# Bezpieczeństwo OS Linux

## Zadanie 1: Monitorowanie sytemu

Zadanie jest wykonane w systemach Ubuntu Live

Otwórz terminal i wykonaj polecenie:
```
$sudo su
```

Sprawdź zalogowanych w systemie użytkowników:
```
#who
```

Polecenie ***who*** wyświetla informację o zalogowanych w systemie użytkownikach.

Sprawdź informację o ostatnich logowaniach w systemie:
```
#last
```

Polecenie ***last*** wyświetla informację o ostatnich logowaniach w systemie.

Pakiety ***who*** i ***last*** odczytują informację z pliku logów systemu które są umieszczone w katalogu: ***/var/log/***

Obejrzyj zawartość tego katalogu:
```
#ls -lh /var/log/
```

Ważnymi dla nas w ramach ćwiczenia są trzy pliki logów:

- ***syslog*** - globalny plik logów zawiera wpisy od momentu uruchomienia systemu, od jądra Linux , różnych usług, wykrytych narzędzi, interfejsów sieciowych i t.d.
- ***auth.log*** – informacja o logowaniu, otwartych i zamkniętych sesjach w systemie.
- ***btmp*** – nieudane próby logowania.

Obejrzyj zawartość pliku ***syslog***:
```
#cat  /var/log/syslog
```

Plik ***syslog*** zawiera dużą ilość informacji. Dla tego nie zawsze jest przydatne korzystać z polecenia **cat**. Lepeij skorzystać z polecenia ***tail -n <liczba_ostatnich_wierszy>***
```
#tail -n 25  /var/log/syslog
```

Dla analizy możemy również skorzystać z poleceń ***more*** i ***less***. Jeżeli wiadomo jest co szukamy możemy skorzystać z polecenia ***grep <co_szukamy_w_pliku>***
```
#cat /var/log/syslog | grep Boot
```

Znajdź i zademonstruj prowadzącemu wierszy zwierające „sudo” w pliku ***syslog***

Obejrzyj zawartość pliku ***auth.log***:
```
#cat /var/log/auth.log
```

Obejrzyj zawartość pliku ***btmp***:
```
#cat /var/log/btmp
```

Dla czego ***btmp*** jest pusty?

Wszystkie procesy uruchomione w systemie możemy obejrzeć za pomocą polecenia ***ps -uax***.
Wyświetl wszystkie procesy:
```
#ps -aux
#ps -aux | more
```

Znajdź proces/procesy uruchomione z poleceniem „sudo” za pomocą poleceń ***ps -aux*** i ***grep***.
Zademonstruj prowadzącemu wykryty proces/procesy.

W systemie Linux jest polecenie ***top***, które pozwala na bieżąco monitorować aktywne procesy i sortować ich po wybranych parametrach (domyślne procesy są sortowane po obciążeniu procesora). Co pozwala wykrywać procesy obciążające procesor lub pamięć lub inne anomalie.
Obejrzyj aktywne procesy za pomocą polecenia ***top***:
```
#top
```

Otwórz w przeglądarce internetowej dowolną stronę, na przykład wp.pl. Co jest widoczne w wyświetlanej poprzez ***top*** informacji? Zademonstruj prowadzącemu.

Ważnym aspektem bezpieczeństwa systemu jest monitorowanie otwartych portów sieciowych. Obejrzeć otwarte porty sieciowe można za pomocą polecenia ***netstat***.
Sprawdź otwarte porty w systemie za pomocą polecenia ***netstat***:
```
#netstat –tupan
```

Monitorowanie otwartych portów na bieżąco w celu wykrycia anomalii jest możliwe za pomocą kombinacji poleceń ***watch -n<cas_ powtarzania_w_sekundach*** > i ***netstat***.
Uruchom monitorowanie otwartych portów co 0.2 sekundy:
```
#watch -n0.2 netstat -tupan
```

Otwórz w przeglądarce internetowej dowolną stronę, na przykład wp.pl. Jakie zmiany w otwartych portach to powoduje? Zademonstruj prowadzącemu.

