

SCR - Sieci Komputerowe

Sprawozdanie 1 i 2

Tworzenie kont użytkowników i grup

Michał Kowalski
Amadeusz Janiszyn

poniedziałek TP 13¹⁵ – 15⁰⁰ Dr inż. Mariusz Makuchowski

16 marca 2020

Spis treści

| | | |
|----------|---------------------------------------------------------|----------|
| 1 | Cel ćwiczeń | 2 |
| 2 | Maszyna wirtualna | 2 |
| 3 | System Ubuntu | 2 |
| 4 | Konta użytkowników | 3 |
| 4.1 | Podstawowe informacje | 3 |
| 4.2 | Tworzenie kont użytkownika | 3 |
| 5 | Grupy | 4 |
| 5.1 | Podstawowe informacje | 4 |
| 5.2 | Tworzenie grup | 4 |
| 5.3 | Dodawanie użytkownika do grupy | 4 |
| 6 | Prawa dostępu do plików i folderów | 5 |
| 6.1 | polecenie <code>chmod</code> i jego parametry | 6 |
| 7 | Podsumowanie | 7 |

1 Cel ćwiczeń

Ćwiczenie pierwsze miało na celu zapoznanie się z praktycznym wykorzystaniem wirtualizacji, nabycie doświadczenia w obsłudze systemu operacyjnego Ubuntu, a także nauczenie poruszania się po terminalu oraz wykorzystywania podstawowych poleceń powłoki bash.

Dzięki ćwiczeniu nr 2 nauczyć się można było podstawowego zarządzania użytkownikami w systemie Linux z konta roota. Operacje takie jak tworzenie nowego konta, dodawanie do grupy oraz nadawanie odpowiednich praw dostępu do różnych plików i folderów zostało przedstawione.

2 Maszyna wirtualna

Wirtualizacja jest to możliwość emulacji komputera - maszyny wirtualnej. Od strony programowej pozwala to na uruchomienie wirtualnego systemu operacyjnego wewnątrz działającego systemu hostującego oraz programu do obsługi maszyn wirtualnych. Rozwiązanie to pozwala na testowanie działania programów na różnych (emulowanych) konfiguracjach sprzętowych różniących się ilością pamięci RAM oraz szybkością taktowania procesora. Maszyna wirtualna ma także swój własny dysk, który jest zwykłym plikiem na maszynie rzeczywistej. Zwykle nie ma dostępu do fizycznego dysku komputera, przez co można bezpiecznie testować na niej różne programy.

Dzięki wirtualizacji możliwe jest także jednoczesne uruchomienie kilku wirtualnych komputerów, połączenie ich w sieć i przesyłanie między nimi plików, zaznajamiając się w ten sposób z protokołami sieciowymi, bez konieczności posiadania kilku maszyn w domu. Dla potrzeb tego kursu, systemem hostującym jest Windows 10, programem obsługującym wirtualizację VirtualBox, a systemem emulowanym - Ubuntu 18.04 LTS.

3 System Ubuntu

Ubuntu jest systemem operacyjnym rozwijanym przez lata przez firmę Canonical na wolnej licencji typu GNU GPL. Jest zrobiony na bazie Debiana, jego domyślnym środowiskiem graficznym jest GNOME, jak wiele systemów uniksopodobnych korzysta z kernelu Linuxa.

W systemie tym (i innych podobnych jak Mint, Debian, GNU itp) bardzo dużą rolę odgrywa powłoka, która wspomaga komunikację między systemem operacyjnym a użytkownikiem. Przykładem takiej powłoki jest bash. Po włączeniu terminala, wpisane

polecenia są przez `bash` interpretowane, a następnie wykonywane przez inne składniki systemu operacyjnego (najniższą warstwą w tym procesie jest `kernel` - jądro, które komunikuje się bezpośrednio ze sprzętem). Włączając programy graficzne, także wybieranie odpowiednich funkcji powoduje wysyłanie kolejnych poleceń do powłoki, która zleca ich wykonanie odpowiednim częścią systemu.

