

## T: Konfiguracja usługi SSH (SECURE SHELL) w systemie Linux.

### Zadanie1:

Odszukaj w serwisie internetowym Wikipedii informacje na temat usługi SSH.

SSH (ang. secure shell) to standard protokołów komunikacyjnych używanych w sieciach komputerowych w architekturze klient-serwer. SSH służący do terminalowego łączenia się ze zdalnymi komputerami. Transfer wszelkich danych jest zaszyfrowany oraz możliwe jest rozpoznawanie użytkownika na wiele sposobów. Protokoły z rodziny SSH korzystają zwykle z portu 22 protokołu TCP.

### Zadanie2:

Sprawdź dostępną w systemie pomoc na temat klienta ssh (`man ssh`).

W celu odzyskania informacji o zainstalowanych programach usługi ftp należy uruchomić YaST i przejść do Zarządzania oprogramowaniem lub w konsoli tekstowej wydać polecenie:

```
rpm -qa | grep ssh
```

Do połączenia się ze zdalnym serwerem możemy wykorzystać klienta konsoli tekstowej wydając polecenie:

```
ssh serwer.domena.pl
ssh 127.0.0.1
ssh -l username servername
ssh username@servername -p 22
```

Połączenie możemy również realizować przy pomocy nakładki Midnight Commander wybierając z menu Lewy/Prawy => Połączenie po powłoce i wpisując:

```
/#sh:username@hostname/etc/sysconfig
username@hostname
```

Do pobierania plików poprzez usługę ssh możemy wykorzystać polecenie konsoli tekstowej:

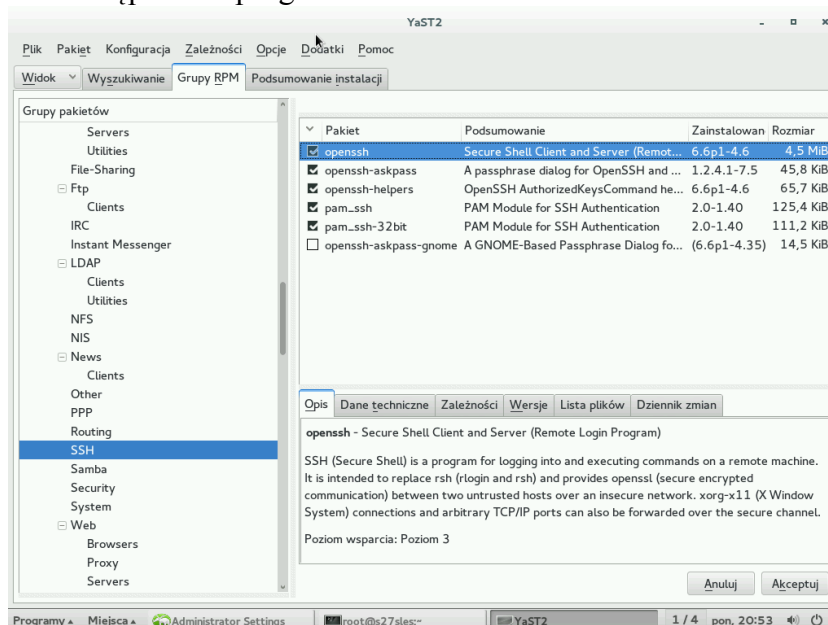
```
scp username@ip_server:/path/filename /local_path/filename
```

Do wysyłania plików poprzez usługę ssh możemy wykorzystać polecenie konsoli tekstowej:

```
scp /local_path/filename username@ip_server:/path/filename
```

### Zadanie3:

Sprawdź przy użyciu konsoli tekstowej dostępność oprogramowania ssh w systemie Linux, a w razie potrzeby przeprowadź instalację pakietu, np. `openssh-server`. Uruchom centrum sterowania YaST w celu sprawdzenia dostępności oprogramowania serwera ssh.



W celu sprawdzenia w konsoli tekstowej, czy zainstalowane jest oprogramowanie ssh należy wydać polecenie:

```
rpm -qa | grep ssh
dpkg -l | grep ssh
```

Instalację usługi ssh w systemie Linux Ubuntu przeprowadzimy za pomocą konsoli tekstowej wydając polecenie:

```
apt-get update
apt-get install openssh-server
apt-get install openssh-client #domyślnie jest zainstalwoany
```

Usługa po instalacji jest automatycznie uruchamiana na porcie 22 tcp. Stan portów w lokalnym komputerze sprawdzimy poleceniem:

```
netstat -ant | grep :22
```

Dodatkowe polecenia konsoli tekstowej:

```
chkconfig sshd on
iptables -L | grep ssh #lub numer portu 22
man sshd
cat /etc/hosts.allow
cat /etc/hosts.deny
    sshd : all except s27nau
~/.ssh/
/etc/init.d/sshd status
/etc/init.d/sshd start
service ssh restart #sshd
initctl reload-configuration #wykonać, gdy występuje błąd Unknown job:ssh
```

Pliki konfiguracyjne serwera (sshd\_config) i klienta (ssh\_config) ssh znajdują się w katalogu /etc/ssh. W celu zapoznania się z dostępnymi opcjami konfiguracji serwera możemy w przeglądarce Konqueror w polu adresu wpisać #sshd\_config lub w konsoli tekstowej podać polecenie man sshd\_config.

**Zadanie5:**

Przeanalizuj zawartość pliku /etc/ssh/sshd\_config.

Generowanie kluczy publicznego i prywatnego na potrzeby demona sshd:

```
ssh-keygen -t rsa (ssh_host_rsa_key - nazwa pliku)
ssh-keygen -t dsa (ssh_host_dsa_key - nazwa pliku)
ssh-keygen -t rsa1 (ssh_host_key - nazwa pliku)
```

Uzyskanie informacji na temat odcisku palca (fingerprint) klucza:

```
ssh-keygen -l -f /etc/ssh/ssh_host_rsa_key
```

Sekwencja poleceń dokonujących podpisania klucza user@poczta.pl oraz wyświetlający informacje o odcisku (należy wydać w katalogu /home/user/.ssh):

```
gpg --edit-key user@poczta.pl
Polecenie> sign
Polecenie> check
Polecenie> save
Polecenie> quit
```

W celu zablokowania usługi ssh dla wszystkich komputerów za wyjątkiem komputera nauczyciela należy w pliku /etc/hosts.deny dokonać następującego wpisu:

```
sshd : all except s27nau
sshd : all except 192.168.19.35
sshd : lamer.pl
ssh : all
rsh : all
```

