**Obraz zawierający tekst, Czcionka, logo, wizytówka

Opis wygenerowany automatycznie**

**Systemy operacyjne 2**

*Korzystanie z zasobów systemu Linux na systemie Windows*

Prowadzący: Autor:

Dr Krzysztof Balicki Wojciech Olejko

Kierunek: Informatyka, grupa Lab 01

Rzeszów r.a. 2023/2024

Spis treści

[1. Wstęp 3](#_Toc155294232)

[2. WSL 2 3](#_Toc155294233)

[2.1 Konfiguracja 3](#_Toc155294234)

[2.2 Użytkowanie 4](#_Toc155294235)

[3. SSH 5](#_Toc155294236)

[3.1 Konfiguracja 5](#_Toc155294237)

[3.2 Użytkowanie 7](#_Toc155294238)

[4. Samba 8](#_Toc155294239)

[4.1 Konfiguracja 8](#_Toc155294240)

[4.2 Użytkowanie 10](#_Toc155294241)

[5. Udostępnianie folderów 11](#_Toc155294242)

[5.1 Konfiguracja 11](#_Toc155294243)

[5.2 Użytkowanie 13](#_Toc155294244)

1. Wstęp

W niniejszym referacie przedstawione zostaną sposoby korzystania z zasobów maszyny wirtualnej z zainstalowanym systemem Ubuntu (18.04.1 LTS) bez nakładki graficznej w programie Oracle VM Virtual Box. Do przeglądania tych zasobów na systemie Windows oraz wykonywania operacji na ww. maszynie zostanie wykorzystane IDE (Integrated Development Environment) Visual Studio Code od firmy Microsoft.

1. WSL 2

WSL (Windows Subsystem for Linux) w wersji 2, to jedna z funkcji systemu Windows (od wersji 10), pozwalająca na uruchomienie dowolnej dystrybucji Linux, w wysoce zoptymalizowanej maszynie wirtualnej, z pomocą technologii Hyper-V. Taka maszyna posiada specjalną wersję jądra, zaprojektowaną przez firmę Microsoft, która posiada pewne różnice.

* 1. Konfiguracja

Aby zainstalować WSL 2, należy uruchomić wiersz poleceń systemu Windows w trybie administratora a następnie wykonać polecenie wsl.exe --install.

A computer screen with white text

Description automatically generated

Rys . Efekt działania polecenia

Domyślnie zostanie pobrany system Ubuntu w najnowszej wersji. Po zrestartowaniu komputera uruchomi się wiersz poleceń, w którym będzie trzeba podać dane użytkownika dla nowej maszyny.

A screenshot of a computer

Description automatically generated

Rys . Po zrestartowaniu

* 1. Użytkowanie

W Visual Studio Code wybieramy opcję zdalnego połączenia znajdująca się w lewym dolnym rogu okna programu.

A screenshot of a computer program

Description automatically generated

Rys . Opcja zdalnego połączenia w Visual Studio Code

Następnie wybieramy opcję „Connect to WSL”, po czym Visual Studio Code automatycznie połączy nas z WSL, ewentualnie prosząc o zainstalowanie dodatkowego rozszerzenia. Po połączeniu można swobodnie poruszać się po katalogach na wirtualnej maszynie oraz przeciągać pliki do zakładki „Explorer”.

A screenshot of a computer

Description automatically generated

Rys . Udane połączenie z WSL

1. SSH

SSH (Secure Shell) to następca protokołu Telnet, służący do łączenia się ze zdalnymi komputerami, poprzez konsolę. Pozwala również na przesyłanie plików między maszynami   
z pomocą SCP (Secure Copy) oraz SFTP (Secure FTP).

* 1. Konfiguracja

Aby połączyć się z maszyną wirtualną, musimy skorzystać z wirtualnej karty sieciowej którą oferuje VirtualBox „VirtualBox Host-Only Ethernet Adapter”. Domyślnie karta ta ma włączony serwer DHCP z pulą adresów od 192.168.56.101 - 192.168.56.254.

A computer screen shot of a network

Description automatically generated

Rys . Konfiguracja maszyny

Po uruchomieniu maszyny, należy sprawdzić jaki adres IP został jej przydzielony. Można to zrobić za pomocą polecenia ifconfig.

A screenshot of a computer

Description automatically generated

Rys . Sprawdzenie adresu IP maszyny

Jeśli maszyna nie posiada zainstalowanego SSH, należy zmienić ustawienia karty na NAT, uruchomić maszynę i pobrać poleceniem sudo apt install ssh. Jeśli wszystko zostało wykonane poprawnie, można przystąpić do łączenia się do maszyny.

* 1. Użytkowanie

W Visual Studio Code, po wybraniu opcji zdalnego połączenia, wybieramy opcję „SSH”, co poskutkuje pobraniem rozszerzenia „Remote – SSH”.

