

# **Akademia Nauk Stosowanych w Nowym Sączu**

Wydział Nauk Inżynierijnych

## **Systemy operacyjne – projekt**

studia stacjonarne

semestr letni 2023/2024

### **Temat projektu:**

1. Zaprojektować infrastrukturę informatyczną na potrzeby firmy Binary-Builders. Realizacja serwerowa w oparciu o system operacyjny Linux, np. Fedora Server 39, stacje klienckie np. Linux MINT.
2. Wdrożyć niezbędne usługi wynikające z założeń takie jak: SSH, DHCP, DNS, HTTP/S, motor bazodanowy (MySql)+PHP+phpMyAdmin, CMS WordPress, RAID, SAMBA, SQUID, Postfix(SMTP) + Dovecot(POP/IMAP), oraz wybraną usługę. Wdrożyć automatyzację przy użyciu skryptu np. Bash, oraz usługi cron.
3. Cele projektu zweryfikować z założeniami zapisanymi w dokumencie „Szczegółowy zarys projektu”.

Imię i nazwisko:

Maciej Wójs

Data oddania:

1 czerwca 2024

Nr grupy:

L3

Ocena:

# Spis treści

<b>1 Założenia projektowe – wymagania</b>	<b>4</b>
<b>2 Opis użytych technologii</b>	<b>5</b>
2.1 SSH (Secure Shell) . . . . .	5
2.2 DHCP (Dynamic Host Configuration Protocol) . . . . .	5
2.3 DNS (Domain Name System) . . . . .	5
2.4 HTTP/S (Hypertext Transfer Protocol/Secure) . . . . .	5
2.5 MySQL . . . . .	5
2.6 PHP . . . . .	5
2.7 phpMyAdmin . . . . .	5
2.8 CMS WordPress . . . . .	5
2.9 RAID (Redundant Array of Independent Disks) . . . . .	6
2.10 SAMBA . . . . .	6
2.11 SQUID . . . . .	6
2.12 Postfix (SMTP) + Dovecot (POP/IMAP) . . . . .	6
2.12.1 Postfix . . . . .	6
2.12.2 Dovecot . . . . .	6
2.13 Automatyzacja za pomocą skryptów Bash i usług cron . . . . .	6
2.13.1 Skrypty Bash . . . . .	6
2.13.2 cron . . . . .	6
<b>3 Schemat logiczny projektowanej infrastruktury sieciowej</b>	<b>7</b>
<b>4 Procedury instalacyjne poszczególnych usług</b>	<b>8</b>
4.1 Instalacja systemu klienta – Linux Mint . . . . .	8
4.1.1 Proces instalacji . . . . .	8
4.1.2 Wstępna konfiguracja systemu . . . . .	12
4.2 Instalacja serwera – Fedora 40 . . . . .	13
4.2.1 Proces instalacji . . . . .	13
4.2.2 Wstępna konfiguracja . . . . .	19
4.3 Konfiguracja SSH . . . . .	23
4.4 Nazwa serwera – hostname . . . . .	25
4.5 DNS – instalacja i konfiguracja . . . . .	25
4.6 DHCP – instalacja i konfiguracja . . . . .	32
<b>5 Testy działania wdrożonych usług</b>	<b>35</b>
5.1 DNS . . . . .	35
5.2 DHCP . . . . .	35
<b>6 Kod skryptu BASH, oraz tablica crontab</b>	<b>37</b>
<b>7 Wnioski</b>	<b>37</b>
<b>8 Literatura</b>	<b>38</b>

## Spis rysunków

1	Schemat logiczny sieci . . . . .	7
2	Tworzenie nowej maszyny wirtualnej . . . . .	8
3	Przydzielanie zasobów maszynie wirtualnej . . . . .	8
4	Określenie rozmiaru dysku wirtualnego. . . . .	9
5	Podsumowanie konfiguracji maszyny wirtualnej . . . . .	9
6	Rozpoczęcie instalacji Linux Mint . . . . .	10
7	Wybór trybu instalacji na dysku twardym. . . . .	10
8	Tworzenie konta użytkownika . . . . .	11
9	Zakończenie instalacji systemu Linux Mint. . . . .	11
10	Instalacja dodatków gościa . . . . .	12
11	Aktualizacja pakietów . . . . .	12
12	Podsumowanie maszyny wirtualnej Fedora 40 . . . . .	13
13	Dodanie pierwszej karty sieciowej . . . . .	13
14	Dodanie drugiej karty sieciowej . . . . .	14
15	Dodanie trzeciej karty sieciowej . . . . .	14
16	Uruchomienie instalatora Fedory. . . . .	15
17	Rozpoczęcie instalacji Fedora . . . . .	15
18	Wybór dysku instalacji . . . . .	16
19	Ustawienie konta root . . . . .	16
20	Stworzenie użytkownika . . . . .	17
21	Ekran postępującej instalacji . . . . .	17
22	Ekran przed restartem do systemu. . . . .	18
23	Zainstalowany system Fedora 40 . . . . .	18
24	Konfiguracja dnf . . . . .	19
25	Aktualizacja pakietów . . . . .	20
26	plik /etc/default/grub przed zmianą . . . . .	20
27	plik /etc/default/grub po zmianie . . . . .	21
28	Zastosowanie zmian po edycji grub . . . . .	21
29	Zwiększenie wygody wpisywania haseł . . . . .	22
30	Efekt działania zmiany ustawień . . . . .	22
31	konfiguracja karty sieciowej . . . . .	23
32	Konfiguracja PuTTY . . . . .	23
33	Próba podłączenia poprzez PuTTY . . . . .	24
34	Wynik połączenia poprzez PuTTY . . . . .	24
35	Zmiana nazwy serwera . . . . .	25
36	Edycja /etc/hosts . . . . .	25
37	Instalacja DNS . . . . .	26
38	Kopia zapasowa pliku konfiguracyjnego DNS . . . . .	27
39	zawartość named.conf . . . . .	29
40	zawartość pliku strefy podstawowej . . . . .	30
41	zawartość pliku strefy dla przeszukiwania wstępznego . . . . .	31
42	Uruchomienie usługi DNS . . . . .	32
43	Instalacja DHCP . . . . .	32
44	Konfiguracja DHCP . . . . .	33
45	Instalacja DHCP . . . . .	34
46	Test DNS . . . . .	35
47	Instalacja DHCP . . . . .	35

# 1 Założenia projektowe – wymagania

- a) Systemy operacyjne: Fedora Server 39 lub inny serwer z rodziną Linux, oraz system kliencki np. Linux MINT.
- b) zarządzanie serwerem poprzez SSH, oraz emulator putty.exe
- c) nazwa serwera ma być zgodna z nazewnictwem: svrXX-firma, gdzie XX oznaczają dwie ostatnie cyfry numeru albumu wykonawcy, a firma to skrót nazwy swojej firmy (niepowtarzalny) – wymyślonej,
- d) na podstawie nazwy firmy należy założyć lokalną domenę o nazwie np. firma.ns i skonfigurować usługę DNS Server,
- e) adres IP serwera, zakres adresacji IP, oraz brama domyślna od strony sieci wewnętrznej VirtualBOXa (sieć LAN firmy) w której ma działać serwer DHCP ma mieć następujące wartości:

adres IP:	192.168.230.1/24,
zakres:	192.168.230.10–60
brama domyślna:	192.168.230.1
- f) należy utworzyć macierz dyskową programową na poziomie RAID 5 z dyskiem zapasowym. Uzyskać wypadkową pojemności macierzy 10GB. Przestrzeń macierzy podzielić na dwie równe partycje,
- g) Pierwszą partycję zamontować do punktu **/dysksieciowy**, a drugą do punktu **/kopie**. Zapewnić ich automatyczne montowanie podczas startu systemu,
- h) serwer ma udostępniać zasób sieciowy o adresie UNC **\sfs.firma.ns\dysk** odnoszący się do systemu plików **/dysksieciowy** (ppkt. g),
- i) należy wdrożyć usługę WEB Server z obsługą PHP, oraz serwer bazodanowy zarządzany przez phpMyAdmin, oraz CMS WordPress, skonfigurować UserDir dla WEB Serwer'a,
- j) dostęp do sieci Internet z sieci wewnętrznej ma się odbywać za pośrednictwem serwera PROXY(squid), a aktywność pracowników firmy ma być monitorowana,
- k) w firmie należy wdrożyć serwer pocztowy, oraz klienta mail,
- l) zapewnić aby popularne usługi były dostępne jako oddzielne nazwy hostów, jak np.:
  - **www.firma.ns** (serwer www),
  - **poczta.firma.ns** (serwer poczty),
  - **sfs.firma.ns** (serwer samby),
- m) wdrożyć automatyczną archiwizację systemu plików /home zawierającego katalogi użytkowników. Archiwizacja ma rozpoczynać się automatycznie codziennie o 21:00. W wyniku archiwizacji ma powstać plik **home\_20240510.tar.gz** zapisany w **/kopie** (ppkt. g)
- n) Dodatkowo wdrożyć dowolną usługę, ale taką która nie była wdrażana podczas zajęć.

## **2 Opis użytych technologii**

### **2.1 SSH (Secure Shell)**

SSH to protokół sieciowy, który umożliwia bezpieczne zdalne logowanie oraz wykonywanie poleceń na odległym serwerze. Zapewnia szyfrowanie komunikacji, co chroni przed podsłuchiwaniem oraz atakami typu man-in-the-middle.

### **2.2 DHCP (Dynamic Host Configuration Protocol)**

DHCP to protokół używany do automatycznego przydzielania adresów IP i innych parametrów konfiguracyjnych urządzeniom w sieci. Ułatwia zarządzanie siecią poprzez automatyczne przypisywanie ustawień.

### **2.3 DNS (Domain Name System)**

DNS to system, który przekształca łatwe do zapamiętania nazwy domen (np. www.example.com) na adresy IP, które są wykorzystywane przez urządzenia sieciowe do komunikacji. DNS działa jak książka telefoniczna internetu.

### **2.4 HTTP/S (Hypertext Transfer Protocol/Secure)**

HTTP to protokół komunikacyjny używany do przesyłania stron internetowych. HTTPS to jego bezpieczna wersja, która wykorzystuje TLS/SSL do szyfrowania danych, zapewniając poufność i integralność komunikacji między przeglądarką a serwerem.

### **2.5 MySQL**

Popularny system zarządzania relacyjnymi bazami danych. Umożliwia przechowywanie i zarządzanie dużą ilością danych w strukturach tabelarycznych.

### **2.6 PHP**

Skryptowy język programowania, często używany do tworzenia dynamicznych stron internetowych. PHP może komunikować się z bazami danych, takimi jak MySQL.

### **2.7 phpMyAdmin**

Narzędzie webowe do zarządzania bazami danych MySQL. Umożliwia wykonywanie operacji na bazach danych za pomocą interfejsu graficznego.

### **2.8 CMS WordPress**

WordPress to system zarządzania treścią (CMS), który pozwala na łatwe tworzenie i zarządzanie stronami internetowymi. Jest bardzo popularny ze względu na swoją elastyczność, prostotę obsługi oraz bogaty ekosystem wtyczek i motywów.

