego dostępu do komputera. Jednocześnie nie powinni oni mieć dostępu do konta administratora. Te dwa założenia da się połączyć, modyfikując plik */etc/sudoers* i korzystając z narzędzia `sudo`.

Przygotuj się

Poniższe czynności nie powinny uszkodzić Twojego systemu. Wykonamy je z poziomu konta użytkownika utworzonego powyżej. W ramach tej porady będzie niezbędny dostęp do konta administratora.

Jak to zrobić

1. Najpierw zrób kopię zapasową pliku */etc/sudoers*:

   ```
   cp /etc/sudoers /tmp/sudoers.orig
   ```

2. Pliku *sudoers* nie edytuje się bezpośrednio — służy do tego celu polecenie `visudo`. Nazwa tego polecenia nie jest zbyt praktyczna, ponieważ nie musisz korzystać akurat z `vi` — możesz ustawić dowolny edytor, korzystając ze zmiennej `EDITOR`. Ustaw edytor zgodnie ze swoimi preferencjami, a następnie wykonaj polecenie:

   ```
   visudo
   ```

3. To polecenie utworzy tymczasową kopię pliku *sudoers* i otworzy ją do edycji. Jeśli wszystko zostanie wykonane poprawnie, po zakończeniu edycji tymczasowa kopia zastąpi oryginalny plik.
4. Przejrzyj część pliku poświęconą aliasom. Zostały one podzielone na grupy, takie jak: *networking* (sieć), *software* (oprogramowanie), *services* (usługi), *locate* i inne. Za chwilę przekonamy się, jak działają przedstawione mechanizmy.
5. Najpierw jednak wykonamy pewne testy. Otwórz inną sesję (jako gość). Ja skorzystam ze swojego konta *autor*.
6. Jako *autor* wejdź do katalogu */tmp*:

   ```
   cd /tmp
   ```

7. Utwórz plik, korzystając z polecenia:

   ```
   ls>f1
   ```

8. Skopiuj ten plik do katalogu */usr/bin*:

   ```
   cp f1 /usr/bin
   ```

9. Powinieneś otrzymać komunikat o błędzie. Nie jest to nic dziwnego — zwykły użytkownik standardowo nie ma uprawnień do zapisu do katalogu */usr/bin*. Teraz powróćmy do sesji z otwartym narzędziem `visudo`.
10. Będzie Ci potrzebna nazwa hosta Twojego komputera. W tym przykładzie skorzystamy po prostu z adresu IP. W razie potrzeby możesz go uzyskać, korzystając z polecenia `ifconfig`.
11. Tuż za fragmentem pliku poświęconym poleceniu `shutdown` dodamy wiersz dla naszego użytkownika-gościa. Składnia jest następująca: użytkownik, nazwa hosta, polecenia i opcje. Dodaj więc poniższy wiersz:

    ```
    autor 192.168.1.115=(ALL)   ALL
    ```

12. Zapisz plik i zamknij sesję `visudo`. Spróbuj wykonać polecenie ponownie — z poziomu katalogu */tmp* wykonaj polecenie `cp f1 /usr/bin`. Nadal powinieneś otrzymać komunikat o błędzie. Teraz za to spróbuj wykonać polecenie po lekkiej modyfikacji:

    ```
    sudo cp f1 /usr/bin
    ```

13. Uch, czyżby kolejna prośba o hasło? W rzeczy samej, i co ważne, chodzi o hasło użytkownika, a nie administratora. W takiej sytuacji nietrudno o błąd. Najprościej będzie zapamiętać, że użytkownik-gość nie powinien w ogóle znać hasła administratora. Wprowadź zatem swoje hasło.
14. Jeśli jest to pierwsza sytuacja, w której korzystasz z polecenia `sudo`, zostanie wyświetlony dodatkowy komunikat. Warto się z nim dokładnie zapoznać.
15. Wreszcie polecenie powinno zostać wykonane bez problemów. Świetnie, prawda? Dzięki zastosowaniu ciągu `ALL` w pliku *sudoers* użytkownik otrzymał pełne uprawnienia. Zwróć uwagę, że niektóre mechanizmy dalej nie będą działać poprawnie — np. przekierowania.