**Przykładowe opcje konfiguracji serwera sshd w pliku /etc/ssh/sshd\_config:**

```
# $OpenBSD: sshd_config,v 1.56 2002/06/20 23:37:12 markus Exp $
#####
# Jest to plik konfiguracyjny serwera sshd.
# Przeglądaj sshd_config(5) dla uzyskania dodatkowych informacji.
# Ten sshd został skompilowany ze ścieżką:
# PATH=/usr/local/bin:/bin:/usr/bin:/usr/X11R6/bin
#
# UWAGA: po zainstalowaniu sshd popraw plik /etc/hosts.allow
# (sshd: IPklienta oraz w nowym wierszu ssh: IPklienta)
# Pamiętaj o potrzebie dopuszczenia sshd w firewallu
#####
# Nr portu na którym nasłuchuje Twój serwer (demon sshd)
Port 22
# Dostępne protokoły: ssh2 i ssh1
Protocol 2,1
# Na jakim IP Twojego serwera będzie nasłuchiwać sshd
# (ważne w przypadku kilku kart sieciowych na serwerze)
# TU nasłuchuje na wszystkich dostępnych adresach serwera
# czyli wszystkich kartach sieciowych i ew. modemie.
ListenAddress 0.0.0.0
#ListenAddress ::
#
# Klucz HostKey dla protokołu version 1
HostKey /etc/ssh/ssh_host_key
# Klucz HostKeys dla protokołu version 2
HostKey /etc/ssh/ssh_host_rsa_key
HostKey /etc/ssh/ssh_host_dsa_key
#
# Długość życia (w sekundach) klucza "version 1 server key"
KeyRegenerationInterval 3600
# Długość klucza (w bitach) klucza "version 1 server key"
ServerKeyBits 768

# Logging
# obsoletes (przestarzały) QuietMode (cichy tryb)
# and FascistLogging
# SyslogFacility czyli SyslogUdogodnienie
#SyslogFacility AUTH
#LogLevel INFO
#
#####
# Parametry autentykacji. (Authentication)
#####
#
# Czas oczekiwania (w sekundach)
LoginGraceTime 600
# Czy można logować się zdalnie na konto roota.
# Wpisz "no" i jak root loguj się poprzez konto
# zwykłego użytkownika i komendy su lub su -l
PermitRootLogin no
StrictModes yes
#####
# Czy zgadzasz się na autentykację RSA ? (TAK!!!)
# Klucz RSA można użyć zamiast lub równocześnie z hasłem.
# Zaraz wybierzesz właściwe opcje.
# Teraz chwila wyjaśnienia: Należy odróżnić ssh (narzędzie klienckie)
# i demona sshd (czyli serwer).
# Tutaj konfigurujemy co prawda sshd, ale pamiętać należy, że odległy
# klient też musi prawidłowo się przygotować.
# Do poprawnej pracy OPENSSH musimy dokonać konfiguracji demona
# sshd oraz klienta ssh.
# Jeżeli zdecydowaliśmy się na używanie ssh z kluczami RSA (zamiast
# lub równocześnie z hasłami), każdy user (czyli Ty oraz twoi kumple)
# przed użyciem powinien wygenerować swoją własną parę kluczy komendą:
# $ ssh-keygen (w Mandrake może być zrobiony automatycznie).
# W takiej chwili będziesz musiał podać tzw. paszport (zapisz sobie na kartce)
# W podfolderze /home/antek/.ssh zostaną utworzone dwa pliki:
```

```
# Identity - ktory zawiera PRYWATNY KLUCZ i nie powinien byc udostepniany
# nikomu (pamietaj o restrykcyjnych prawach do tego pliku)
# identity.pub - czyli klucz publiczny.
# Jezeli zamierzamy jako klient pracowac przez ssh na odleglym serwerze,
# to zawartosc tego klucza powinniśmy skopiowac do katalogu odleglego
# uzytkownika, nza ktorego konto logujemy sie - do pliku np.
# /home/antek/.ssh/ w pliku authorized_keys.
# Oczywiscie wowczas proby logowania na odlegly serwer musza
# byc podejmowane na konto antek i uwaga: wskazane jest na poczatku
# aby logowac sie z konta antek na konto tez antek.
RSAAuthentication yes
PubkeyAuthentication yes
AuthorizedKeysFile .ssh/authorized_keys
#
# Autentykacja rhosts (czyli dopuszczanie do logowania wg listy tzw.
# "zaufanych" maszyn zamiast zmuszania ich do podawania hasel)
# nie powinna byc uzywana ze wzgledow bezpieczenstwa.
# Wystarczy bowiem, ze ktos sie wlamie i dobierze do plikow
# ~/.rhosts .shosts...
# JEZELI WYBRALES "no" (a to polecam) TO MOZESZ ZAHASZOWAC KILKA
# KOLEJNYCH OPCJI, gdz nie maja one w takim razie znaczenia.
# Wszak zrezygnowales z tego rodzaju autentykacji.
RhostsAuthentication no
#
# Ignorowanie plikow ~/.rhosts i ~/.shosts, ktore maja wykaz "zaufanych"
# stacji, z ktorych odlegly uzytkownik moze sie zalogowac bez podania hasla.
# Oczywiscie wpisz "yes" i nie pozwol (ze wzgledu bezpieczenstwa) na
# czytanie plikow ~/.rhosts i ~/.shosts w zastepstwie autoryzowania haslem.
IgnoreRhosts yes
#
# Jezeli w autentykacji rhosts wybrales opcje "RhostsAuthentication yes"
# to mozesz zmusic klientow zapisanych w w/w plikach ~/.rhosts i ~/.shosts by
# dodatkowo legitymowaly sie kluczem "host keys" znajdujacym
# sie w /etc/ssh/ssh_known_hosts (pamietasz jak opisalem koniecznosc
# skonfigurowania ssh u odleglego klienta poprzez wygenerowanie
# kluczy komenda: [ $ ssh-keygenktory ] ) ?
#RhostsRSAAuthentication no
# i podobnie j.w. ale wobec protokolu version 2
#HostbasedAuthentication no
#
# Zmien ponizsze na NIE (!!!) jezeli nie masz zaufania do klienckich
# kluczy zapisanych w ~/.ssh/known_hosts (w autentykacji
# RhostsRSAAuthentication i HostbasedAuthentication).
# Czy serwer ma ignorowac nadzor nad komputerami w/w uzytkownikow?
# Inaczej mowiac - czy serwer ma zadowolic sie tescia kluczy od
# klientow zapisanych w ~/.ssh/known_hosts ?
IgnoreUserKnownHosts no
#
# Aby wylaczyc tunelowanie (co jest fatalnym pomyslem) - wyczysc wpisy
# hasel i zmien na "no"
PasswordAuthentication yes
# Zezwolenie na puste hasla
PermitEmptyPasswords no
#
# NIE WIEM O CO TUTAJ CHODZI.
# Uncomment to disable s/key passwords czyli
# Wyhaszuj po to, aby wylaczyc s/klucz hasel
#ChallengeResponseAuthentication no
#
# Kerberos options
#KerberosAuthentication no
#KerberosOrLocalPasswd yes
#KerberosTicketCleanup yes
#AFSTokenPassing no
# Kerberos TGT Passing only works with the AFS kserver
#KerberosTgtPassing no
#
# Set this to 'yes' to enable PAM keyboard-interactive authentication
```

```
# Warning: enabling this may bypass the setting of 'PasswordAuthentication'
#PAMAuthenticationViaKbdInt yes
#
# Szyfrowanie przez ssh polaczenia graficznego za pomoca X-Window
X11Forwarding yes
X11DisplayOffset 10
X11UseLocalhost yes
#
# Pojawi sie komunikat powitalny o tresci pobranej z pliku /etc/motd
PrintMotd yes
#
# Pojawi sie komunikat powitalny o tresci z pliku tekstowego /etc/issue
#Banner /etc/issue
# Polecam jednak utworzyc inny plik np. /etc/ssh/banner z jakims tekstem
# Oczywiscie nalezy wowczas wpisac ponizej odpowiednia sciezke dostepu
Banner /etc/ssh/baner
#
# Pojawi sie informacja z data ostatniego logowania
PrintLastLog yes
#
KeepAlive yes
#
#UseLogin no
#
#Wiele kodów w OpenSSH które działały wyłącznie pod rootem, obecnie funkcjonują pod
nieuprzywilejowanym użytkownikiem. Ponieważ znacząco podnosi to bezpieczeństwo
OpenSSH, powinno się udostępnić cechę UsePrivilegeSeparation . Niestety, opcja ta
nie działa zbyt dobrze (w wersji OpenSSH 3.4p1) z innymi systemami unixowymi, można
jednak się spodziewać, że następna wersja OpenSSH będzie pozbawiona błędów
UsePrivilegeSeparation yes
#
# Zezwolenie na kompresje danych podczas połączenia
Compression yes
#
MaxStartups 10
#
# Sprawdzanie zgodności nazwy odległego klienta
# (pełnej domeny) z IP. Domyślnie niedostępne.
#VerifyReverseMapping no
#ReverseMappingCheck yes
#
#CheckMail yes
#
#UseLogin no
#
# Zezwolenie na szyfrowane polaczenie sftp.
Subsystem sftp /usr/lib/ssh/sftp-server
```