## **2.9 RAID (Redundant Array of Independent Disks)**

RAID to technologia, która łączy kilka dysków twardych w jedną jednostkę logiczną w celu poprawy wydajności i/lub redundancji danych. Istnieje kilka poziomów RAID, z których każdy oferuje różne kombinacje wydajności i bezpieczeństwa danych.

## **2.10 SAMBA**

SAMBA to pakiet oprogramowania, który umożliwia integrację systemów operacyjnych Linux/Unix z sieciami Windows. Pozwala na udostępnianie plików i drukarek w sieci oraz współpracę z domenami Windows (Active Directory).

## **2.11 SQUID**

SQUID to serwer proxy i buforujący, który może przyspieszyć dostęp do zasobów internetowych poprzez przechowywanie często używanych danych w lokalnej pamięci podręcznej. Może również służyć jako filtr treści i narzędzie do monitorowania ruchu sieciowego.

## **2.12 Postfix (SMTP) + Dovecot (POP/IMAP)**

### **2.12.1 Postfix**

Serwer pocztowy obsługujący protokół SMTP, używany do wysyłania i odbierania wiadomości e-mail. Jest znany z wydajności i bezpieczeństwa.

### **2.12.2 Dovecot**

Serwer IMAP i POP3 używany do odbierania i przechowywania wiadomości e-mail. Jest zoptymalizowany pod kątem wydajności i bezpieczeństwa, oferując wsparcie dla nowoczesnych standardów pocztowych.

## **2.13 Automatyzacja za pomocą skryptów Bash i usług cron**

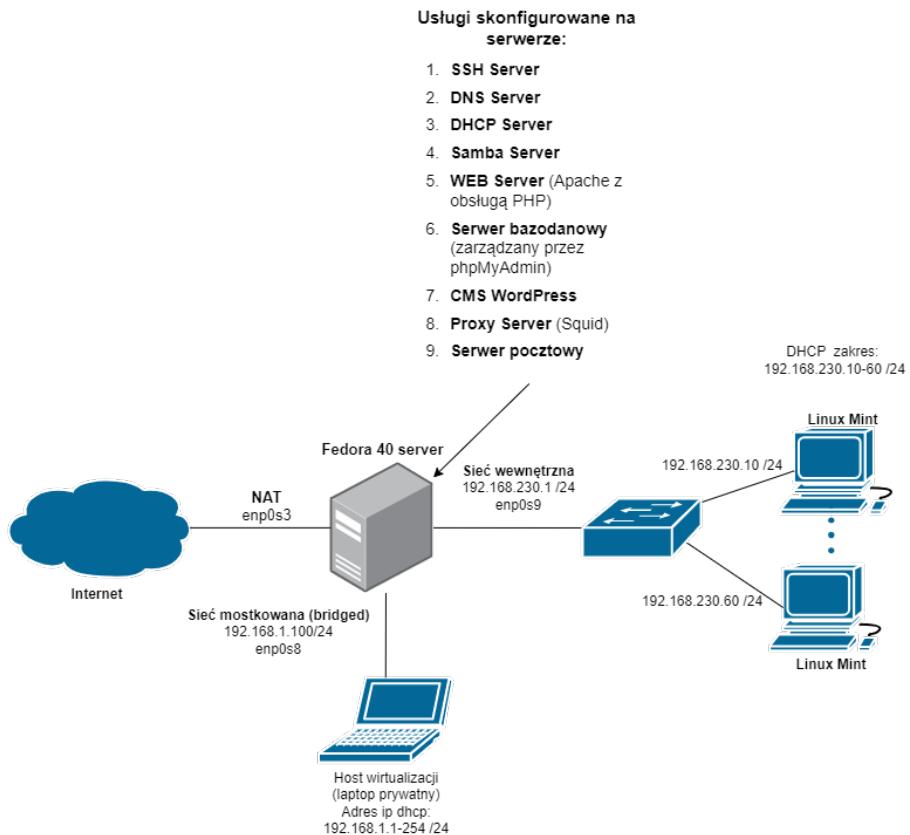
### **2.13.1 Skrypty Bash**

Skrypty napisane w Bash (Bourne Again Shell) służą do automatyzacji zadań w systemach Unix/Linux. Mogą być używane do instalacji oprogramowania, konfiguracji systemu, zarządzania plikami i wielu innych zadań.

### **2.13.2 cron**

Usługa systemowa w Unix/Linux, która pozwala na planowanie zadań do wykonania w określonym czasie lub regularnych odstępach czasu. Jest używana do automatyzacji zadań takich jak backup, aktualizacje systemu czy uruchamianie skryptów.

### 3 Schemat logiczny projektowanej infrastruktury sieciowej

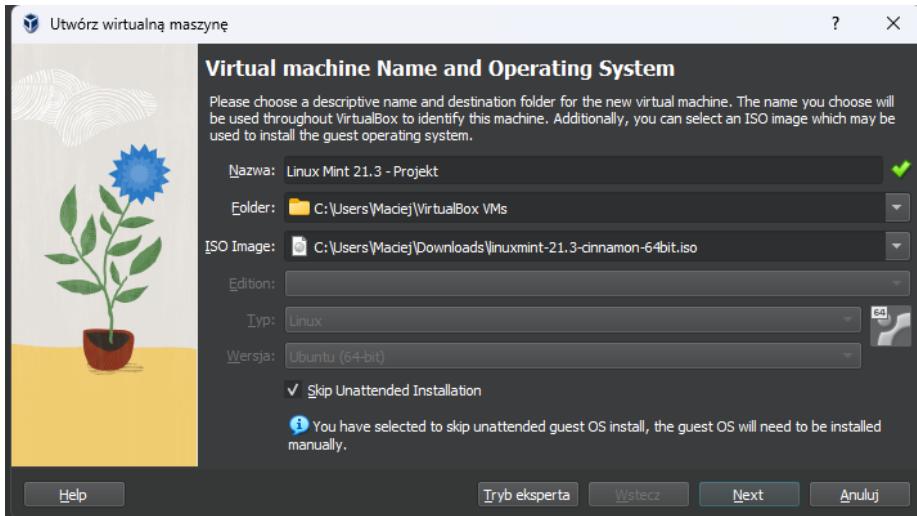


Rysunek 1: Schemat logiczny sieci

## 4 Procedury instalacyjne poszczególnych usług

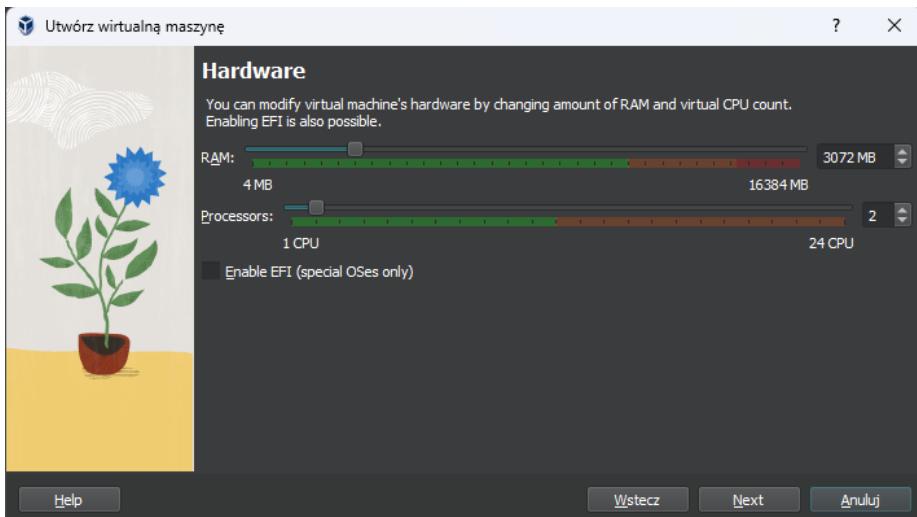
### 4.1 Instalacja systemu klienta – Linux Mint

#### 4.1.1 Proces instalacji



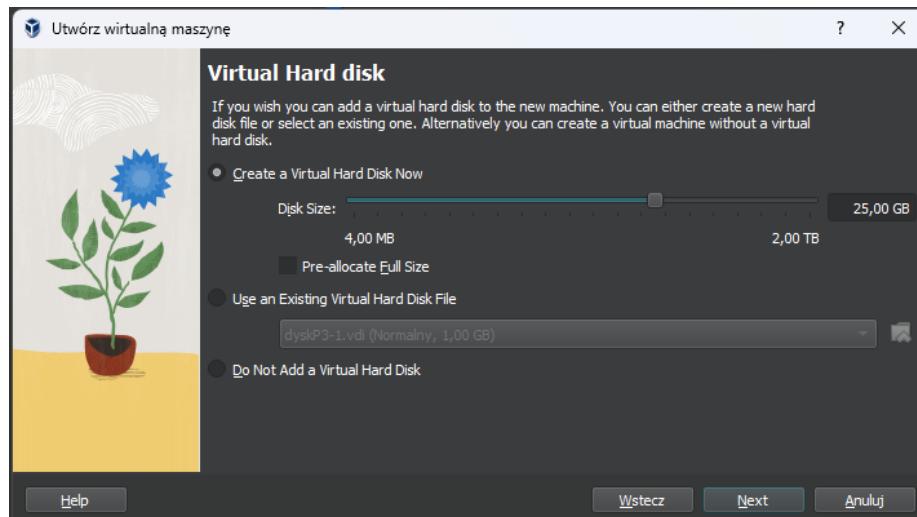
Rysunek 2: Tworzenie nowej maszyny wirtualnej. Ustawienia nazwy, lokalizacji dysku oraz wybór pliku ISO systemu operacyjnego.

Pierwszym krokiem jest utworzenie nowej maszyny wirtualnej (VM). W tym etapie określa się nazwę maszyny, lokalizację dysku, gdzie będzie przechowywana, oraz wybiera odpowiedni plik ISO z systemem Linux Mint.



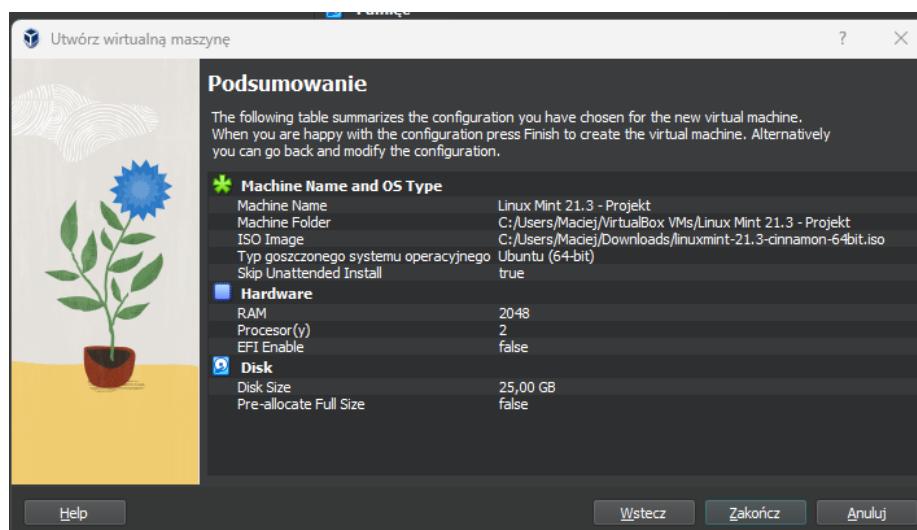
Rysunek 3: Przydzielanie zasobów maszynie wirtualnej, takich jak pamięć RAM i procesor.