16. No cóż, prawdopodobnie nie chcemy, aby użytkownik *autor* zamieszał zbyt dużo w systemie, dlatego ograniczymy jego możliwości. Wykonaj ponownie polecenie `visudo`.
17. Przewiń plik w dół lub znajdź ciąg `Processes`. Odkomentuj wiersz `# Cmnd_Alias PROCESSES`, usuwając początkowy znak `#`.
18. Przewiń plik w dół do wiersza zawierającego ustawienia użytkownika *autor*. Zmień go w następujący sposób:

    ```
    autor 192.168.1.115=(ALL) PROCESSES
    ```

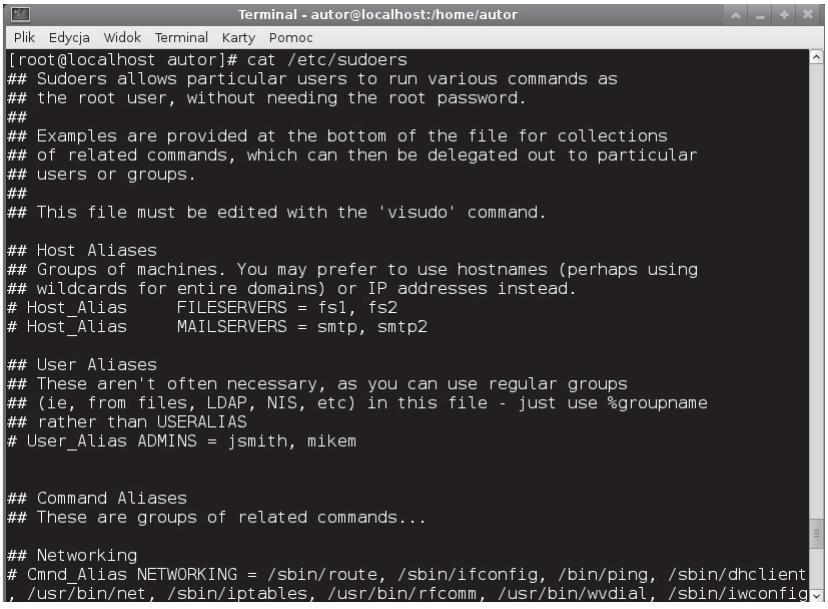
19. Teraz musimy znaleźć proces, który można zabić. Uruchom narzędzie `vi` z poziomu konta administratora — możesz np. wykonać polecenie `vi ksiazka`.
20. W ramach sesji swojego użytkownika wykonaj polecenie `ps auxw | grep "vi ksiazka"`. Zapamiętaj identyfikator procesu (ang. *Process ID* — PID).
21. W ramach sesji swojego użytkownika wykonaj polecenie `kill -9` dla podanego PID. Otrzymasz błąd. Teraz wykonaj je ponownie, korzystając za to z polecenia:

    ```
    sudo kill -9 <PID>
    ```

22. Proces z edytorem `vi` powinien zostać zabity.

> Jeśli ekran pozostanie niebieski, wykonaj polecenie `ls`. W ten sposób oczyścisz terminal.

Oto zrzut ekranu pochodzący z przykładowego pliku */etc/sudoers*:



Zobacz również

Jak widać, plik *sudoers* pozwala na bardzo wyrafinowane przyznawanie uprawnień. Użytkownicy mogą otrzymać przeróżne przywileje. Więcej informacji znajdziesz na stronach podręcznika `man`: `man sudoers`.

Moim zdaniem, wielu użytkownikom można zaufać, dając im pełen dostęp do konta *root* bez szkody dla systemu. Czasami zdarzają się jednak wyjątki. Konfigurując polecenie `sudo`, możesz spędzić wiele czasu na próbie ustawienia go w odpowiedni sposób, i to tylko po to, aby się na końcu dowiedzieć, że i tak musisz coś jeszcze dodać. Użytkownicy będą poirytowani, ponieważ nie będą w stanie wykonywać swojej pracy, dopóki nie wprowadzisz zmian. Wreszcie, gdy skonfigurujesz `sudo` w pełni, okaże się, że ktoś korzystając z niego, narobi szkód. Z mojego doświadczenia wynika, że jeżeli użytkownik popełni raz jakiś błąd, to będzie go popełniał stale.