**Konfiguracja klienta SSH** zapisana jest w pliku `/etc/ssh/ssh_config`. Opcje konfiguracyjne sprawdzane są w następującej kolejności:

- opcje podane w linii komend,
- plik konfiguracyjny użytkownika (`$HOME/.ssh/config`),
- plik ogólnosystemowy.

### Opis ustawień w pliku konfiguracyjnym klienta ssh (`/etc/ssh/ssh_config`)

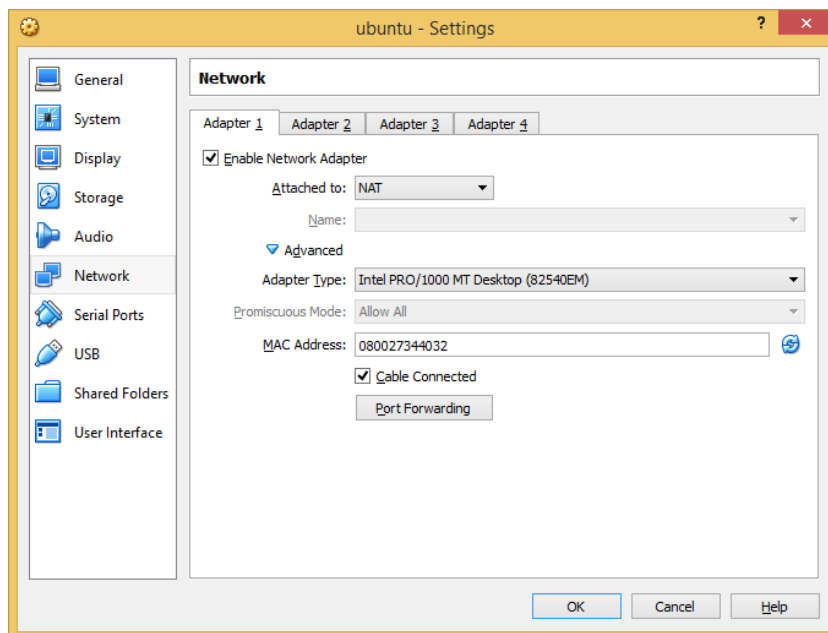
`Hosts *` - otwiera sekcję dotyczącą połączeń do danego hosta - \* oznacza wszystkie hosty,  
`ForwardAgent yes` - określa, czy agent autentykacyjny ma być przekazywany na kolejne systemy  
na które następuje logowanie,  
`ForwardX11 yes` - zezwala na automatyczne przekazywanie połączeń X11 ponad szyfrowanym  
kanałem SSH,  
`RhostsAuthentication no`  
`RhostsRSAAuthentication no` - zezwalanie na autentykację za pomocą mechanizmu rhosts,  
`PasswordAuthentication yes` - autentykacja za pomocą haseł,  
`RSAAuthentication yes`

TISAuthentication no - wybór metody autentykacji (wybrać tylko RSA),  
PasswordPromptHost yes  
PasswordPromptLogin yes - czy program ma pytać o hasła,  
FallbackToRsh no  
UseRsh no - możliwość użycia rsh w przypadku niepowodzenia połączenia za pomocą ssh. Można włączać, ale administrator zdalnej maszyny prawie na pewno to wyłączył,  
BatchMode no - możliwość użycia ssh w trybie wsadowym,  
EscapeChar ~ - jaki znak powoduje wyjście z połączenia (jak w telnetcie **ctrl+]**),  
Cipher 3DES - algorytm stosowany do szyfrowania przy połączeniu ze zdalną maszyną,  
Compression yes - czy włączona jest kompresja,  
CompressionLevel 9 - poziom kompresji, 0 - wyłącza,  
IdentityFile ~/.ssh/identity - położenie i nazwa pliku identyfikacji.

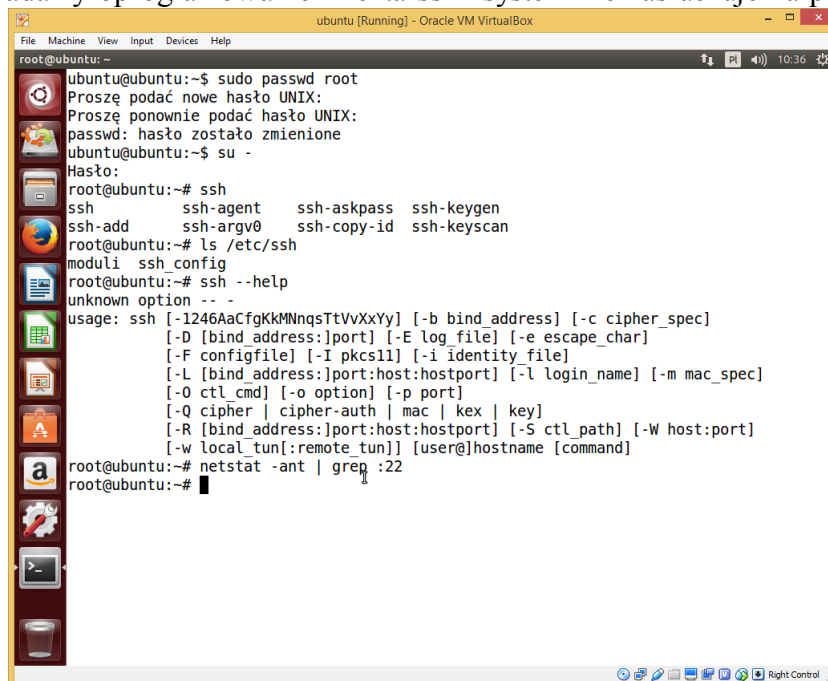
#### Zadanie6:

Wykorzystując dowolny procesor tekstu wykonaj sprawozdanie na temat instalacji, konfiguracji i sprawdzenia poprawności działania usługi SSH w systemie Linux Ubuntu. Na jednej stronie o rozmiarze A4, orientacji pionowej, umieść dwa opisane zrzuty z ekranu, o szerokości minimum 14 cm. Każdą stronę podpisz swoim imieniem i nazwiskiem w nagłówku strony a w stopce oznacz numer strony wg schematu Strona X z Y. Pracę zachowaj w pliku pod nazwą **\$nazwisko\_ssh** i prześlij pocztą elektroniczną do nauczyciela na adres [greszata@zs9elektronik.pl](mailto:greszata@zs9elektronik.pl).