## Zadanie 2: Zarządzanie użytkownikami

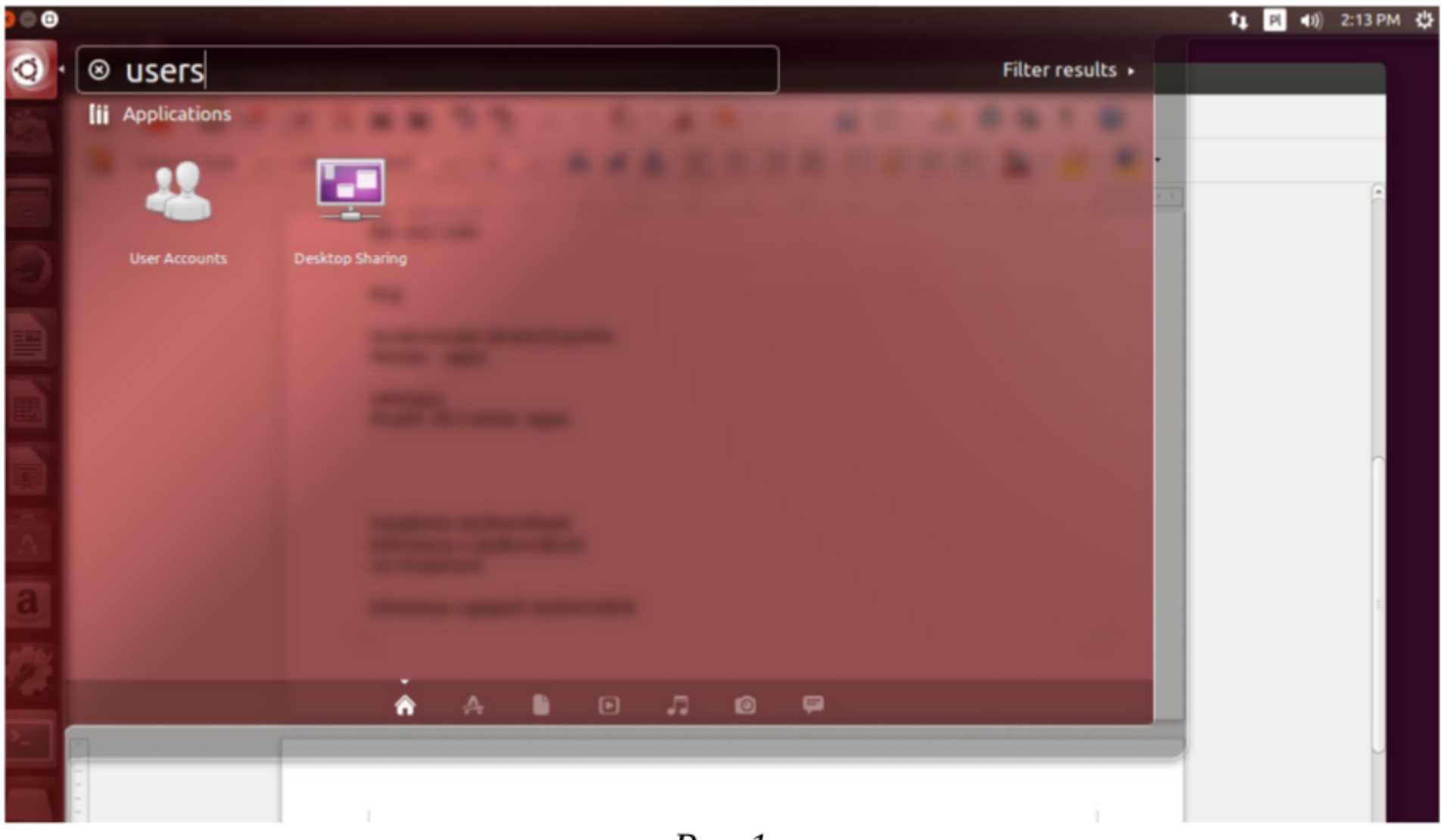
Zadanie jest wykonane w systemach Ubuntu Live

Informację o użytkownikach w systemie Linux można wyświetlić poleceniem
**#cat /etc/passwd**

Informacja o grupach użytkowników jest dostępna w pliku ***/etc/group***
Wyświetl informację o grupach użytkowników:
**#cat /etc/group**

We współczesnych systemach Linux jest możliwe zarządzanie użytkownikami w trybie graficznym.

Załóż nowego użytkownika-administratora o nazwie „User_GUI”.
Dla tego uruchom pakiet „User Accounts” jak jest zademonstrowane na rysunku poniżej.