4 Konta użytkowników

4.1 Podstawowe informacje

W systemie Ubuntu użytkownicy mają możliwość pracy na wielu kontach użytkownika, tak aby na jednym komputerze (PC lub serwerze) mogło bezpiecznie pracować kilka osób (w przypadku serwerowym także jednocześnie). Każdy użytkownik ma swój folder (zwykle w `/home/`), nad którego zawartością powinien mieć pełną władzę. Nie powinien jednak mieć dostępu do rzeczy, które nie są jego. W tym celu stosuje się podział na grupy. Domyślnie na komputerze zawsze jest konto administratora nazywające się `root`. Konto to ma dostęp do każdego pliku na dysku komputera oraz pełną kontrolę nad kontami pozostałych użytkowników. Będąc zalogowanym jako `root` można przez nieuwagę mocno zaszkodzić systemowi operacyjnemu, co w przypadku innych kont jest dużo trudniejsze. Podstawowe informacje o użytkownikach znaleźć można w pliku `/etc/passwd`.

4.2 Tworzenie kont użytkownika

Aby utworzyć konto użytkownika można skorzystać z dwóch poleceń:

- `useradd`,
- `adduser`.

Polecenie `useradd` jest bardzo proste w działaniu - dodaje po prostu wpis o nowym użytkowniku do pliku `/etc/passwd`. Wymaga ono, by potem ustalić domyślną powłokę dla nowego użytkownika, dodać mu hasło, założyć folder itp.

Dużo bardziej złożony jest `adduser`. W tym programie podać trzeba będzie od razu nowe hasło dla użytkownika oraz można będzie wpisać dodatkowe dane (imię i nazwisko, nr tel itp.). Program ten automatycznie założy użytkownikowi folder oraz ustawi powłokę na `bash`.

Przykład użycia:

- `useradd abc` - Stworzy podstawę pod utworzenie konta użytkownika zwanego „abc”,
- `adduser abc` - Zada dodatkowe pytania i stworzy w pełni gotowe konto użytkownika o nazwie „abc”.

5 Grupy

5.1 Podstawowe informacje

Grupy są bardzo ważną funkcjonalnością systemu, zwłaszcza przy serwerze, na którym użytkownicy pracują zdalnie w jednym czasie. Dzięki grupom możemy rozdzielić użytkowników na kilka działów i zarządzać każdym z nich z osobna. Każdy użytkownik może być w więcej niż jednej grupie. Na przykład użytkownik X jest w grupie elektroników, użytkownik Y jest w grupie programistów, a użytkownik Z w obu tych grupach. Przy prawidłowo skonfigurowanych prawach dostępu do plików (o czym później), tylko użytkownicy Y i Z będą mieli dostęp do kodu programów, a tylko użytkownicy X i Z dostęp do projektów płytek PCB.

W systemie Ubuntu informacje o grupach przechowywane są w `/etc/group`.

5.2 Tworzenie grup

Aby utworzyć grupę korzysta się z polecenia `addgroup`. Jego używanie jest bardzo proste - wystarczy podać jako parametr polecenia nazwę tworzonej grupy. Dozwolone jest korzystanie z wielu opcji dodatkowych, jak np. dodanie własnego numeru GID (zamiast następnego w kolejności).

Przykład użycia:

`addgroup studenci` - Tworzy pustą grupę o nazwie „Studenci”.

5.3 Dodawanie użytkownika do grupy

Aby dodać użytkownika do grupy potrzebne będzie polecenie `usermod`. Korzysta się zazwyczaj z niego w następujący sposób:

`usermod -a -G grupa użytkownik`

Na przykład:

`usermod -a -G studenci abc` - Dodaje konto „abc” do grupy „studenci”. Grupa musi istnieć - w razie pomyłki w nazwie nie zostanie utworzona automatycznie nowa grupa, tylko operacja zostanie przerwana.

Stan grup możemy zawsze kontrolować w pliku `/etc/group`.