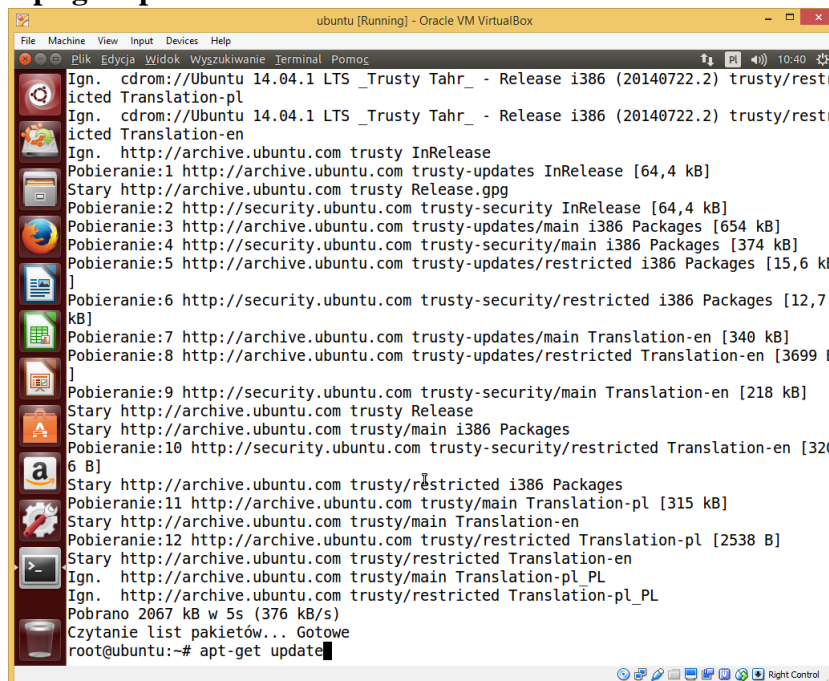
Przed przystąpieniem do instalacji usługi SSH należy sprawdzić, czy w maszynie wirtualnej systemu Linux włączyliśmy kartę sieciową i odpowiednio ją skonfigurowaliśmy. I tak dla maszyny wirtualnej **ubuntu** wybieramy ustawienia (**Settings**), przechodzimy do kategorii **Network** i w zakładce **Adapter 1** kartę podłączamy w trybie **NAT**.



W systemie operacyjnym Linux Ubuntu uruchamiamy konsolę terminala. Domyślnie w systemie Linux Live Ubuntu zainstalowane jest oprogramowanie klienta ssh (openssh-client). Uprawnienia do zarządzania systemem posiada administrator systemu, dlatego należy w oknie podać hasło użytkownika root. W oknie terminala upewniamy się, że posiadamy oprogramowanie klienta ssh i system nie nasłuchuje na porcie 22 protokołu TCP:

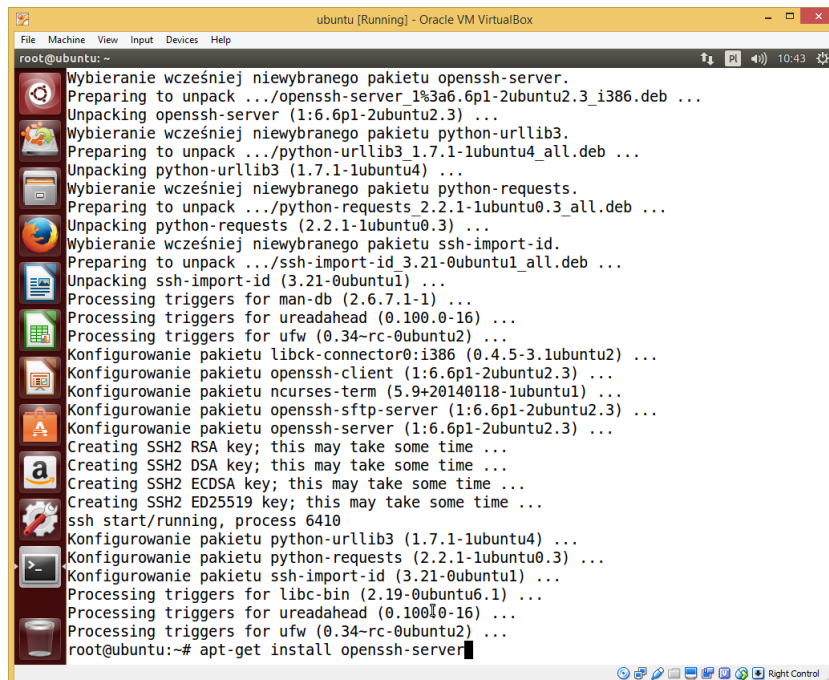


Przed instalacją niezbędnego oprogramowania aktualizujemy adresy serwerów źródłowych i listy dostępnych programów poleceniem **apt-get update**:



```
ubuntu [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Plik Edycja Widok Wyszukiwanie Terminal Pomoc
Ign. cdrom://Ubuntu 14.04.1 LTS _Trusty Tahr_ - Release i386 (20140722.2) trusty/restr
icted Translation-pl
Ign. cdrom://Ubuntu 14.04.1 LTS _Trusty Tahr_ - Release i386 (20140722.2) trusty/restr
icted Translation-en
Ign. http://archive.ubuntu.com trusty InRelease
Pobieranie:1 http://archive.ubuntu.com trusty-updates InRelease [64,4 kB]
Stary http://archive.ubuntu.com trusty Release.gpg
Pobieranie:2 http://security.ubuntu.com trusty-security InRelease [64,4 kB]
Pobieranie:3 http://archive.ubuntu.com trusty-updates/main i386 Packages [654 kB]
Pobieranie:4 http://security.ubuntu.com trusty-security/main i386 Packages [374 kB]
Pobieranie:5 http://archive.ubuntu.com trusty-updates/restricted i386 Packages [15,6 kB
]
Pobieranie:6 http://security.ubuntu.com trusty-security/restricted i386 Packages [12,7
kB]
Pobieranie:7 http://archive.ubuntu.com trusty-updates/main Translation-en [340 kB]
Pobieranie:8 http://archive.ubuntu.com trusty-updates/restricted Translation-en [3699 B
]
Pobieranie:9 http://security.ubuntu.com trusty-security/main Translation-en [218 kB]
Stary http://archive.ubuntu.com trusty Release
Stary http://archive.ubuntu.com trusty/main i386 Packages
Pobieranie:10 http://security.ubuntu.com trusty-security/restricted Translation-en [320
6 B]
Stary http://archive.ubuntu.com trusty/restricted i386 Packages
Pobieranie:11 http://archive.ubuntu.com trusty/main Translation-pl [315 kB]
Stary http://archive.ubuntu.com trusty/main Translation-en
Pobieranie:12 http://archive.ubuntu.com trusty/restricted Translation-pl [2538 B]
Stary http://archive.ubuntu.com trusty/restricted Translation-en
Ign. http://archive.ubuntu.com trusty/main Translation-pl_PL
Ign. http://archive.ubuntu.com trusty/restricted Translation-pl_PL
Pobrano 2067 kB w 5s (376 kB/s)
Czytanie list pakietów... Gotowe
root@ubuntu:~# apt-get update
```