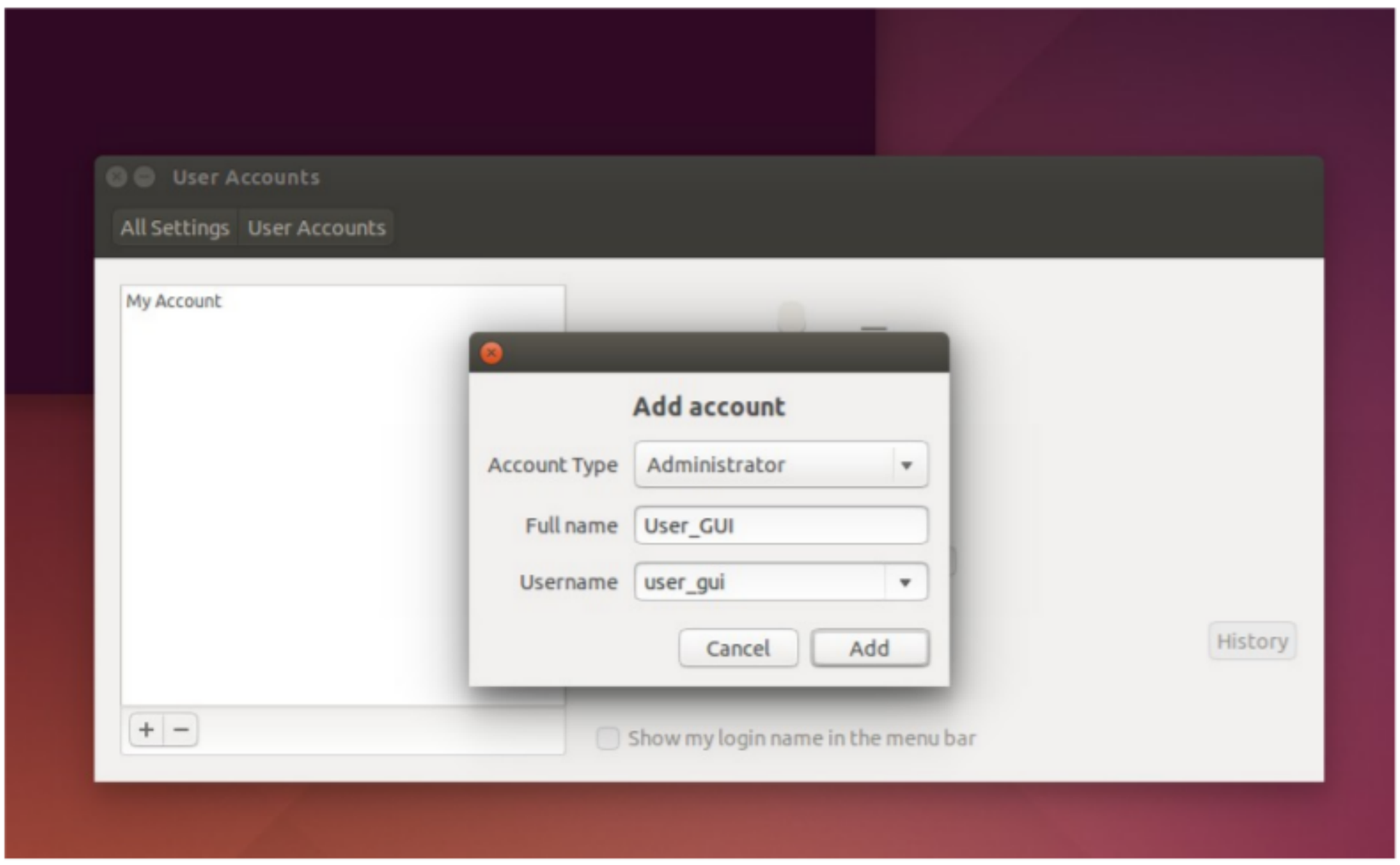



*Rys. 1*

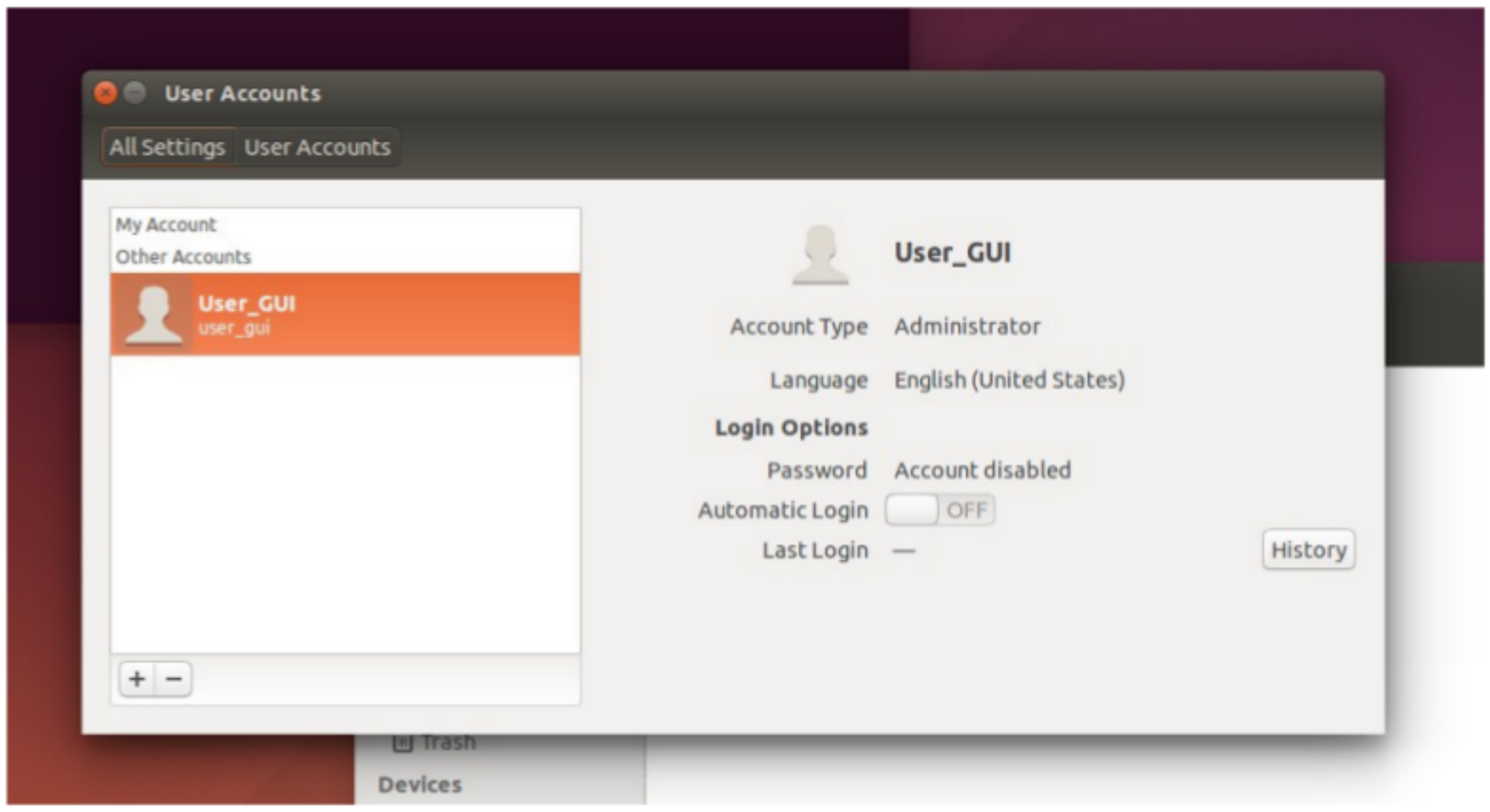
Załóż nowego użytkownika profilu „Adminstrator” o nazwie „User_GUI” i o grupie „user_gui” (jest zademonstrowane na rysunkach 2 i 3).

Ustaw hasło użytkownika (Rys. 4) **UWAGA!!! Warto zapamiętać nadawane hasło, ono jeszcze przyda się. Bez tego hasła nie uda się odrobić dalszej części ćwiczenia**.

W wyniku tych operacji okienko aplikacji „User Accounts” ma wyglądać w sposób prezentowany na rysunku 5