W kolejnym kroku przydzielane są zasoby dla maszyny wirtualnej, w tym ilość pamięci RAM oraz liczba rdzeni procesora.



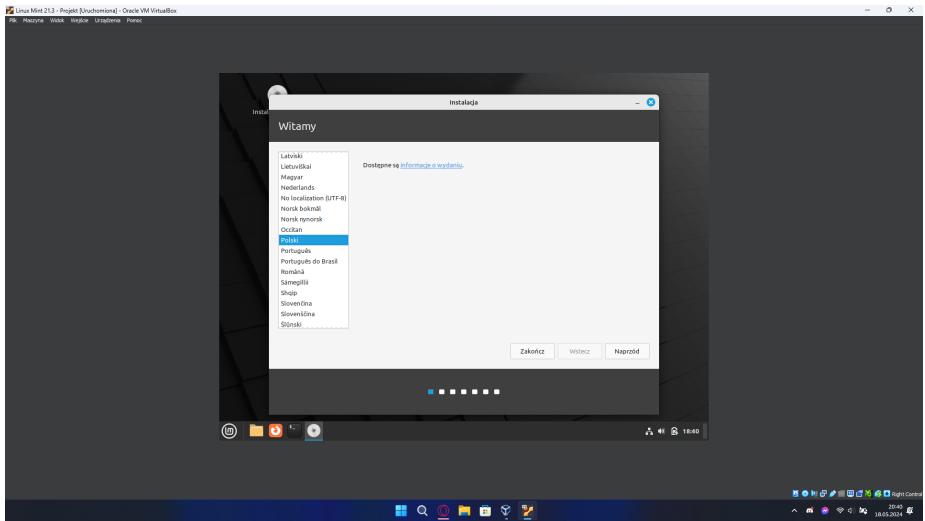
Rysunek 4: Określenie rozmiaru dysku wirtualnego.

Następnie należy zdefiniować rozmiar wirtualnego dysku twardego, który będzie używany przez maszynę wirtualną.



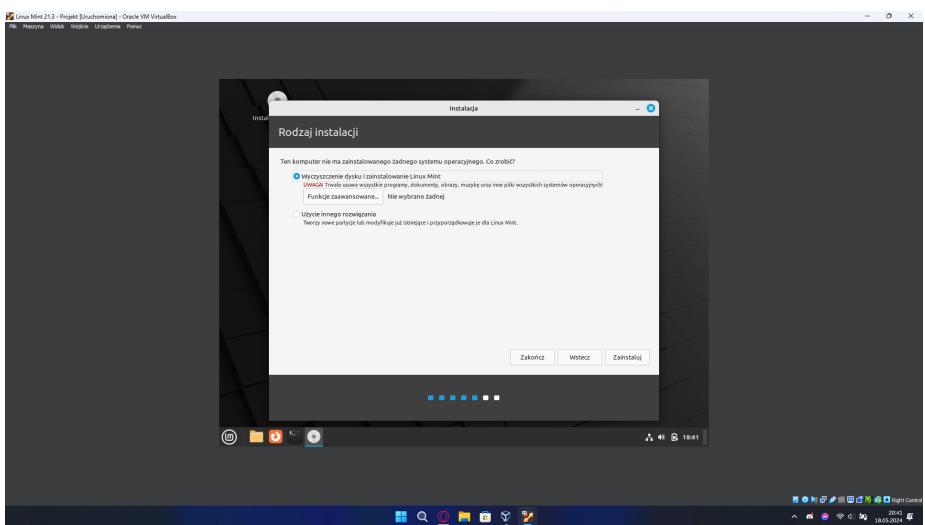
Rysunek 5: Podsumowanie konfiguracji maszyny wirtualnej przed rozpoczęciem instalacji systemu.

Po skonfigurowaniu wszystkich ustawień, wyświetlane jest podsumowanie zawierające wszystkie wybrane opcje dla nowo utworzonej maszyny wirtualnej.



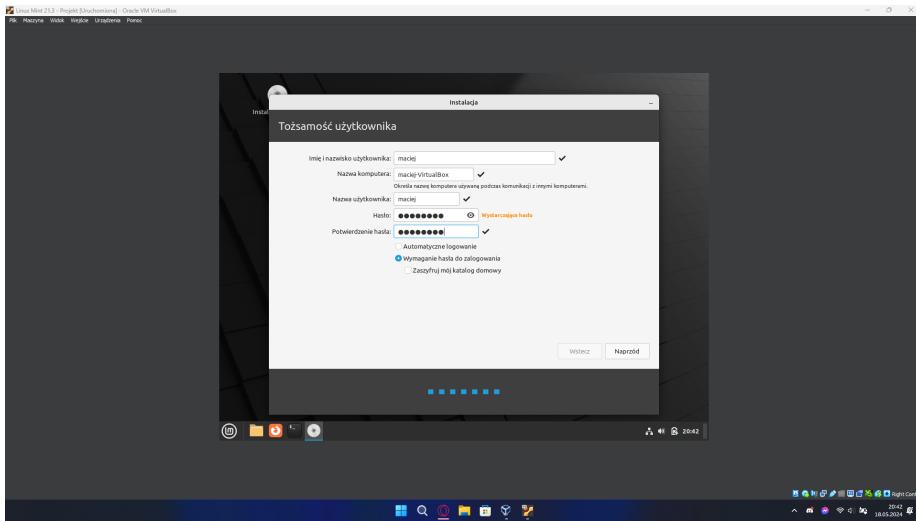
Rysunek 6: Rozpoczęcie instalacji Linux Mint – wybór języka instalacji.

Rozpoczyna się proces instalacji Linux Mint. Pierwszym krokiem jest wybór języka, który będzie używany podczas instalacji i w systemie.



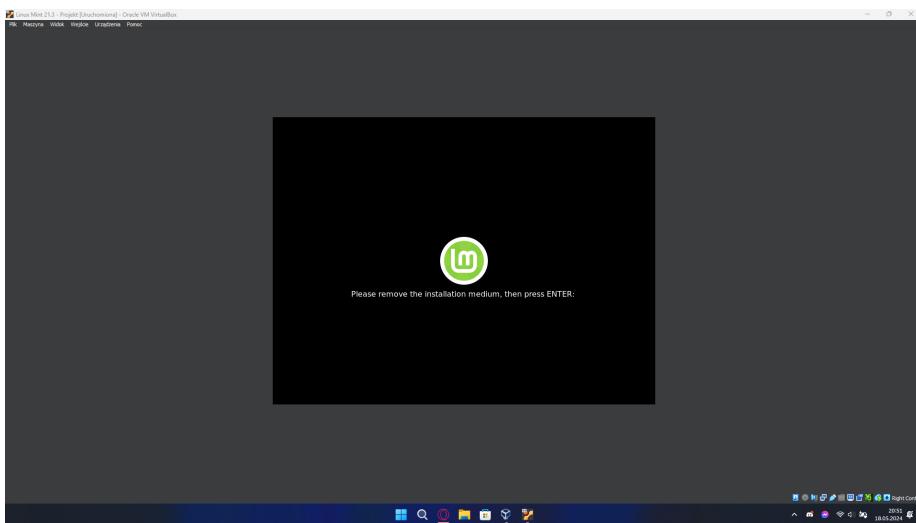
Rysunek 7: Wybór trybu instalacji na dysku twardym.

Następnie użytkownik wybiera sposób instalacji systemu na dysku twardym, na przykład automatyczne partycjonowanie lub ręczne tworzenie partycji.



Rysunek 8: Tworzenie konta użytkownika i konfiguracja podstawowych ustawień.

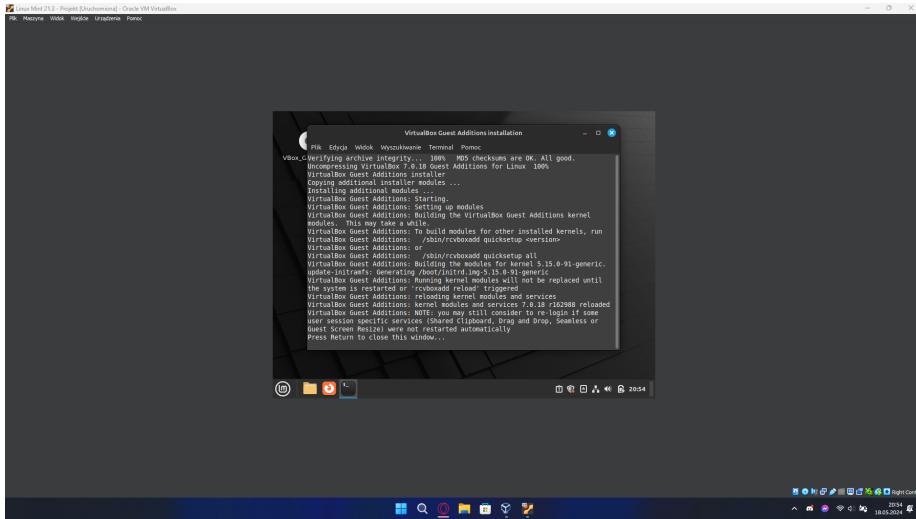
Kolejnym krokiem jest utworzenie konta użytkownika, wprowadzenie nazwy użytkownika, hasła oraz nazwy komputera.



Rysunek 9: Zakończenie instalacji systemu Linux Mint.

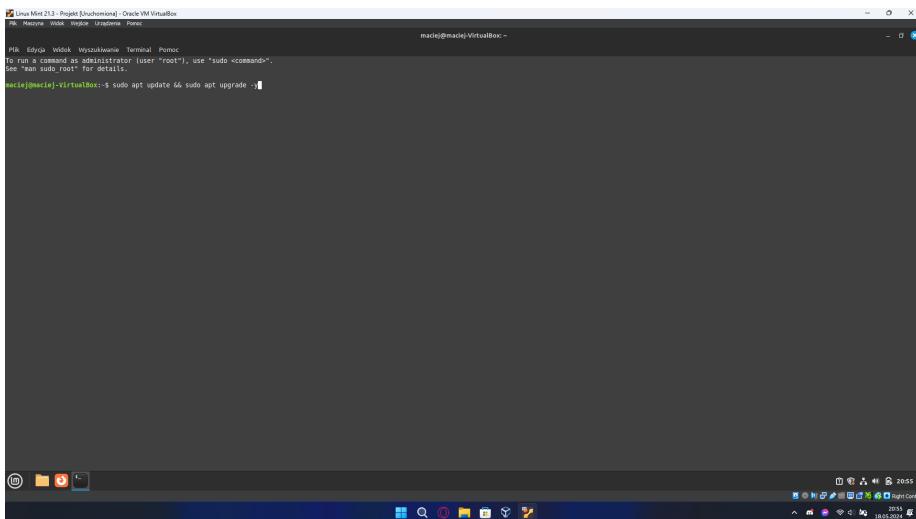
Wyświetlony zostaje monit z prośbą o usunięcie nośnika instalacyjnego. Po zakończeniu instalacji system wyświetla ekran informujący o pomyślnym zakończeniu procesu.