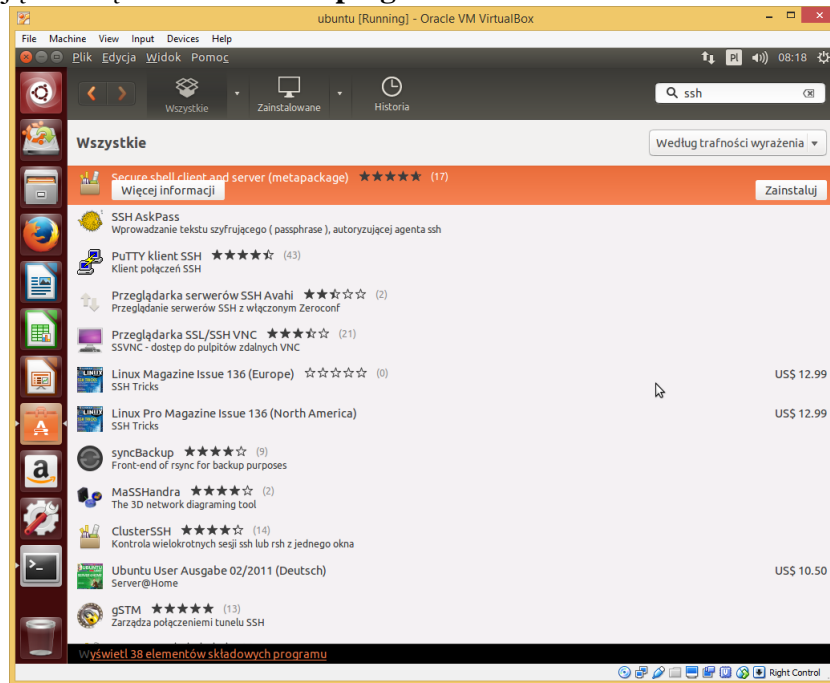
Następnie przeprowadzamy instalację oprogramowania serwera ssh wydając w konsoli polecenie **apt-get install openssh-server**:



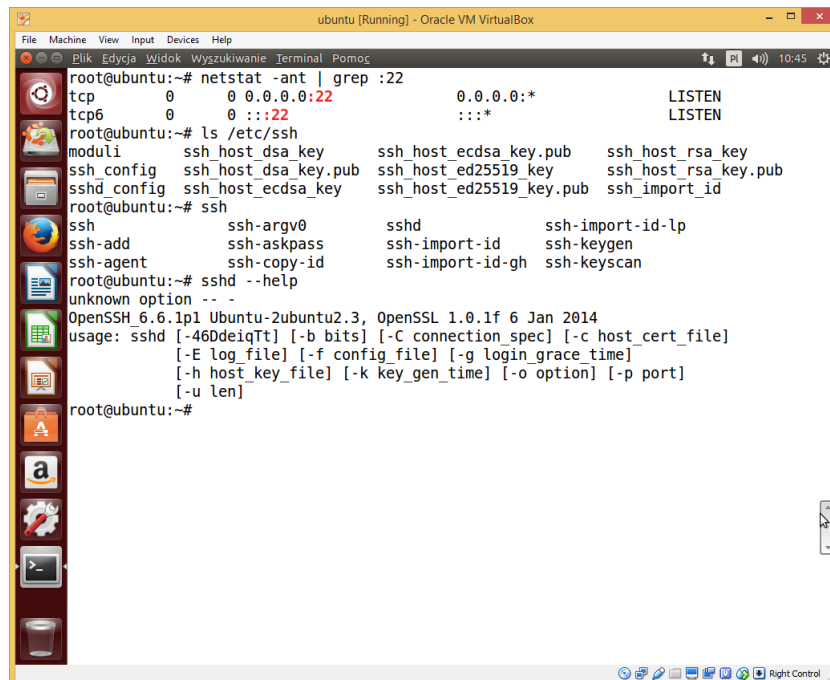
```
ubuntu [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
root@ubuntu:~#
Wybieranie wcześniej niewybranego pakietu openssh-server.
Preparing to unpack ../openssh-server 1%3a6.6p1-2ubuntu2.3_i386.deb ...
Unpacking openssh-server (1:6.6p1-2ubuntu2.3) ...
Wybieranie wcześniej niewybranego pakietu python-urllib3.
Preparing to unpack ../python-urllib3 1.7.1-1ubuntu4_all.deb ...
Unpacking python-urllib3 (1.7.1-1ubuntu4) ...
Wybieranie wcześniej niewybranego pakietu python-requests.
Preparing to unpack ../python-requests 2.2.1-1ubuntu0.3_all.deb ...
Unpacking python-requests (2.2.1-1ubuntu0.3) ...
Wybieranie wcześniej niewybranego pakietu ssh-import-id.
Preparing to unpack ../ssh-import-id 3.21-0ubuntu1_all.deb ...
Unpacking ssh-import-id (3.21-0ubuntu1) ...
Processing triggers for man-db (2.6.7.1-1) ...
Processing triggers for ureadahead (0.100.0-16) ...
Processing triggers for ufw (0.34-rc-0ubuntu2) ...
Konfigurowanie pakietu libck-connector0:i386 (0.4.5-3.1ubuntu2) ...
Konfigurowanie pakietu openssh-client (1:6.6p1-2ubuntu2.3) ...
Konfigurowanie pakietu ncurses-term (5.9+20140118-1ubuntu1) ...
Konfigurowanie pakietu openssh-sftp-server (1:6.6p1-2ubuntu2.3) ...
Konfigurowanie pakietu openssh-server (1:6.6p1-2ubuntu2.3) ...
Creating SSH2 RSA key; this may take some time ...
Creating SSH2 DSA key; this may take some time ...
Creating SSH2 ECDSA key; this may take some time ...
Creating SSH2 ED25519 key; this may take some time ...
ssh start/running, process 6410
Konfigurowanie pakietu python-urllib3 (1.7.1-1ubuntu4) ...
Konfigurowanie pakietu python-requests (2.2.1-1ubuntu0.3) ...
Konfigurowanie pakietu ssh-import-id (3.21-0ubuntu1) ...
Processing triggers for libc-bin (2.19-0ubuntu6.1) ...
Processing triggers for ureadahead (0.100.0-16) ...
Processing triggers for ufw (0.34-rc-0ubuntu2) ...
root@ubuntu:~# apt-get install openssh-server
```