Katalog /tmp

Katalog */tmp* jest wyjątkowy, ponieważ mogą do niego zapisywać pliki wszyscy użytkownicy. Oto wpis dla katalogu */tmp* na moim komputerze:

```
drwxrwxrwt.  10 root    root      4096 Mar 31 03:48 tmp
```

Jak widać, jest on dostępny dla wszystkich. Litera `t` w uprawnieniach oznacza, że bit ograniczonego usunięcia jest ustawiony dla tego katalogu. W przypadku katalogów oznacza to, że zwykli użytkownicy nie mogą usunąć lub zmienić nazwy tego katalogu, o ile nie mają specjalnych uprawnień.

Korzystając z katalogu */tmp* jako zwykły użytkownik, powinieneś uważać, ponieważ obowiązują w nim pewne ograniczenia.

Jak to zrobić

1. Wykonaj poniższe polecenie:

   ```
   cd /tmp
   ```

2. Jeśli w katalogu znajdują się pliki *.txt* pochodzące z poprzednich porad, usuń je — zwykłe polecenie `rm *.txt` powinno wystarczyć.

3. Wykonaj poniższe polecenie:

   ```
   ls>root1.txt
   ```

4. W kolejnej sesji (ja jako użytkownik-gość skorzystam z konta *autor*) wykonaj poniższe polecenie:

   ```
   cd /tmp
   ```

5. Wykonaj kolejne polecenie:

```
ls > autor.txt
```

6. Powyższe polecenie nie powinno sprawić kłopotów. Teraz wykonaj następne polecenie:

```
ls > root1.txt
```

7. Powinien zostać wyświetlony komunikat *Odmowa dostępu*. Dlaczego? Ponieważ mimo możliwości zapisu do katalogu przez wszystkich użytkowników tradycyjne uprawnienia (DAC) wciąż muszą być przestrzegane.

Zobacz również

Z opisanych powyżej względów nie radzę korzystać z katalogu */tmp* do czegokolwiek z wyjątkiem plików tymczasowych. Co więcej, spora część dystrybucji systemu Linux regularnie czyści katalog */tmp*, co powoduje, że wszystkie pliki niebędące w posiadaniu administratora są usuwane. Pamiętaj o tym, tworząc pliki tymczasowe — zwłaszcza gdy piszesz skrypty (więcej informacji na ten temat znajdziesz w rozdziale 8.).

Skorowidz

I

J

K

L

M

N

O

P

R

S

Ś

T

U

V

W

X

Z

Najlepsze narzędzia w systemie Linux

Linux to darmowy system operacyjny, ceniony przez wielu użytkowników na całym świecie. Jego niezawodność, wydajność i możliwości sprawiły, że jest on podstawowym systemem dla największych serwerów na świecie. Domowi użytkownicy mają za to dostęp do atrakcyjnego wizualnie oraz przyjaznego interfejsu graficznego. W świecie Linuksa każdy znajdzie coś dla siebie. Niezależnie od sposobu, w jaki wykorzystujesz ten system, powinieneś poznać możliwości jego konsoli, czyli trybu tekstowego. Może to być zaskakujące, ale właśnie dzięki wpisywanym poleceniom jesteś w stanie najszybciej zrealizować wiele zadań.

Oddajemy w Twoje ręce kolejną książkę z cenionej serii „Receptury". Książki te charakteryzują się doskonałym przedstawieniem typowych problemów i najlepszych metod ich rozwiązywania. W trakcie lektury tej książki dowiesz się, jak sprawnie korzystać z linii poleceń oraz jakie środowisko graficzne wybrać. Ponadto zaczniesz bez problemu poruszać się po systemie plików, wyszukiwać potrzebne dane oraz je kompresować. W kolejnych rozdziałach nauczysz się pobierać strony bez przeglądarki, tworzyć własny serwer WWW oraz kopiować pliki pomiędzy różnymi komputerami. Jeżeli interesuje Cię bezpieczeństwo systemu, tworzenie skryptów, zarządzanie procesami lub budowa własnego jądra systemu — znajdziesz tu liczne przykłady i wartościowe porady. Książka ta jest wspaniałą lekturą dla wszystkich pasjonatów i użytkowników systemu Linux!

Opanuj najlepsze techniki pracy z systemem Linux!

Dzięki tej książce:

- poznasz linię poleceń systemu Linux
- skonfigurujesz i uruchomisz własny serwer WWW
- poznasz dostępne środowiska graficzne
- zobaczysz, jak zbudować własne jądro
- zaczniesz biegle posługiwać się systemem Linux

ISBN 978-83-246-8980-4

9 788324 689804

Cena: 39,90 zł