#### 4.1.2 Wstępna konfiguracja systemu



Rysunek 10: Instalacja dodatków gościa dla poprawy wydajności i integracji z systemem hosta.

Po zainstalowaniu systemu operacyjnego warto zainstalować dodatki gościa, które poprawiają integrację maszyny wirtualnej z systemem hosta, co zwiększa komfort pracy.

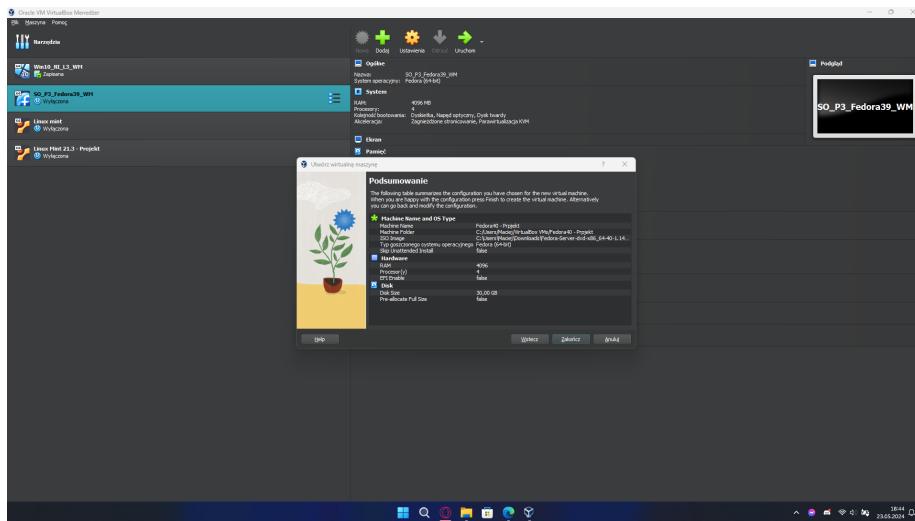


Rysunek 11: Aktualizacja pakietów systemowych.

Ostatnim krokiem wstępnej konfiguracji jest aktualizacja pakietów systemowych, aby zapewnić, że system operacyjny ma najnowsze poprawki i funkcje.

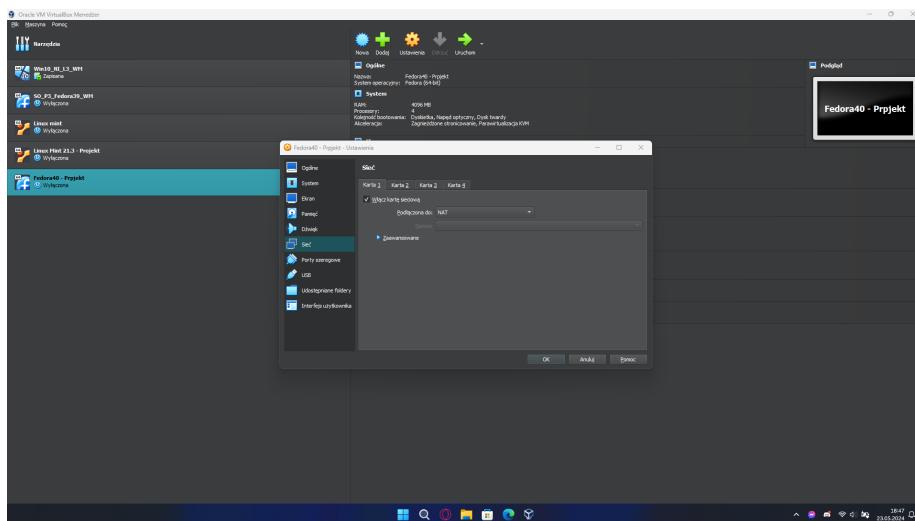
## 4.2 Instalacja serwera – Fedora 40

### 4.2.1 Proces instalacji



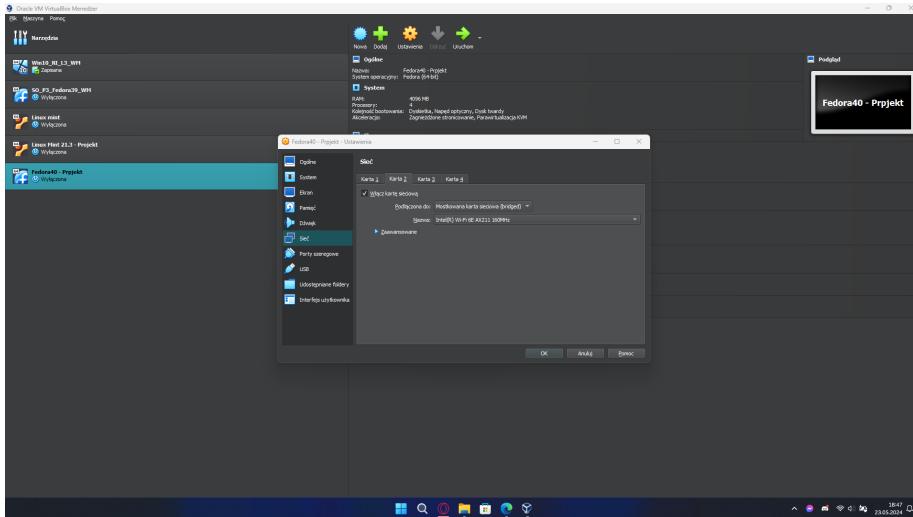
Rysunek 12: Analogicznie jak w przypadku instalacji Linux Mint – wymagane jest ustawienie nazwy maszyny wirtualnej, przydzielenie jej zasobów, ustalenie rozmiaru dysku. Powyższe zdjęcie ukazuje ekran z podsumowaniem wybranych opcji

Aby maszyna wirtualna miała dostęp do internetu wymagane jest dodanie karty sieciowej NAT, co widać na poniższym zdjęciu.



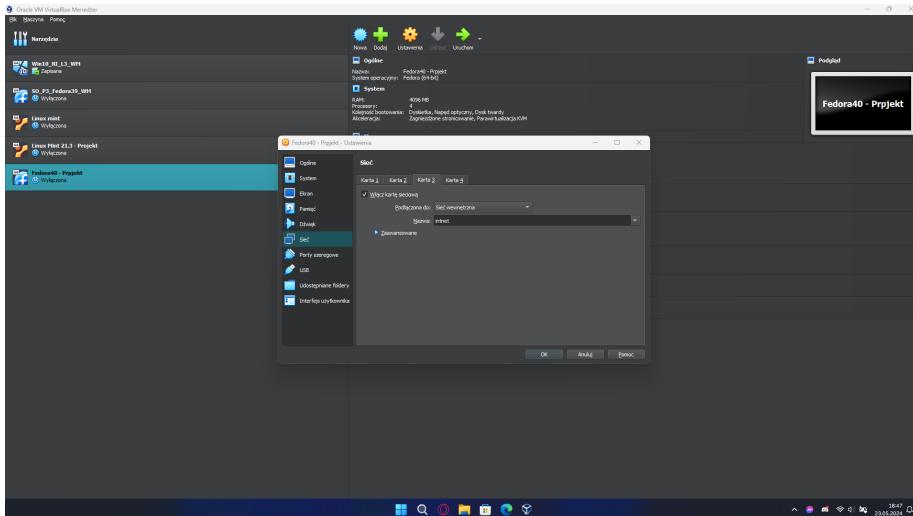
Rysunek 13: Dodanie pierwszej karty sieciowej – NAT

Druga karta sieciowa jest dodana w celu połączenia się hosta z maszyną wirtualną poprzez protokół SSH oraz udostępnienia usług takich jak http czy samba. Połączenie poprzez SSH umożliwia łatwiejszą konfigurację maszyny wirtualnej.



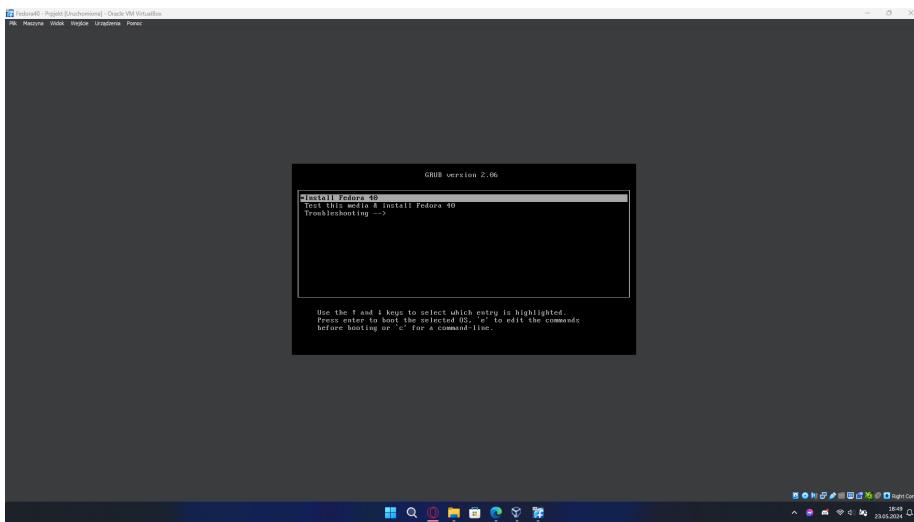
Rysunek 14: Dodanie pierszej drugiej karty sieciowej – sieć mostkowana (bridged)

Trzecia karta sieciowa posłuży do stworzenia sieci wewnętrznej dla maszyn wirtualnych w sposób taki aby się one wzajemnie widziały (tzn. były dostępne), a nie były dostępne z poziomu hosta.



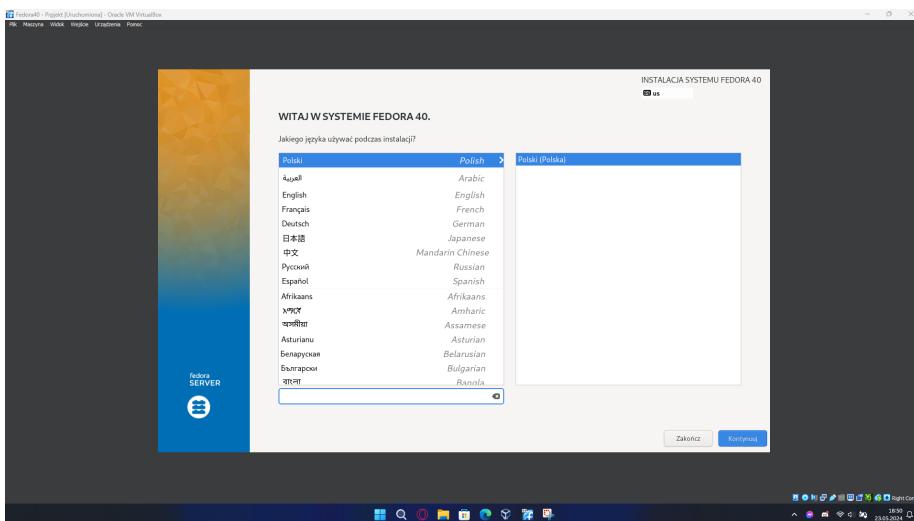
Rysunek 15: Dodanie pierszej trzeciej karty sieciowej – sieć wewnętrzna

Po dodaniu kart sieciowych można uruchomić maszynę wirtualną. Po chwili ukazuje się menu grub z opcją instalacji Fedory 40. Tą opcję należy wybrać w celu dalszej instalacji.