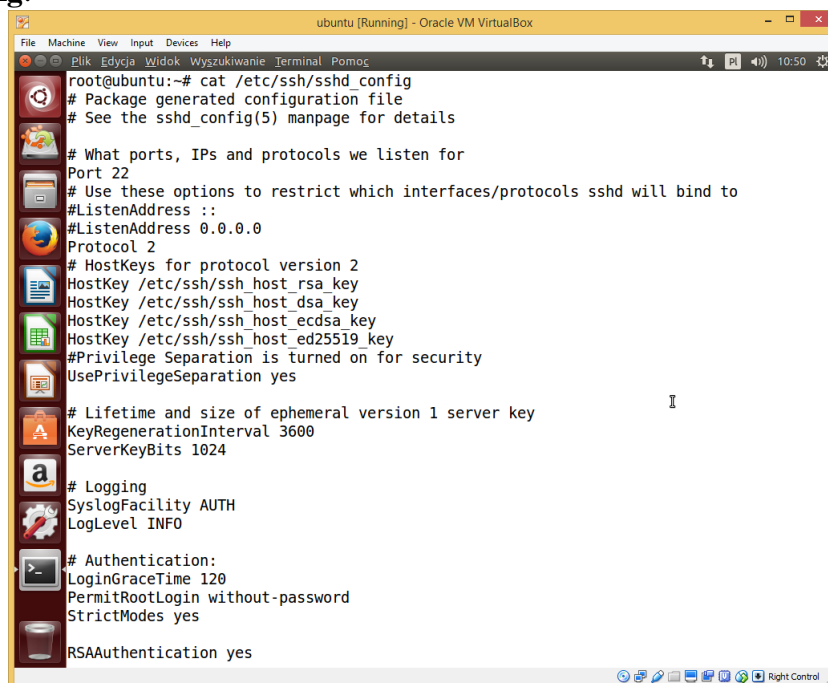
W systemie Linux Ubuntu możemy dokonać instalacji niezbędnego oprogramowania również w trybie graficznym wykorzystując narzędzie **Centrum oprogramowania Ubuntu**:



Po instalacji serwera ssh sprawdzamy bieżący stan usługi ssh (service sshd status) oraz zmiany programowe dokonane w systemie:



Sprawdzenie zawartości pliku konfiguracyjnego usługi ssh dokonamy wydając polecenie `cat /etc/ssh/sshd_config`:



```
ubuntu [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Plik Edycja Widok Wyszukiwanie Terminal Pomoc
root@ubuntu:~# cat /etc/ssh/sshd_config
# Package generated configuration file
# See the sshd_config(5) manpage for details

# What ports, IPs and protocols we listen for
Port 22
# Use these options to restrict which interfaces/protocols sshd will bind to
#ListenAddress ::
#ListenAddress 0.0.0.0
Protocol 2
# HostKeys for protocol version 2
HostKey /etc/ssh/ssh_host_rsa_key
HostKey /etc/ssh/ssh_host_dsa_key
HostKey /etc/ssh/ssh_host_ecdsa_key
HostKey /etc/ssh/ssh_host_ed25519_key
#Privilege Separation is turned on for security
UsePrivilegeSeparation yes

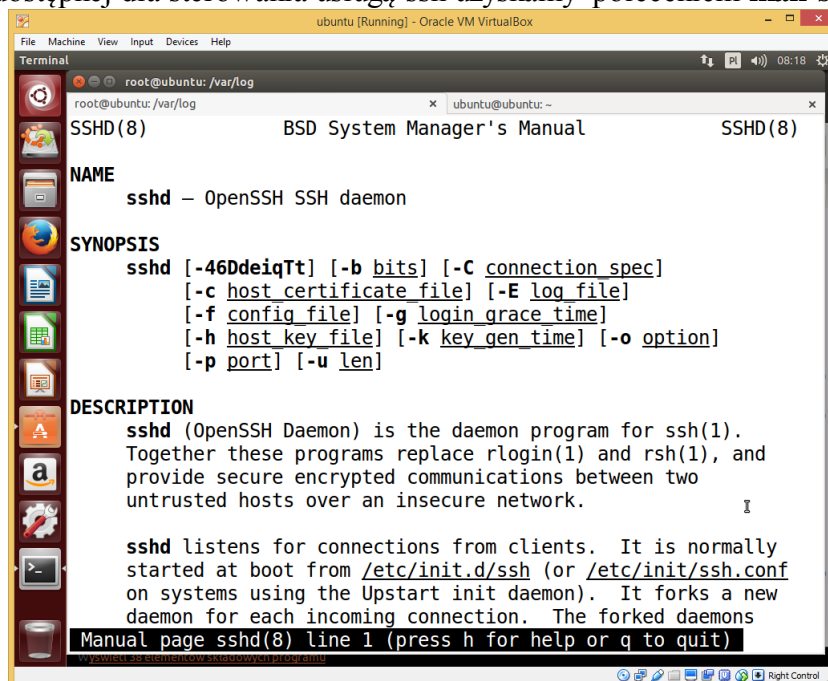
# Lifetime and size of ephemeral version 1 server key
KeyRegenerationInterval 3600
ServerKeyBits 1024

# Logging
SyslogFacility AUTH
LogLevel INFO

# Authentication:
LoginGraceTime 120
PermitRootLogin without-password
StrictModes yes

RSAAuthentication yes
```

Sprawdzenie pomocy dostępnej dla sterowania usługą ssh uzyskamy poleceniem `man sshd`:



```
ubuntu [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Terminal
root@ubuntu:~/var/log
root@ubuntu:~/var/log x ubuntu@ubuntu: ~
SSHD(8) BSD System Manager's Manual SSHD(8)

NAME
  sshd – OpenSSH SSH daemon

SYNOPSIS
  sshd [-46DdeiqTt] [-b bits] [-C connection spec]
        [-c host_certificate file] [-E log file]
        [-f config file] [-g login_grace_time]
        [-h host_key file] [-k key_gen_time] [-o option]
        [-p port] [-u len]

DESCRIPTION
  sshd (OpenSSH Daemon) is the daemon program for ssh(1).
  Together these programs replace rlogin(1) and rsh(1), and
  provide secure encrypted communications between two
  untrusted hosts over an insecure network.

  sshd listens for connections from clients. It is normally
  started at boot from /etc/init.d/ssh (or /etc/init/ssh.conf
  on systems using the Upstart init daemon). It forks a new
  daemon for each incoming connection. The forked daemons
  Manual page sshd(8) line 1 (press h for help or q to quit)
```