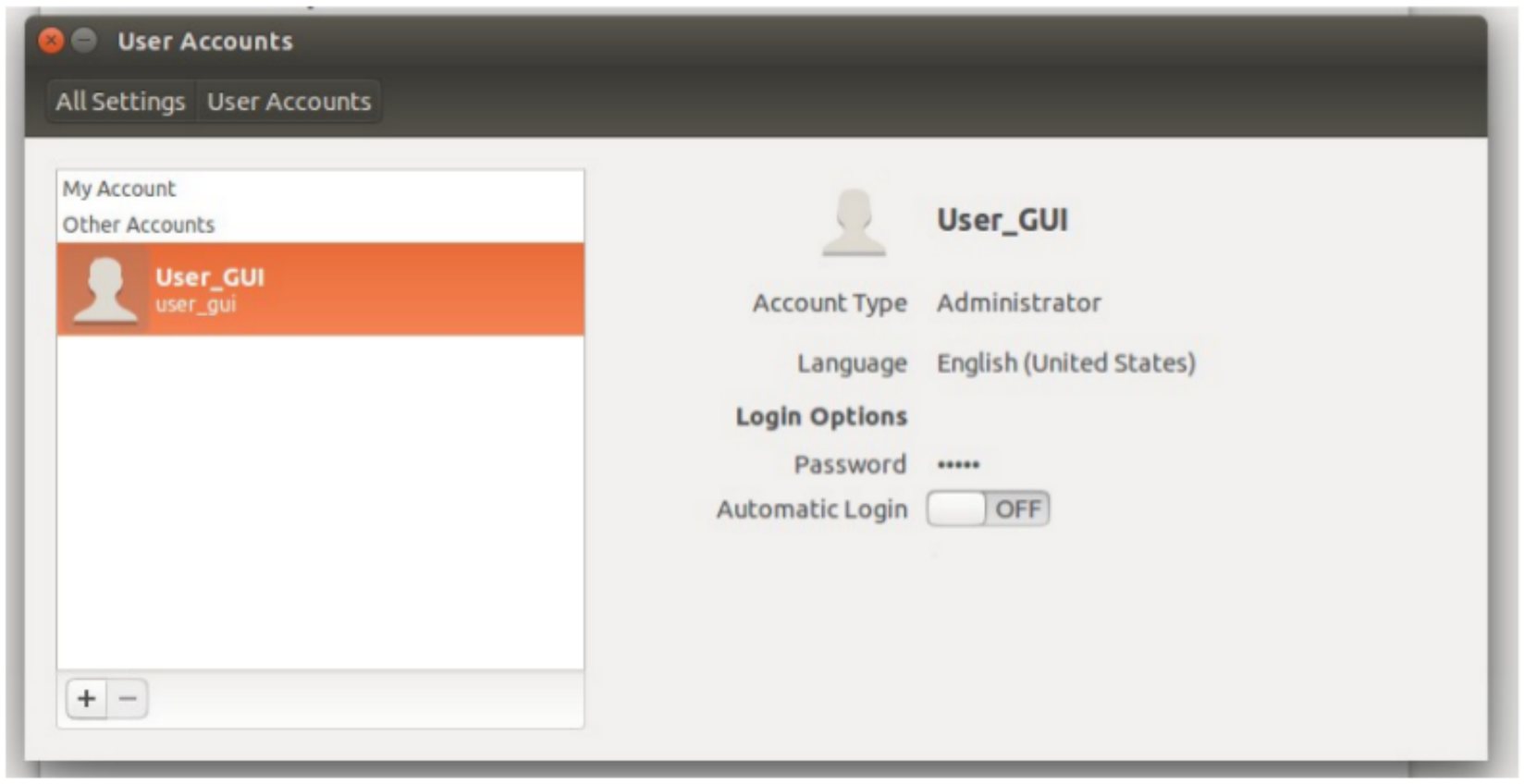



*Rys. 2*




*Rys. 3*


*Rys. 4*




*Rys. 5*

Zaloguj się pod użytkownikiem „User_GUI”. Dla tego wyloguj się z systemu i zaloguj się na nowo wybierając dostępnego użytkownika „User_GUI” i wprowadzając ustawione wcześniej hasło.
Otwórz terminal i wykonaj polecenie:

```
$sudo su
```

Sprawdź zalogowanych w systemie użytkowników:

```
#who
```

Sprawdź informację o ostatnich logowaniach w systemie:

```
#last
```

Obejrzyj zawartość pliku ***auth.log***:

```
#cat /var/log/auth.log
```

Obejrzyj zawartość pliku ***syslog***:

```
#tail -n 30 /var/log/syslog
```

Bardzo często Linux spełnia funkcję systemu operacyjnego serwerów i innych narzędzi bez instalacji pakietów interfejsu graficznego. Dla tego warto znać polecenia które pozwalają zarządzać użytkownikami bez stosowania graficznego interfejsu.

Załóż użytkownika o nazwie „user_add” za pomocą polecenia ***useradd***:

```
#useradd -m user_add
```

klucz „-m” wskazuje programowi na to, że dla użytkownika ma być założony domowy katalog w dyrektorii **/home/**

Po założeniu użytkownika poprzez polecenie ***useradd*** on nie ma ustawionego hasła i nie jest aktywny. Ustaw hasło dla użytkownika „user_add” za pomocą polecenia:

```
#passwd user_add
```

**UWAGA!!! Warto zapamiętać nadawane hasło, ono jeszcze przyda się. Bez tego hasła nie uda się odrobić dalszej części ćwiczenia.**

Załóż użytkownika „add_user” poprzez polecenie ***adduser***:

```
#adduser add_user
```

**UWAGA!!! Warto zapamiętać nadawane hasło, ono jeszcze przyda się. Bez tego hasła nie uda się odrobić dalszej części ćwiczenia.**

Sprawdź zawartość plików ***/etc/passwd*** i **/etc/group**. Zademonstruj prowadzącemu wiersze, które wskazują na to, że w systemie są założone użytkowniki „User_GUI”, „user_add”, „add_user”.