6 Prawa dostępu do plików i folderów

W systemie Ubuntu (i podobnych) każdy plik może być używany w różny sposób przez różnych użytkowników. Uprawnienia do pliku dzielą się na 3 typy:

- **r** - Prawo do odczytu (możliwość wyświetlenia pliku, skopiowania go oraz wyświetlenia listy plików w przypadku folderu),
- **w** - Prawo do zapisu (możliwość nadpisania pliku, przeniesienia, a także usunięcia),
- **x** - Prawo do wykonywania (możliwość wykonania programu oraz wejścia do folderu).

Dodatkowo każde uprawnienie można nakładać na 3 różne grupy zasięgowe:

- właściciel pliku,
- grupa,
- inni - użytkownicy spoza grupy.

Po wpisaniu polecenia `ls -l` ukaże się lista plików wraz z kodem uprawnień w takiej postaci:

```
drwxrwxrwx
```

gdzie pierwszy znak informuje o tym, czy jest to folder, 3 kolejne o prawach dostępu dla właściciela, 3 kolejne o prawach dostępu dla grupy i 3 ostatnie - o prawach dostępu dla innych użytkowników.

Na przykład:

drwxr-x-- - Folder, do którego właściciel może wejść, odczytać jego zawartość, zapisywać w nim pliki. Grupa właściciela może jedynie wejść i odczytać zawartość. Inni nie mogą zrobić żadnej czynności z folderem.

-rwx-x-x - Plik, który może być wyświetlony i edytowany tylko przez właściciela, ale uruchomiony przez dowolnego użytkownika.

-r-r-rwx - Plik, który może być tylko odczytany przez właściciela i jego grupę, jednak także edytowany i uruchamiany przez innych użytkowników. Jest to sytuacja bardzo rzadka i raczej wynikająca z pomyłki.

6.1 polecenie chmod i jego parametry

Każdy plik oraz folder ma przypisane określone prawa dostępu dla różnych użytkowników systemu.

Polecenie `chmod` zmienia parametry zezwolenia dostępu do plików w systemie Ubuntu.

Parametry polecenia `chmod` :

`chmod [opcje] uprawnienia plik`

Opcje:

| | |
|----------------------------------------|----------------------------------------------------------------------------------|
| <code>-c, --changes</code> | jak <code>-v</code> , ale podanie tylko kiedy zaszła zmiana |
| <code>--no-preserve-root</code> | bez traktowania katalogu <code>'/'</code> w specjalny sposób |
| <code>--preserve-root</code> | odmowa rekursywnego działania na <code>'/'</code> |
| <code>-f, --silent, --quiet</code> | wyłączenie większości komunikatów o błędach |
| <code>-v, --verbose</code> | wypisanie informacji o każdym przetwarzanym pliku |
| <code>--reference=plik_wzorcowy</code> | użycie uprawnień pliku <i>plik_{wzorcowy} zamiast wartości uprawnień</i> |
| <code>-R, --recursive</code> | zmiany też w plikach w podkatalogach |
| <code>--help</code> | wyświetlenie tego opisu i zakończenie |
| <code>--version</code> | wyświetlenie informacji o wersji i zakończenie |

Uprawnienia jakie można nadać plikom:

- `r` - odczyt
- `w` - zapis
- `x` - wykonanie

Opis klas użytkowników, którym można zmienić uprawnienia:

- `user` - właściciel (`u`)
- `group` - grupa (`g`)
- `inni` - użytkownicy spoza grupy (`o`)
- `a` - wszyscy użytkownicy (`a`)

Przykład użycia:

`chmod o+x plik.txt` - Nadaje prawo do uruchomienia dla innych dla pliku „plik.txt”.

7 Podsumowanie

W systemie Ubuntu dowolnie można modyfikować grupy oraz dostęp do różnych plików. Będąc administratorem, posiada się dostęp do całej palety poleceń, dzięki którym można modyfikować właściwości plików, grup oraz poszczególnych użytkowników. Dokumentacja poleceń pozwala na bezproblemową modyfikację składników systemu, w zależności od tego, co w danym momencie potrzebujemy zrobić.