Restartujemy usługę ssh wydając polecenie **service ssh restart**. Dostęp do usługi możemy kontrolować poprzez pliki **/etc/hosts.allow** oraz **/etc/hosts.deny**. Dodając wpis **sshd : all except localhost** do pliku **/etc/hosts.deny** spowodujemy zablokowanie dostępu do usługi wszystkim komputerom, oprócz systemu lokalnego:

```

root@ubuntu:~#
# If you're going to protect the portmapper use the name "rpcbind" for the
# daemon name. See rpcbind(8) and rpc.mountd(8) for further information.
#
# The PARANOID wildcard matches any host whose name does not match its
# address.
#
# You may wish to enable this to ensure any programs that don't
# validate looked up hostnames still leave understandable logs. In past
# versions of Debian this has been the default.
# ALL: PARANOID
root@ubuntu:~# echo 'sshd : all except localhost' >> /etc/hosts.deny
root@ubuntu:~# cat /etc/hosts.deny
# /etc/hosts.deny: list of hosts that are _not_ allowed to access the system.
# See the manual pages hosts_access(5) and hosts_options(5).
#
# Example:   ALL: some.host.name, .some.domain
#           ALL EXCEPT in.fingerd: other.host.name, .other.domain
#
# If you're going to protect the portmapper use the name "rpcbind" for the
# daemon name. See rpcbind(8) and rpc.mountd(8) for further information.
#
# The PARANOID wildcard matches any host whose name does not match its
# address.
#
# You may wish to enable this to ensure any programs that don't
# validate looked up hostnames still leave understandable logs. In past
# versions of Debian this has been the default.
# ALL: PARANOID
sshd : all except localhost
root@ubuntu:~# echo 'sshd : all except localhost' >> /etc/hosts.deny

```

Połączenie z usługą nawiążemy podając polecenie **ssh -l ubuntu localhost**. Podczas pierwszego logowania należy zaakceptować klucz szyfrowania stosowany do zabezpieczenia połączenia:

```

ubuntu@ubuntu:~#
root@ubuntu:~# passwd ubuntu
Proszę podać nowe hasło UNIX:
Proszę ponownie podać hasło UNIX:
passwd: hasło zostało zmienione
root@ubuntu:~# ssh -l ubuntu localhost
The authenticity of host 'localhost (127.0.0.1)' can't be established.
ECDSA key fingerprint is eb:a0:ab:93:bb:13:f5:53:6a:2e:55:7c:e1:43:af:a1.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'localhost' (ECDSA) to the list of known hosts.
ubuntu@localhost's password:
Welcome to Ubuntu 14.04.1 LTS (GNU/Linux 3.13.0-32-generic i686)

 * Documentation:  https://help.ubuntu.com/

ubuntu@ubuntu:~# w
 20:20:32 up 12 min,  9 users,  load average: 0,10, 0,24, 0,22
USER      TTY      FROM            LOGIN@   IDLE   JCPU   PCPU   WHAT
ubuntu    tty6                         20:08   12:17   0.09s   0.09s  -bash
ubuntu    tty4                         20:08   12:17   0.07s   0.07s  -bash
ubuntu    tty5                         20:08   12:17   0.10s   0.07s  -bash
ubuntu    tty2                         20:08   12:17   0.05s   0.05s  -bash
ubuntu    tty3                         20:08   12:17   0.05s   0.05s  -bash
ubuntu    tty1                         20:08   12:17   0.05s   0.05s  -bash
ubuntu    :0           :0              20:09   ?xdm?   3:52   0.30s  init --user
ubuntu    pts/2        :0              20:10   0.00s   0.19s   3.19s  gnome-terminal
ubuntu    pts/0        localhost       20:20   0.00s   0.05s   0.00s  w
ubuntu@ubuntu:~#

```

W trakcie połączenia z serwerem ssh użytkownik będzie wyświetlony na liście zalogowanych użytkowników w systemie. Listę wywołamy poleceniem **w** lub **who**. Połączenie ssh zakończymy poleceniem **exit**:

```
root@ubuntu:~# ssh -l ubuntu localhost
ubuntu@localhost's password:
Welcome to Ubuntu 14.04.1 LTS (GNU/Linux 3.13.0-32-generic i686)

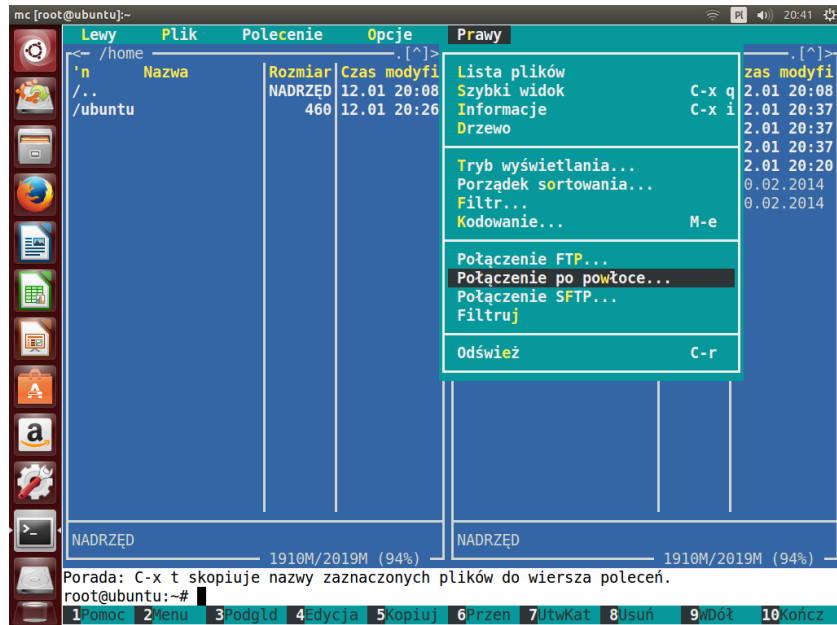
 * Documentation:  https://help.ubuntu.com/

Last login: Tue Jan 12 20:27:22 2016 from localhost
ubuntu@ubuntu:~$ who
ubuntu  tty6      2016-01-12 20:08
ubuntu  tty4      2016-01-12 20:08
ubuntu  tty5      2016-01-12 20:08
ubuntu  tty2      2016-01-12 20:08
ubuntu  tty3      2016-01-12 20:08
ubuntu  tty1      2016-01-12 20:08
ubuntu  :0        2016-01-12 20:09 (:0)
ubuntu  pts/2     2016-01-12 20:10 (:0)
ubuntu  pts/0     2016-01-12 20:27 (localhost)
ubuntu@ubuntu:~$ exit
logout
Connection to localhost closed.
root@ubuntu:~# who
ubuntu  tty6      2016-01-12 20:08
ubuntu  tty4      2016-01-12 20:08
ubuntu  tty5      2016-01-12 20:08
ubuntu  tty2      2016-01-12 20:08
ubuntu  tty3      2016-01-12 20:08
ubuntu  tty1      2016-01-12 20:08
ubuntu  :0        2016-01-12 20:09 (:0)
ubuntu  pts/2     2016-01-12 20:10 (:0)
root@ubuntu:~#
```