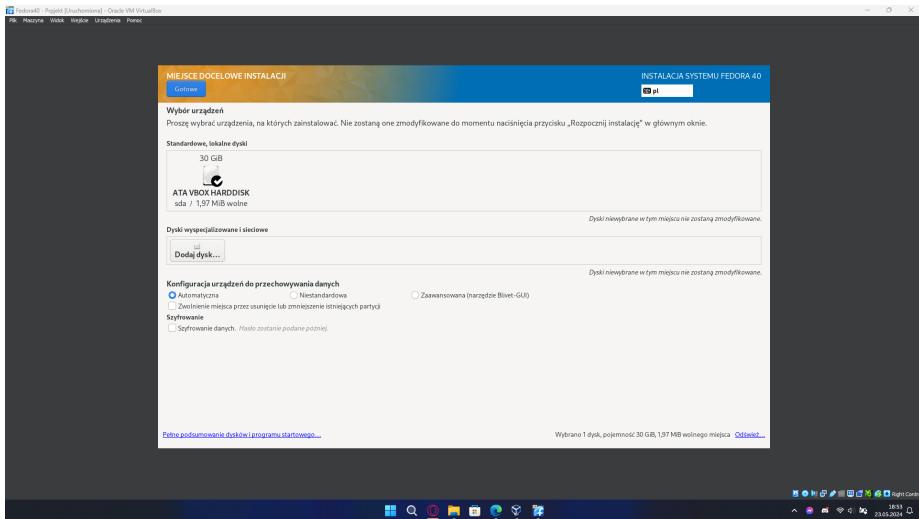
Rysunek 16: Uruchomienie instalatora Fedory.

W następnym kroku wybiera się język instalatora oraz układ klawiatury.



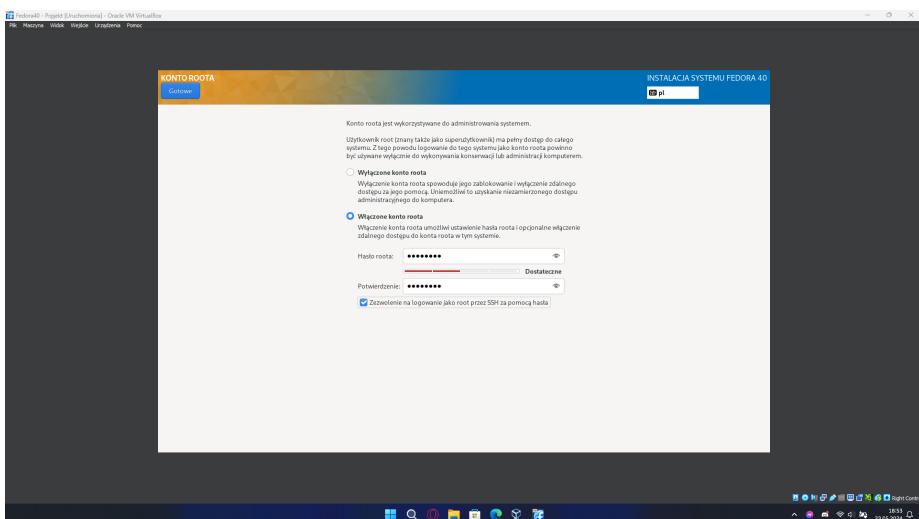
Rysunek 17: Rozpoczęcie instalacji Fedora 40 – wybór języka instalacji.

W kolejnym kroku wybieram dysk na którym ma zostać zainstalowany system. W tym miejscu można podzielić dysk na partie (podzielić na części które w systemie będą widoczne jako samodzielne dyski), sformatować go, zaszyfrować, wybrać system plików (np. ext3, ext4, zfs).



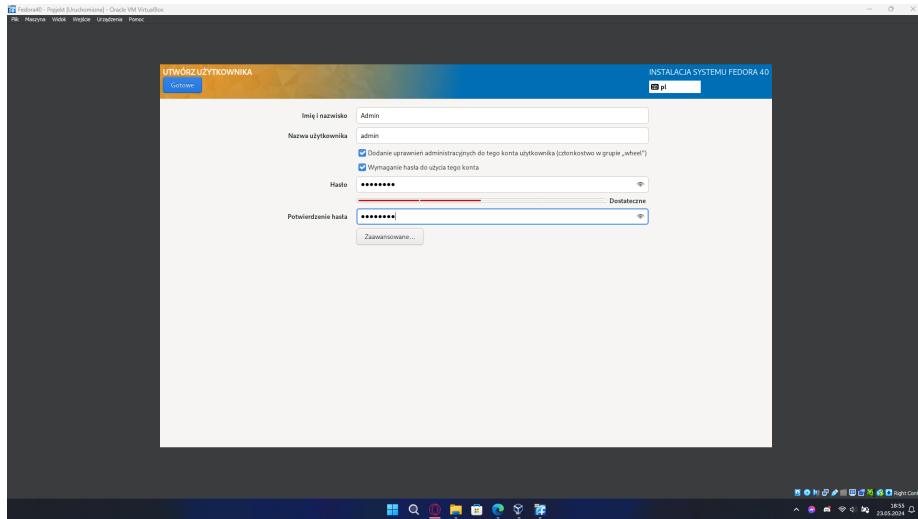
Rysunek 18: Wybór dysku na którym zostanie zainstalowany serwer

Następnie przechodzę do zakładki z ustawieniami dotyczącymi konta root. W tej zakładce ustawiam hasło do konta oraz zezwalam na połączenia SSH tym kontem. Na serwerze produkcyjnym połączenie poprzez konto root nie jest zalecanym rozwiązaniem, gdyż stanowi zagrożenie bezpieczeństwa sieci firmowej.



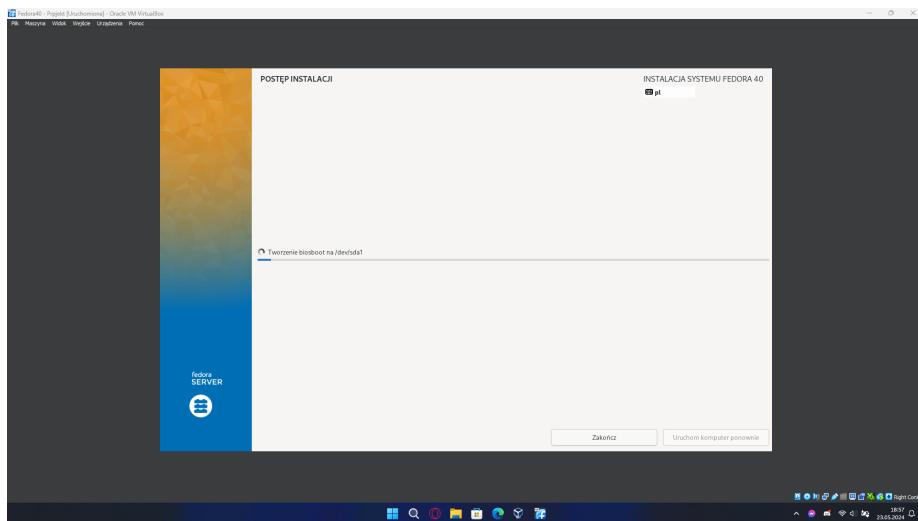
Rysunek 19: Ustawienie konta root – włączenie konta, ustawienie hasła i zezwolenie na połączenie ssh jako root

Po ustawieniu konta root'a zabieram się za stworzenie konta użytkownika. W tej części konfiguracji zaznaczam checkbox'a dotyczącego dodania konta admin do grupy wheel. Umożliwi mi to wykonywanie komendy sudo (Super User DO).



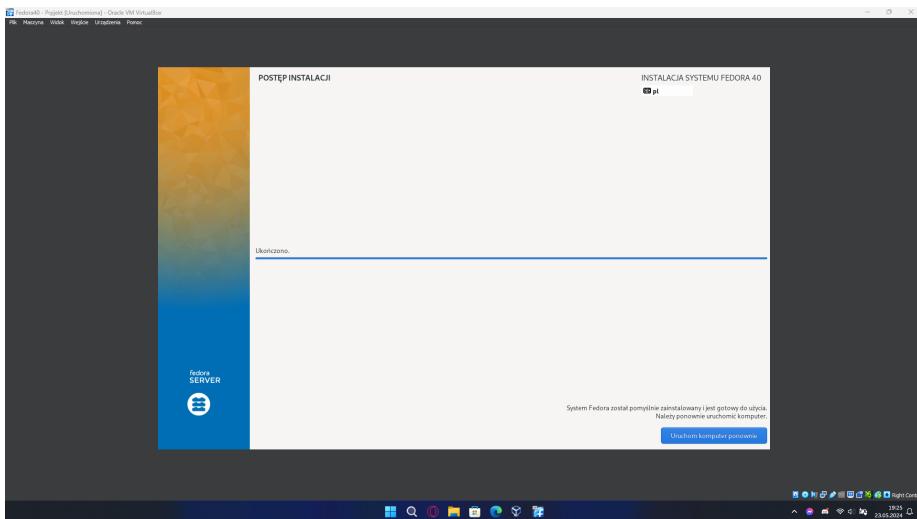
Rysunek 20: Stworzenie użytkownika – admin

Po wykonaniu powyższych kroków nie pozostaje nic innego jak rozpoczęcie instalacji.



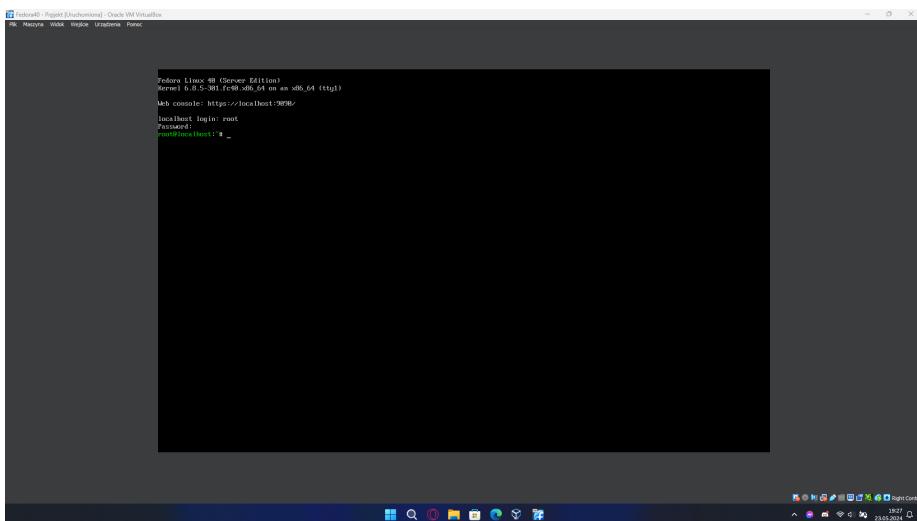
Rysunek 21: Ekran postępującej instalacji

Po jakimś czasie mogę uruchomić ponownie serwer kończąc tym samym instalację systemu.