A screenshot of a computer

Description automatically generated

Rys . Opcja SSH

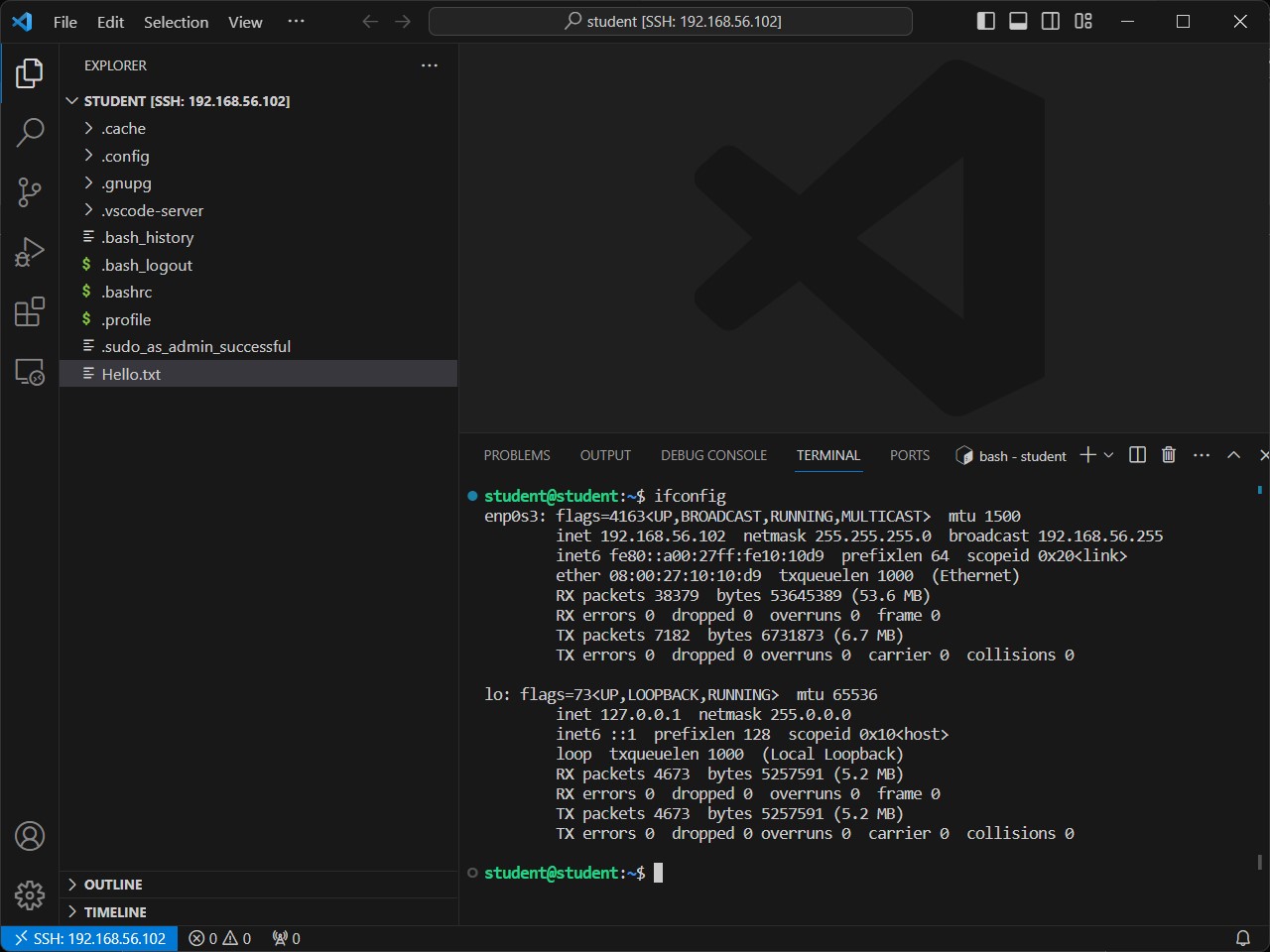
Następnie podajemy adres maszyny wraz z nazwą użytkownika, na którego będziemy się logować. Dane te podajemy w formacie ***nazwa\_użytkownika@adres\_ip\_maszyny***.

A screenshot of a phone

Description automatically generated

Rys . Dane do połączenia z maszyną

W trakcie całej operacji, wielokrotnie możemy zostać spytani o hasło użytkownika. Przy każdym otwieraniu folderu w środowisku IDE, również musimy podać hasło. Przy pierwszym połączenie Visual Studio Code zainstaluje na serwerze, serwer do zdalnego połączenia.



Rys . Udane połączenie przez SSH

Tak jak w przypadku WSL, możemy przeciągać pliki do eksploratora plików i wykonywać polecenia na danej maszynie.

1. Samba

Samba to darmowy serwer plików zapoczątkowany przez Andrew Tridgell’a. Korzysta   
z implementacji protokołu SMB (Server Message Block), który pozwala na korzystanie   
z udostępnianych zasobów na różnych systemach operacyjnych.