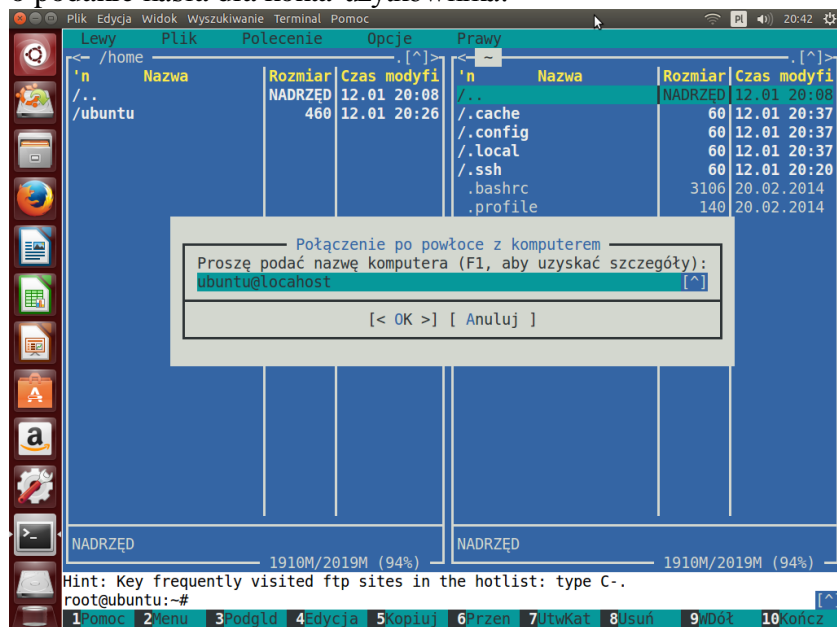
W dowolnym momencie pracy systemu możemy sterować usługą ssh. Wykorzystujemy do tego polecenia **service ssh stop** lub **service ssh start** lub **service ssh restart**. Pamiętajmy, że w przypadku zmian w zawartości pliku konfiguracyjnego serwera ssh należy zrestartować usługę:

```
root@ubuntu:~# netstat -ant
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 127.0.0.1:53            0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:22             0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.1:631         0.0.0.0:*               LISTEN
tcp6       0      0 :::22                  :::*                    LISTEN
tcp6       0      0 :::1:631               :::*                    LISTEN
tcp6       1      0 :::1:41231             :::1:631                CLOSE_WAIT
root@ubuntu:~# service ssh stop
ssh stop/waiting
root@ubuntu:~# netstat -ant
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 127.0.0.1:53            0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.1:631         0.0.0.0:*               LISTEN
tcp6       0      0 :::1:631               :::*                    LISTEN
tcp6       1      0 :::1:41231             :::1:631                CLOSE_WAIT
root@ubuntu:~# service ssh start
ssh start/running, process 6881
root@ubuntu:~# netstat -ant
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 127.0.0.1:53            0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:22             0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.1:631         0.0.0.0:*               LISTEN
tcp6       0      0 :::22                  :::*                    LISTEN
tcp6       0      0 :::1:631               :::*                    LISTEN
tcp6       1      0 :::1:41231             :::1:631                CLOSE_WAIT
root@ubuntu:~#
```

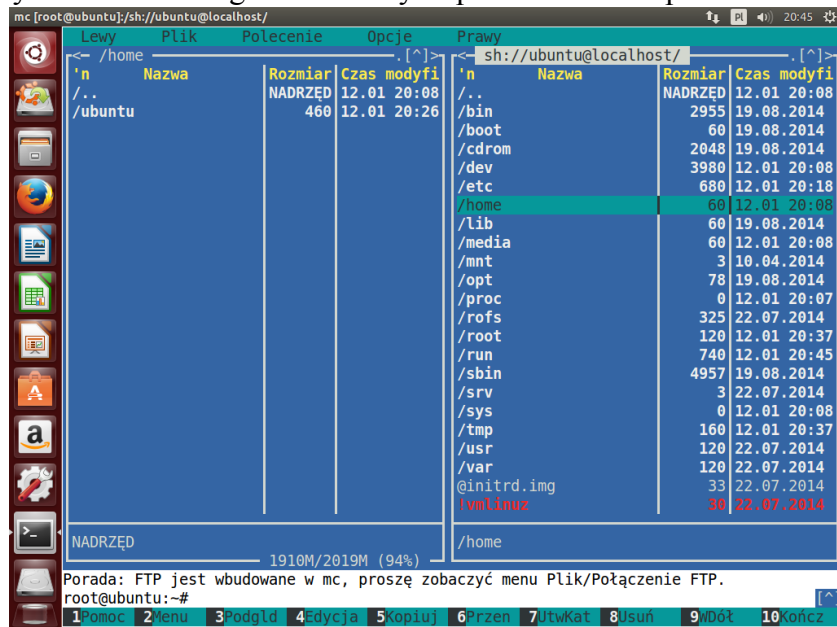
Z usługą ssh możemy również połączyć się poprzez menedżera plików Midnight Commander. Po uruchomieniu programu poleceniem **mc** wybieramy z menu górnego (**F9**) w kategorii **Lewy/Prawy** polecenie **Połączenie po powłoce...**:



W wyświetlonym oknie podajemy nazwę użytkownika i adres serwera ssh i po kliknięciu przycisku **OK** zostaniemy poproszeni o podanie hasła dla konta użytkownika:



Pozostaje już tylko korzystać z ułatwień gwarantowanych przez menedżer plików MC:



mc [root@ubuntu]:sh://ubuntu@localhost/ 20:45

Lewy				Prawy			
	Nazwa	Rozmiar	Czas modyfi		Nazwa	Rozmiar	Czas modyfi
..	NADRZĘD	12.01	20:08	..	NADRZĘD	12.01	20:08
ubuntu	460	12.01	20:26	bin	2955	19.08.2014	
				boot	60	19.08.2014	
				cdrom	2048	19.08.2014	
				dev	3980	12.01.20:08	
				etc	680	12.01.20:18	
				home	60	12.01.20:08	
				lib	60	19.08.2014	
				media	60	12.01.20:08	
				mnt	3	10.04.2014	
				opt	78	19.08.2014	
				proc	0	12.01.20:07	
				rofs	325	22.07.2014	
				root	120	12.01.20:37	
				run	740	12.01.20:45	
				sbin	4957	19.08.2014	
				srv	3	22.07.2014	
				sys	0	12.01.20:08	
				tmp	160	12.01.20:37	
				usr	120	22.07.2014	
				var	120	22.07.2014	
				@initrd.img	33	22.07.2014	
				linux	30	22.07.2014	

NADRZĘD 1910M/2019M (94%)

Porada: FTP jest wbudowane w mc, proszę zobaczyć menu Plik/Połączenie FTP.  
root@ubuntu:~#

1 Pomoc 2 Menu 3 Podgląd 4 edycja 5 kopiuj 6 wklej 7 Utwórz 8 Usuń 9 Dół 10 Kończ