Rysunek 22: Ekran postępującej instalacji – koniec instalacji

Po Uruchomieniu ponownym mogę zalogować się na konto root'a i zacząć konfigurację wstępna serwera.



Rysunek 23: Zainstalowany system – przed wstępnią konfiguracją

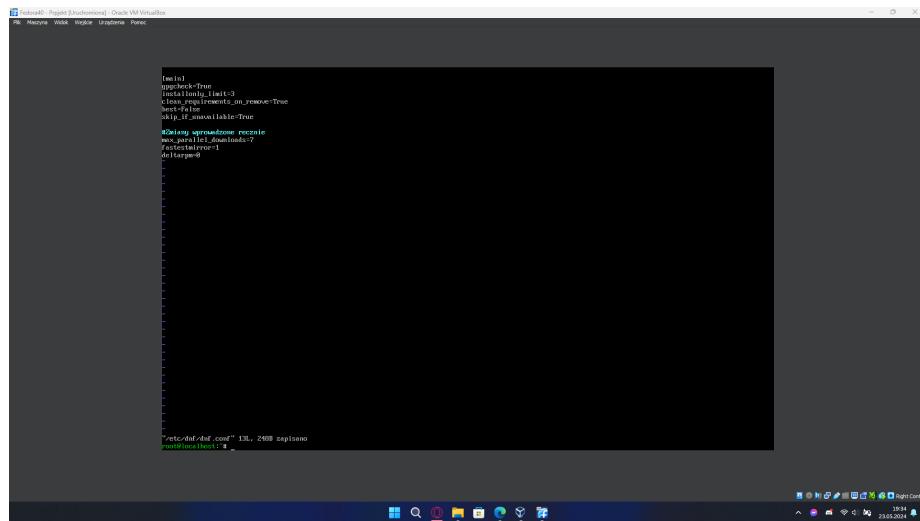
#### 4.2.2 Wstępna konfiguracja

Po zainstalowaniu systemu, następnym krokiem powinno być zaktualizowanie pakietów aby zapewnić najnowszą funkcjonalność oraz poprawki bezpieczeństwa. Jednakże przed tym krokiem decyduję się na konfigurację menadżera pakietów dnf, aby przyśpieszyć pobieranie pakietów. Do pliku /etc/dnf/dnf.conf dodaje następujące wpisy:

```
#Zmiany wprowadzone ręcznie
max_parallel_downloads=7
fastestmirror=1
deltarpm=0
```

Wytłumaczenie opcji:

- max\_parallel\_downloads=7 Opcja ta pozwala menadżerowi pakietów na pobieranie do 7 pakietów na raz.
- fastestmirror=1 Opcja ta wymusza wyszukiwanie najszybszego serwera zwierciadlanego.
- deltarpm=0



```
[main]
http_ca_trust=True
installonly_limit=3
strict_requirements_on_remove=True
best=False
skip_if_unavailable=True
fastestmirror=1
max_parallel_downloads=7
deltarpm=0

cat > /etc/dnf/dnf.conf
[main]
http_ca_trust=True
installonly_limit=3
strict_requirements_on_remove=True
best=False
skip_if_unavailable=True
fastestmirror=1
max_parallel_downloads=7
deltarpm=0

cat > /etc/dnf/dnf.conf
```

Rysunek 24: Dodanie wpisów do /etc/dnf/dnf.conf aby przyśpieszyć działanie menadżera pakietów dnf

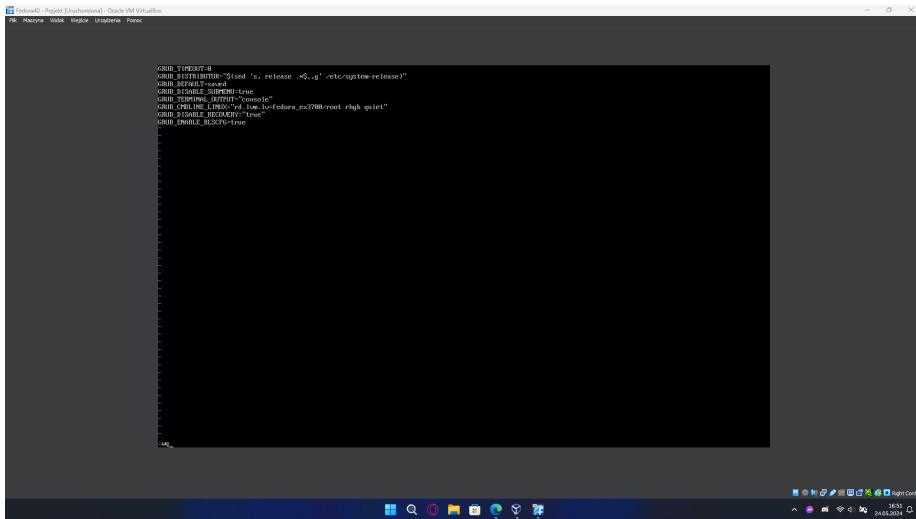
Teraz po skonfigurowaniu menadżera pakietów można wykonać aktualizację pakietów.

Rysunek 25: Aktualizacja pakietów systemowych – test konfiguracji dnf

Po aktualizacji pakietów postanowiłem edytować irytującą mnie rzeczą tj. uruchamianie się grubą przy jednym systemie operacyjnym. Na poniższym zdjęciu jest plik /etc/default/grub oryginalny (przed modyfikacją)

Rysunek 26: plik /etc/default/grub przed zmianą

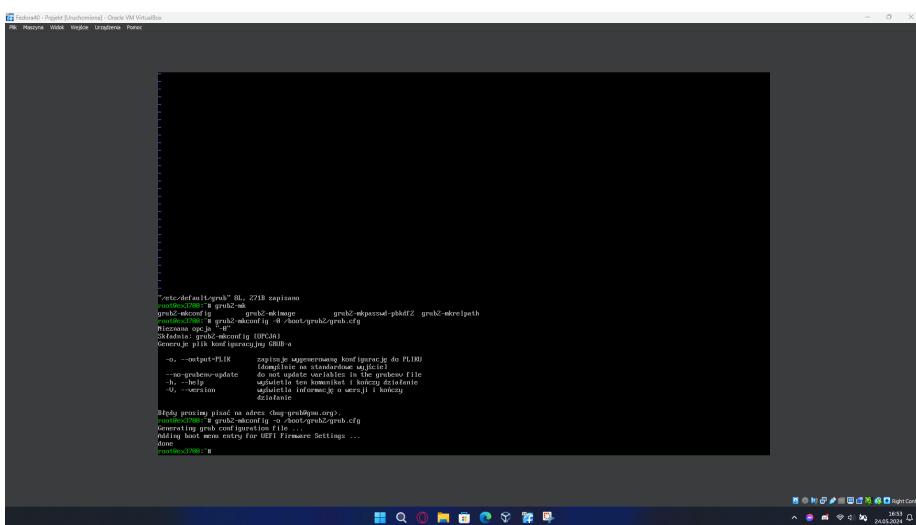
W kolejnym kroku zmieniłem GRUB\_TIMEOUT=5 na GRUB\_TIMEOUT=0  
Co można zauważyć poniższym zdjęciu.



Rysunek 27: plik /etc/default/grub po zmianie.

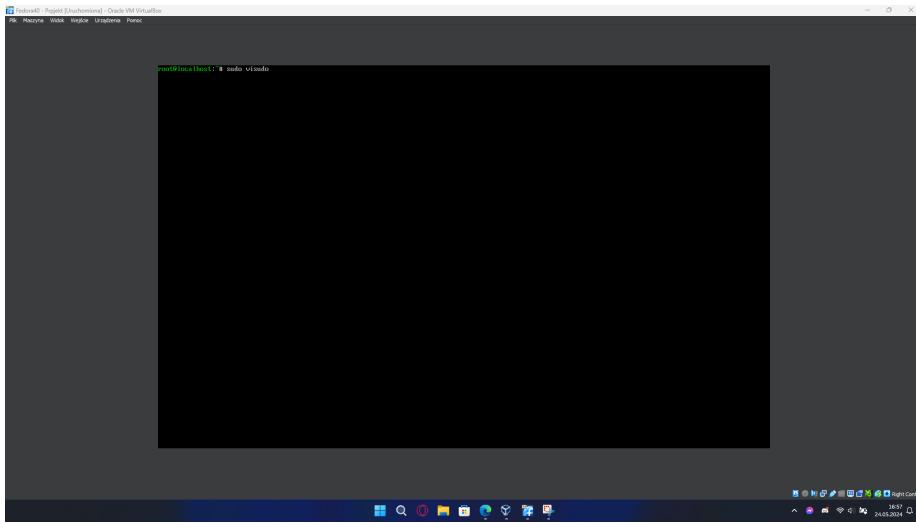
Aby zatwierdzić zmiany należy użyć komendy:

```
grub2-mkconfig -o /boot/grub2/grub2.cfg
```



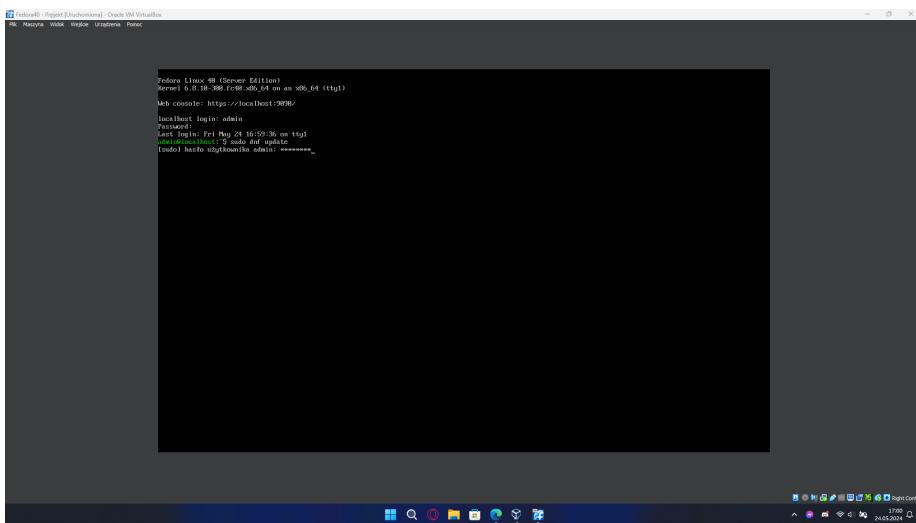
Rysunek 28: Zastosowanie zmian po edycji grub

W kolejnym kroku postanowiłem ułatwić wpisywanie hasła, gdy korzystam z sudo.