## Zadanie 3: Discretionary Access Control (DAC)

Zadanie jest wykonane w systemach Ubuntu Live

Załóż w katalogu ***/home/ubuntu/*** pliki ***wszystkie.txt***, ***dla_user_add.txt***, ***dla_add_user.txt*** o dowolnej treści:

```
#cd /home/ubuntu
#nano wsystkie.txt
```

Zapisz w plik cokolwiek

```
#nano dla_user_add.txt
```

Zapisz w plik cokolwiek

```
#nano dla_add_user.txt
```

Zapisz w plik cokolwiek

Wyświetl zawartość katalogu:

```
#ls -lh
drwxr-xr-x 2 ubuntu ubuntu 80 Nov 26 14:22 Desktop
-rw-r--r-- 1 root   root   13 Nov 26 14:45 dla_add_user.txt
-rw-r--r-- 1 root   root   17 Nov 26 14:44 dla_user_add.txt
drwxr-xr-x 2 ubuntu ubuntu 40 Nov 26 14:22 Documents
drwxr-xr-x 2 ubuntu ubuntu 40 Nov 26 14:22 Downloads
drwxr-xr-x 2 ubuntu ubuntu 40 Nov 26 14:22 Music
drwxr-xr-x 2 ubuntu ubuntu 80 Nov 26 14:24 Pictures
drwxr-xr-x 2 ubuntu ubuntu 40 Nov 26 14:22 Public
drwxr-xr-x 2 ubuntu ubuntu 40 Nov 26 14:22 Templates
drwxr-xr-x 2 ubuntu ubuntu 40 Nov 26 14:22 Videos
-rw-r--r-- 1 root   root   14 Nov 26 14:44 wszystkie.txt
```

Dla zmiany właściciela pliku obecnie jest stosowane polecenie ***chown***:
***chown <nazwa_użytkownika>:<grupa_użytkowników> <nazwa pliku>***

Zmień właściciela pliku ***dla_add_user.txt*** za pomocą polecenia ***chown***:

```
#chown add_user:add_user dla_add_user.txt
```

Zmień właściciela pliku ***dla_user_add.txt*** za pomocą polecenia ***chown***.
Wyświetl zawartość katalogu za pomocą plecenia ***ls -lh***.
Kto jest właścicielem pliku ***dla_add_user.txt***?
Kto jest właścicielem pliku ***dla_user_add.txt***?

Dla zmiany uprawnień dostępu do pliku obecnie jest stosowane polecenie ***chmod***.
***chmod <dw><dgw><dpu> <nazwa pliku>***
gdzie pola:

- <***dw***> - reprezentuje uprawnienia **d**ostępu dla **w**łaściciela pliku
- <***dgw***> - reprezentuje uprawienia **d**ostępu dla **g**rupy **w**łaścicieli
- <***dpu***> - reprezentuje uprawienia **d**ostępu dla **p**ozostałych **u**żytkowników

Uprawienia są reprezentowane w postaci liczbowej od 0 do 7.
Dla wyjaśniania mechanizmu tej reprezentacji warto zobaczyć jak wyglądają tamte liczby (0-7) w postaci dwójkowej:

| liczba | trzeci bit | drugi bit | pierwszy bit | Uprawnienia dostępu |
|---|---|---|---|---|
| **0** | 0 | 0 | 0 | Nie można nić |
| **1** | 0 | 0 | 1 | Można uruchomić plik: (--x) e**x**ecute |
| **2** | 0 | 1 | 0 | Można zrobić wpis (-w-): **w**rite |
| **3** | 0 | 1 | 1 | Można zrobić wpis i uruchomić: (-wx) **w**rite+e**x**ecute |
| **4** | 1 | 0 | 0 | Można odczytać plik: (r--) **r**ead |
| **5** | 1 | 0 | 1 | Można odczytać i uruchomić: (r-x) **r**ead+e**x**ecute |
| **6** | 1 | 1 | 0 | Można odczytać i zrobić wpis: (rw-) **r**ead+**w**rite |
| **7** | 1 | 1 | 1 | Można wszystko: (rwx) **r**ead+**w**rite+e**x**ecute |
| | **r** | **w** | **x** | **< = Znaczenia bitów** |

Jeżeli pierwszy bit jest 1 – pozwala to uruchomiać plik (plik jest aplikacją lib skryptem).
Jeżeli drugi bit jest 1 – pozwala to robić wpis w plik, edytować go.
Jeżeli trzeci bit jest 1 – pozwala to odczyt pliku.