* 1. Konfiguracja

Na maszynie z kartą sieciową ustawioną na „NAT” instalujemy Sambę poleceniem sudo apt-get install samba. Aby sprawdzić, czy została zainstalowana poprawnie uruchamiamy polecenie samba, jeśli instalacja przebiegła poprawnie, powinniśmy otrzymać informację o zainstalowanej wersji Samby.

A computer screen shot of a black screen

Description automatically generated

Rys . Poprawna instalacja Samby

Od teraz Samba będzie uruchamiać się przy każdym uruchomieniu maszyny. Aby sprawdzić, czy Samba jest aktywna, można użyć polecenia service smbd status. Samba powinna mieć status active (running), jeśli tak nie jest, należy ją uruchomić, wykonując polecenie   
service smbd start.

A screenshot of a computer

Description automatically generated

Rys . Sprawdzenie stanu Samby

Konfiguracja serwera znajduje się w pliku ***/etc/samba/smbd.conf***, w którym należy dodać udział, z którego będziemy korzystać na innych maszynach. W tym przypadku udostępniony zostanie folder domowy użytkownika student. Za pomocą edytora nano został dodany zasób ***[sambashare]*** która udostępnia katalog ***/home/student***. Następnie w Sambie należy utworzyć użytkownika, którym powinien być użytkownik istniejący już na maszynie. Robi się to poleceniem sudo smbpasswd -a nazwa\_uzytkownika, po wykonaniu którego jesteśmy poproszeni o hasło dla nowego użytkownika, nie musi być to to same hasło co użytkownika systemowego. W tym przypadku jest to student. Na koniec należy zrestartować Sambę poleceniem sudo service smbd restart.

A screenshot of a computer

Description automatically generated

Rys . Konfiguracja Samby

* 1. Użytkowanie

Po tych krokach serwer Samby jest już gotowy i można się do niego dostać za pomocą Eksploratora Plików w systemie Windows. Można to zrobić wpisując w pasek adresu url   
**\\adres\_ip\_maszyny**, gdzie powinien znajdować się folder **sambashare**. Po wejściu w tenże folder użytkownik zostanie poproszony o podanie danych wcześniej utworzonego użytkownika w Sambie.

A screenshot of a computer

Description automatically generated

Rys . Zasób udostępniony przez serwer Samba

W takim zasobie można teraz wykonywać wszystkie operacje związane z tworzeniem plików i folderów. Wykonywanie poleceń nie jest możliwe z poziomu Windowsa, dlatego też, wszelkie polecenia należy wpisywać na maszynie.

1. Udostępnianie folderów

Oprogramowanie VirtualBox oferuje funkcję udostępniania folderów z maszyny hostującej do maszyny gościa. Wymaga to zainstalowania na maszynie wirtualnej pakietów gościa.

* 1. Konfiguracja

Najpierw należy uruchomić maszynę i zainstalować na niej pakiety ***build-essentials*** ***linux-headers-(wersja kernela)*** oraz ***dkms***. Można to zrobić poleceniem   
sudo apt install build-essentials linux-headers-$(uname -r) dkms. Po udanej instalacji   
z pomocą menu wirtualnej maszyny należy zamontować obraz pakietów gościa.

A screenshot of a computer

Description automatically generated

Rys . Montowanie obrazu dodatków

W tym momencie obraz dodatków znajduje się w wirtualnym napędzie i trzeba go uruchomić na maszynie. Do tego celu tworzymy folder */media/cdrom* który będzie punktem   
do zamontowania zawartości wirtualnego napędu. Robimy to poleceniem  
sudo mkdir -p /media/cdrom. Po utworzeniu folderu można zamontować obraz płyty poleceniem sudo mount /dev/cdrom /media/cdrom. W obrazie płyty znajduje się instalator, który uruchamiamy poleceniem sudo /media/cdrom/VBoxLinuxAdditions.run. Jeżeli wszystko zostało wykonane poprawnie, możemy utworzyć w katalogu domowym folder „Windows” który posłuży za łącznik między systemami.

Na Windowsie należy utworzyć folder, który ma zostać udostępniony na maszynie. Po utworzeniu tego folderu, w ustawieniach maszyny w zakładce „Shared folders” tworzymy nowe połączenie.

A screenshot of a computer

Description automatically generated

Rys . Tworzenie zasobu

„Folder Path” to ścieżka do folderu utworzonego na Windowsie. „Folder Name” to dowolna nazwa służąca jedynie do identyfikacji w oprogramowaniu VirtualBox. „Mount point” to ścieżka na maszynie, w której zostanie zamontowany folder. W tym przypadku jest to katalog Windows w katalogu domowym użytkownika student. Należy również zaznaczyć opcje „Auto-mount” i „Make Permament”.

* 1. Użytkowanie

Wszystko co znajduje się w udostępnionym maszynie folderze, będzie dostępne na tej maszynie w folderze ***/home/student/Windows***. Folder ten można otworzyć w Visual Studio Code, a polecenia wykonywać z konsoli maszyny.