Rysunek 29: Zwiększenie wygody wpisywania haseł – edycja pliku komendą sudo visudo

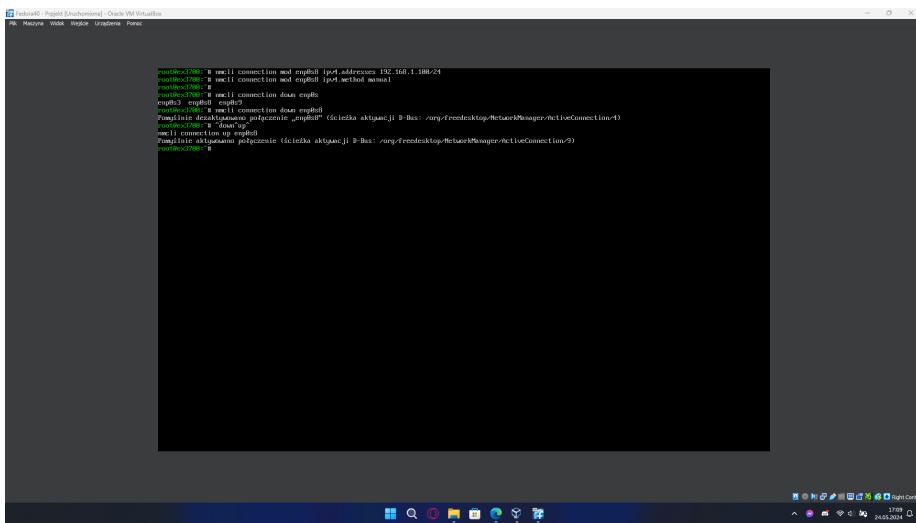
Efekt powyższego kroku:



Rysunek 30: Zwiększenie wygody wpisywania haseł – efekt działania po zmianach

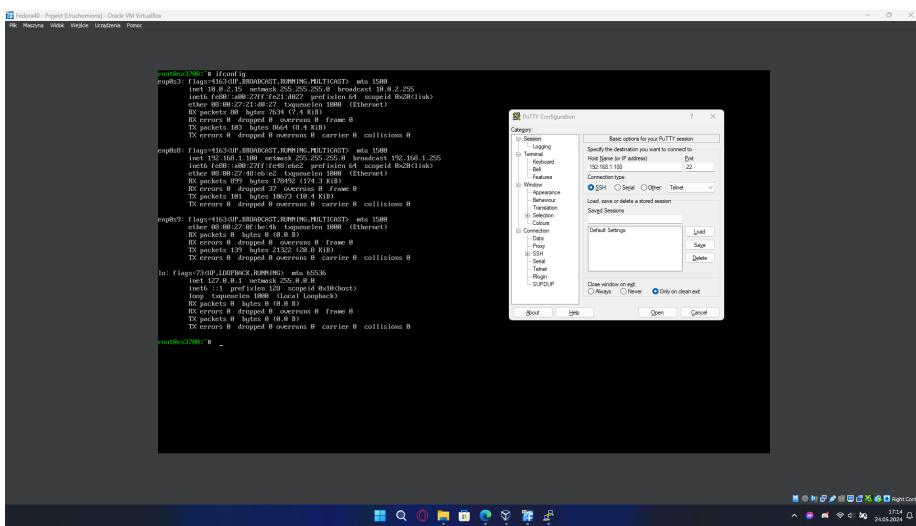
## 4.3 Konfiguracja SSH

Aby umożliwić połączenie z SSH na serwerze (VirtualBox) w pierszej kolejności potrzeba jest ustawienie poprawnego adresu IP z sieci lokalnej dla karty ustawionej na sieć mostkowaną (w moim przypadku jest to enp0s8)



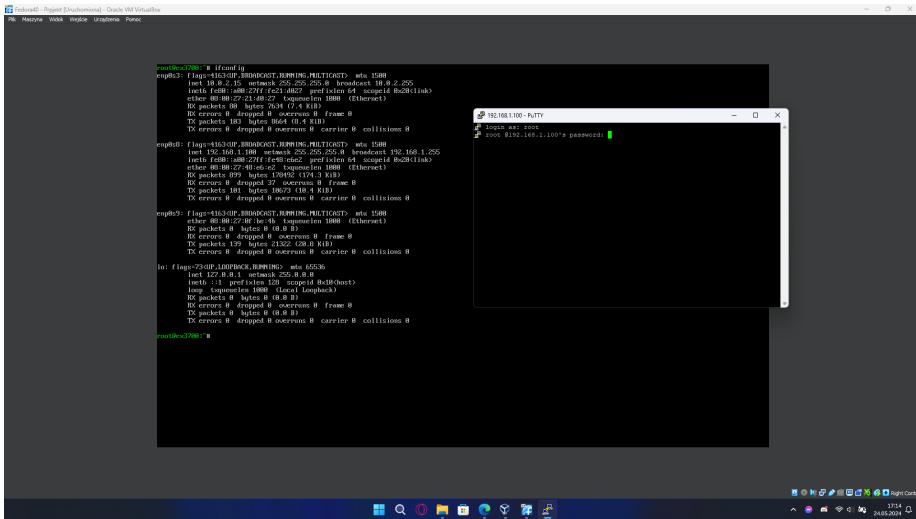
Rysunek 31: konfiguracja karty sieciowej

W serwerze Fedora 40 SSH jest domyślnie włączone i skonfigurowane. Wystarczy tylko się połączyć



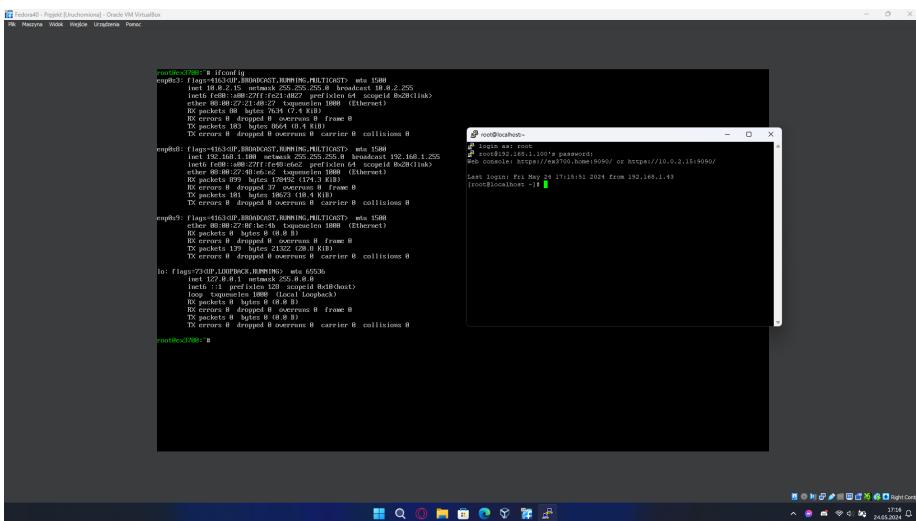
Rysunek 32: Konfiguracja aplikacji PuTTY

Próba zalogowania na konto root'a:



Rysunek 33: Podlaczenie poprzez PuTTY na konto root'a

Wynik powyższego kroku:

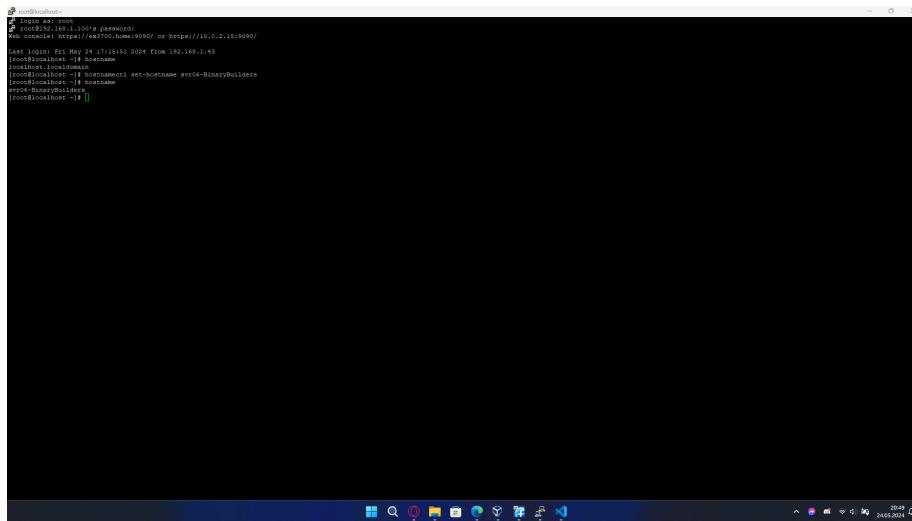


Rysunek 34: Wynik połączenia poprzez PuTTY

## 4.4 Nazwa serwera – hostname

Aby zmienić nazwę serwera (hostname) można użyć komendy:

```
hostnamectl set-hostname nazwa-komputera
```

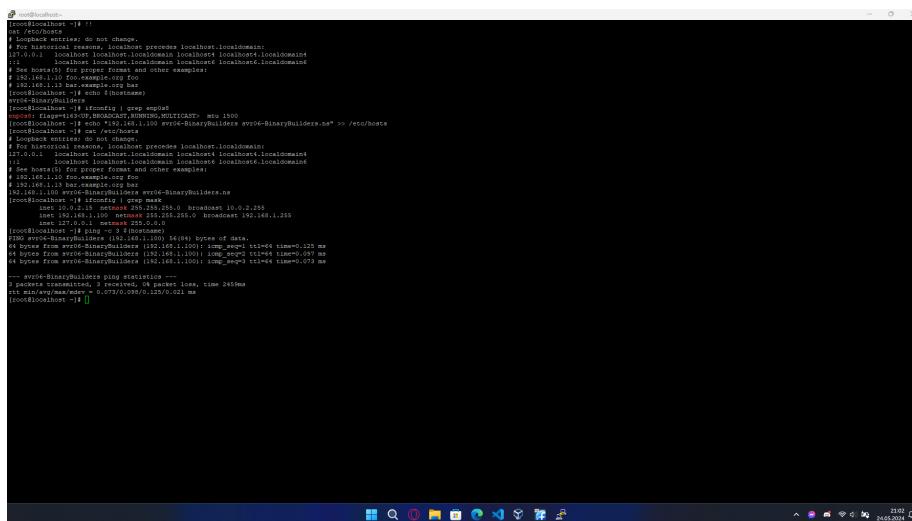


Rysunek 35: Zmiana nazwy serwera

## 4.5 DNS – instalacja i konfiguracja

Pierwszym krokiem w konfiguracji DNS jest dodanie odpowiedniego wpisu do /etc/hosts. W moim przypadku jest to:

```
192.168.230.1 svr06-BinaryBuilders svr06-BinaryBuilders.ns
```



Rysunek 36: Edycja /etc/hosts

Aby zainstalować oprogramowanie do stworzenia serwera DNS należy wydać polecenie:

- Jeśli jesteś na koncie root:

```
dnf install bind bind-utils -y
```

- jeżeli jesteś na innym koncie ale jesteś w grupie sudoers:

```
sudo dnf install bind bind-utils -y
```