Przykład 1:

```
#chmod 421 test.txt
#ls -lh
-r---w---x 1 user1 users   0 Nov 26 22:36  test.txt
```

Polecenie ***chmode 421 test.txt*** ustawia uprawnienia dostępu dla pliku ***test.txt***:

- dla właściciela **4** - odczyt pliku;
- dla grupy właścicieli **2** – wpis, edycja pliku;
- dla pozostałych użytkowników **1** – uruchomienie pliku;

Przykład 2:

```
#chmod 640 test.txt
#ls -lh
-rw-r----- 1 user1 users   0 Nov 26 22:36  test.txt
```

Polecenie ***chmode 640 test.txt*** ustawia uprawnienia dostępu dla pliku ***test.txt***:

- dla właściciela **6** - odczyt pliku, wpis (edycja pliku);
- dla grupy właścicieli **2** – odczyt pliku;
- dla pozostałych użytkowników **0** – nie można nić;

Przykład 3:

```
#chmod 777 test.txt
#ls -lh
-rwxrwxrwx 1 user1 users   0 Nov 26 22:36  test.txt
```

Polecenie ***chmode 777 test.txt*** ustawia uprawnienia dostępu dla pliku ***test.txt***:

- dla właściciela **7** - odczyt pliku, wpis (edycja pliku), uruchomienie;
- dla grupy właścicieli **7** - odczyt pliku, wpis (edycja pliku), uruchomienie;
- dla pozostałych użytkowników **7** - odczyt pliku, wpis (edycja pliku), uruchomienie;

Ustaw dla pliku ***dla_add_user.txt*** uprawnienia dostępu:

- dla właściciela - odczyt pliku, wpis (edycja pliku);
- dla grupy właścicieli - odczyt pliku, wpis (edycja pliku);
- dla pozostałych użytkowników – odczyt pliku;

Ustaw dla pliku ***dla_user_add.txt*** uprawnienia dostępu:

- dla właściciela - odczyt pliku, wpis (edycja pliku);
- dla grupy właścicieli - odczyt pliku, wpis (edycja pliku);
- dla pozostałych użytkowników – nie można nic;

Wyświetl zawartość katalogu i zademonstruj uprawnienia dla plików prowadzącemu:

```
#ls -lh
```

Sprawdź czy ma użytkownik ***root*** dostęp do pliku ***dla_user_add.txt***:

```
#cat dla_user_add.txt
```

Warto zrozumieć, że mimo to, że pozostałe użytkowniki nie mają dostępu do pliku ***dla_user_add.txt*** użytkownik ***root*** jest królem w systemie i zwykły system DAC w żaden sposób nie ogranicza jego absolutnych uprawnień na wszystkie zasoby systemu. Dla tego każdy atakujący będzie spróbował uzyskać uprawnienia ***root*** w systemie.

Wyloguj się z systemu i zaloguj się na nowo wybierając użytkownika „user_add”.
Przejdź do katalogu ***/home/ubuntu***:

```
$cd /home/ubuntu
```

Sprawdź dostęp do pliku ***wszystkie.txt***. Czy można go odczytać:

```
$cat wszystkie.txt
```

Czy można go edytować:

```
$nano wszystkie.txt
```

Sprawdź dostęp do pliku ***dla_user_add.txt***. Czy można go odczytać:

```
$cat dla_user_add.txt
```

Czy można go edytować:

```
$nano dla_user_add.txt
```

Sprawdź dostęp do pliku ***dla_add_user.txt***. Czy można go odczytać:

```
$cat dla_add_user.txt
```

Czy można go edytować:

```
$nano dla_add_user.txt
```

Wyloguj się z systemu i zaloguj się na nowo wybierając użytkownika „add_user”.
Przejdź do katalogu ***/home/ubuntu***:

```
$cd /home/ubuntu
```

Sprawdź dostęp do pliku ***wszystkie.txt***. Czy można go odczytać:

```
$cat wszystkie.txt
```

Czy można go edytować:

```
$nano wszystkie.txt
```

Sprawdź dostęp do pliku ***dla_user_add.txt***. Czy można go odczytać:

```
$cat dla_user_add.txt
```

Czy można go edytować:

```
$nano dla_user_add.txt
```

Sprawdź dostęp do pliku ***dla_add_user.txt***. Czy można go odczytać:

```
$cat dla_add_user.txt
```

Czy można go edytować:

```
$nano dla_add_user.txt
```

## Zadanie 4: Konfiguracja AppArmor

Zadanie jest wykonane w systemach Ubuntu

Zaloguj się w systemu.
Login: student.
Hasło: kti

Otwórz terminal i wykonaj polecenie:

```
$sudo su
```

Zobaczymy profili AppArmor:

```
#ls -lh /etc/apparmor.d
```

Profili AppArmor są plikami tekstowymi. Nazwa profilu jest ścieżką do pliku któremu ten profil należy ze zamianą symbolów „/” na „.”. Na przykład: plik o nazwie „usr.bin.man” jest profilem aplikacji umaszczonej w systemu pod adresem ***usr/bin/man***

Tryby pracy profilu AppArmore:

- **Enforce** – AppArmore gwarantuje spełnienie warunków opisanych w pliku profilu, wszystkie próby uzyskania dostępu poza opisanymi w profilu regalami zostaną zablokowane.
- **Complain** – Tryb nauczenia. AppArmor tylko loguje próby nielegalnego dostępu, ale pozwala dostęp.

Zobacz które profili AppArmor w którym trybie pracują:

```
#aa-status
```

Wiersz „profiles are loaded” mówi o tym ile profilów istnieją w AppArmor, na przykład:
**20 profiles are loaded** - 20 profile istnieją

Wiersz „profiles are in enforce mode” mówi o tym ile profilów AppArmor pracują w trybie **Enforce**, na przykład:
**14 profiles are in enforce mode** - 14 profile są w trybie Enforce.
Poniżej system wyświetla listę programów dla których profili są uruchomione w trybie **Enforce**

Wiersz „profiles are in complain mode” mówi o tym ile profilów AppArmor pracują w trybie **Complain**, na przykład:
**6 profiles are in complain mode**. - 6 profile są w trybie Complain
Poniżej system wyświetla listę programów dla których profili są uruchomione w trybie **Complain**

Skorzystamy program ***/bin/dir*** jako przykładowy program dla którego zbudujemy profil AppArmor. ***/bin/dir*** jest programem analogicznym ***ls*** - wyświetla zawartość katalogu. Ograniczymy dostęp programu ***/bin/dir*** wyłącznie domowym katalogiem i katalogiem ***Desktop/***

Najpierw sprawdzimy czy ma ***/bin/dir*** dostęp bez uruchomionego profilu AppArmor do dowolnego katalogu. Otwór nowy terminal i uruchom:
```
$/bin/dir
$/bin/dir Desktop/
$/bin/dir Pictures/
```

Dla budowania profilu zastosujemy polecenia ***aa-genprof*** z pakietu ***apparmor-utils***. To polecenie pozwala automatycznie generować profili AppArmor i nauczyć system wymaganemu przez aplikację dostępu. Standardowy algorytm uczenia:

1. Uruchomić ***aa-genprof*** dla programu.
2. Uruchomić programu.
3. Skorzystać ze wszystkich wymaganych funkcyj programu.
4. Wyłączyć programu.
5. Restartować programu.
6. Wyłączyć programu.
7. Odpowiedzieć na pytania ***aa-genprof*** o uprawieniach dostępu dla programu

Uruchom ***aa-genprof*** dla programu:
```
#aa-genprof dir
```
Program ***aa-genprof*** założy profil dla programu ***/bin/dir*** i ruchomi go w trybie ***Complain***.

W innym terminalu uruchom ***/bin/dir*** dla wyświetlenia zawartości domowego katalogu i katalogu ***Desktop/***:
```
$/bin/dir
$/bin/dir Desktop/
```

W terminalu z pracującym ***aa-genprof*** wybierz opcję „S”: (S)can system log for AppArmor events
***aa-genprof*** przeskanuje logi i wykryje do których plików i katalogów badana aplikacją miała dostęp w trakcie uczenia. Za tym, ***aa-genprof*** zapyta o każdym takim przypadku (co pozwalać a co odmówić)
Zezwól wszystkie wykryte przez ***aa-genprof*** próby dostępu (wybieraj: (I)nherit lub (A)llow).
Za tym, wybierz „F”: (F)inish.

W innym terminalu znowu uruchom ***/bin/dir*** dla wyświetlenia zawartości domowego katalogu i katalogów ***Desktop/*** , ***Downloads/***, ***Pictures/*** (Zademonstruj wyniki prowadzącemu):

```
$/bin/dir
$/bin/dir Desktop/
$/bin/dir Downloads/
$/bin/dir Pictures/
```

Uruchom ***/bin/dir*** dla wyświetlenia zawartości katalogu ***Pictures/*** z prawami ***root***:

```
#/bin/dir Pictures/
```

Jak jest możliwe, że ***root***, który ma nieograniczony dostęp do wszystkich zasobów w systemie nie ma dostępu do katalogu ***Pictures/***?

Usuń profil programu ***/bin/dir***

```
#rm /etc/apparmor.d/bin.dir
```

Zaprezentuj prowadzącemu zawartość katalogu ***/etc/apparmor.d/***