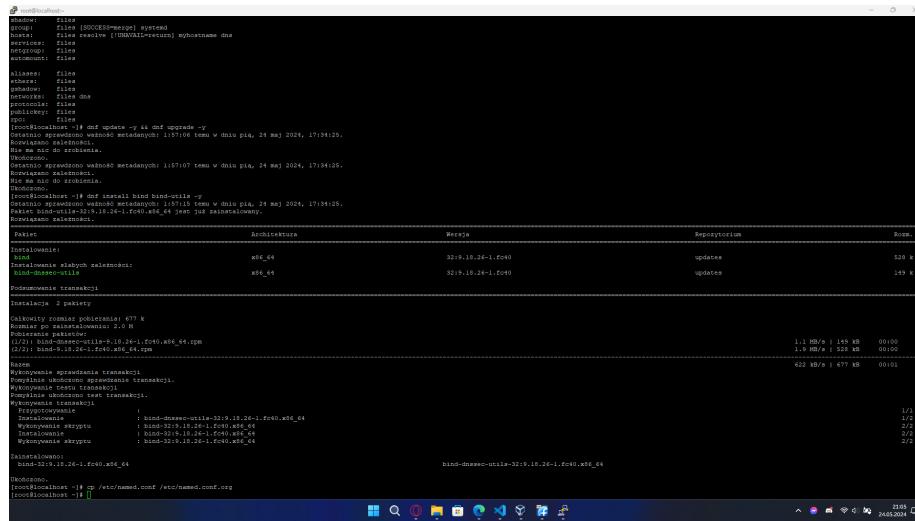
Po zainstalowaniu wymaganych pakietów należy wykonać kopię zapasową plików konfiguracyjnych. Można to zrobić komendą:

- Jeśli jesteś na koncie root:

```
cp /etc/named.conf /etc/named.conf.org
```

- jeżeli jesteś na innym koncie ale jesteś w grupie sudoers:

```
sudo cp /etc/named.conf /etc/named.conf.org
```



Rysunek 37: Instalacja DNS

Następnie trzeba skonfigurować plik /etc/named.conf. Można zrobić to komendą:

```
sudo nano /etc/named.conf
```

```

root@cockhost:~# cp /etc/named.conf /etc/named.conf.org
[  0%] [  1%] [  2%] [  3%] [  4%] [  5%] [  6%] [  7%] [  8%] [  9%]
[ 10%] [ 11%] [ 12%] [ 13%] [ 14%] [ 15%] [ 16%] [ 17%] [ 18%] [ 19%]
[ 20%] [ 21%] [ 22%] [ 23%] [ 24%] [ 25%] [ 26%] [ 27%] [ 28%] [ 29%]
[ 30%] [ 31%] [ 32%] [ 33%] [ 34%] [ 35%] [ 36%] [ 37%] [ 38%] [ 39%]
[ 40%] [ 41%] [ 42%] [ 43%] [ 44%] [ 45%] [ 46%] [ 47%] [ 48%] [ 49%]
[ 50%] [ 51%] [ 52%] [ 53%] [ 54%] [ 55%] [ 56%] [ 57%] [ 58%] [ 59%]
[ 60%] [ 61%] [ 62%] [ 63%] [ 64%] [ 65%] [ 66%] [ 67%] [ 68%] [ 69%]
[ 70%] [ 71%] [ 72%] [ 73%] [ 74%] [ 75%] [ 76%] [ 77%] [ 78%] [ 79%]
[ 80%] [ 81%] [ 82%] [ 83%] [ 84%] [ 85%] [ 86%] [ 87%] [ 88%] [ 89%]
[ 90%] [ 91%] [ 92%] [ 93%] [ 94%] [ 95%] [ 96%] [ 97%] [ 98%] [ 99%]
[ 100%] bind-dnssec-util... 3219.10.26-1.fc40.x86_64
[ 100%] bind-dnssec-util... 3219.10.26-1.fc40.x86_64

```

Rysunek 38: Stworzenie kopii zapasowej pilku konfiguracyjnego DNS

```

options {
    listen-on port 53 { 127.0.0.1; 192.168.230.1; };
    listen-on-v6 port 53 { ::1; };
    directory "/var/named";
    dump-file "/var/named/data/cache_dump.db";
    statistics-file "/var/named/data/named_stats.txt";
    memstatistics-file "/var/named/data/named_mem_stats.txt";
    secroots-file "/var/named/data/named.secroots";
    recursing-file "/var/named/data/named.recursing";
    allow-query { 127.0.0.1; 192.168.230.0/24;};

    /*
     - If you are building an AUTHORITATIVE DNS server, do NOT enable
     ↳ recursion.
     - If you are building a RECURSIVE (caching) DNS server, you need to
     ↳ enable
     recursion.
     - If your recursive DNS server has a public IP address, you MUST
     ↳ enable access
     control to limit queries to your legitimate users. Failing to do so
     ↳ will
     cause your server to become part of large scale DNS amplification
     attacks. Implementing BCP38 within your network would greatly
     reduce such attack surface
    */
    recursion yes;
    /*dnssec-enable yes;*/
    dnssec-validation yes;
    managed-keys-directory "/var/named/dynamic";
    pid-file "/run/named/named.pid";
    session-keyfile "/run/named/session.key";
}

```

```

/* https://fedoraproject.org/wiki/Changes/CryptoPolicy */
include "/etc/crypto-policies/back-ends/bind.config";
};

logging {
    channel default_debug {
        file "data/named.run";
        severity dynamic;
    };

    channel queries_log {
        file "/var/named/queries.log" versions 600 size 20m;
        print-time yes;
        print-category yes;
        print-severity yes;
        severity info;
    };
    category queries { queries_log; };
};

view "internal" {
    match-clients {
        localhost;
        192.168.230.0/24;
    };
};

zone "." IN {
    type hint;
    file "named.ca";
};

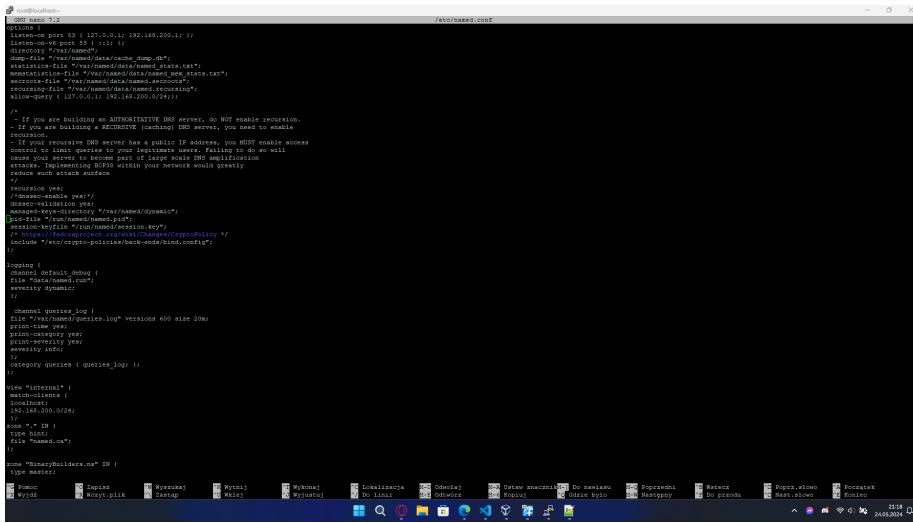
zone "BinaryBuilders.ns" IN {
    type master;
    file "BinaryBuilders.ns.lan_in";
    allow-update { none; };
};

zone "230.168.192.in-addr.arpa" IN {
    type master;
    file "230.168.192.lan_in";
    allow-update { none; };
    include "/etc/named.rfc1912.zones";
    include "/etc/named.root.key";
};

```

Powyżej znajduje się zawartość pliku /etc/named.conf, którą należy wprowadzić.

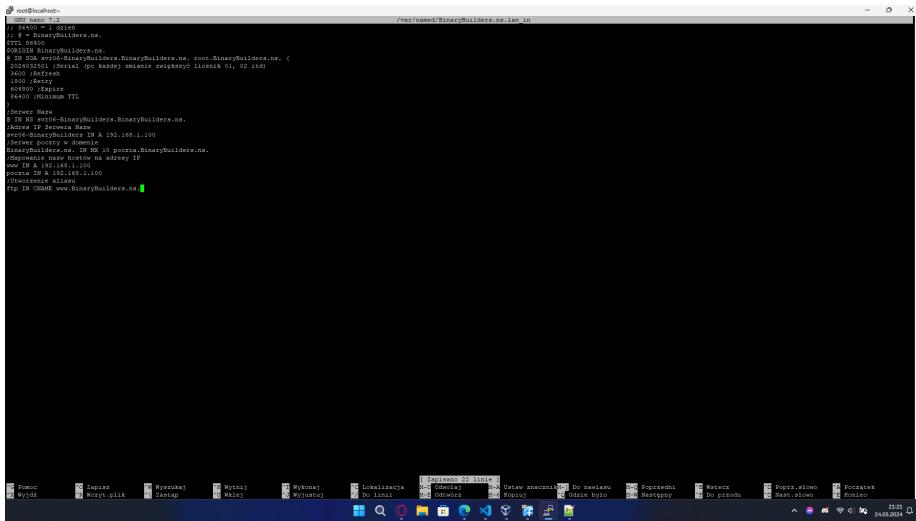
W kolejnym kroku trzeba utworzyć plik strefy podstawowej. W moim przypadku jest to plik /var/BinaryBuilders.ns.lan.in. Zawartość tego pliku:



Rysunek 39: zawartość named.conf

```
;; 86400 = 1 dzień
;; @ = BinaryBuilders.ns.
$TTL 86400
$ORIGIN BinaryBuilders.ns.
@ IN SOA svr06-BinaryBuilders.BinaryBuilders.ns.
    → root.BinaryBuilders.ns. (
        2024032502 ;Serial (po każdej zmianie zwiększyć licznik 01,
        → 02 itd)
        3600 ;Refresh
        1800 ;Retry
        604800 ;Expire
        86400 ;Minimum TTL
    )
;Serwer Nazw
@ IN NS svr06-BinaryBuilders.BinaryBuilders.ns.
;Adres IP Serwera Nazw
svr06-BinaryBuilders IN A 192.168.230.1
;Serwer poczty w domenie
BinaryBuilders.ns. IN MX 10 poczta.BinaryBuilders.ns.
;Mapowanie nazw hostów na adresy IP
www IN A 192.168.230.1
poczta IN A 192.168.230.1
sfs IN A 192.168.230.1
;Utworzenie aliasu
ftp IN CNAME www.BinaryBuilders.ns.
```

W kolejnym kroku trzeba utworzyć plik strefy dla przeszukiwania wstecznego. W moim przypadku jest to plik /var/230.168.192.lan.in. Zawartość tego pliku:



Rysunek 40: zawartość pliku strefy podstawowej

```

$TTL 86400
@ IN SOA svr06-BinaryBuilder.BinaryBuilders.ns.
    root.BinaryBuilders.ns. (
        2023032902 ;Serial (po każdej zmianie zwiększyć licznik
        01,02 itd.)
        3600 ;Refresh
        1800 ;Retry
        604800 ;Expire
        86400 ;Minimum TTL
)
;Serwer Nazw
@ IN NS svr06-BinaryBuilder.BinaryBuilders.ns.
svr06-BinaryBuilder.BinaryBuilders.ns. IN A 192.168.230.1
;Odwrotny wpis dla Serwera Nazw
1 IN PTR svr06-BinaryBuilder.BinaryBuilders.ns.
;PTR adresów IP danych hostów
1 IN PTR poczta.BinaryBuilders.ns.
1 IN PTR www.BinaryBuilders.ns.

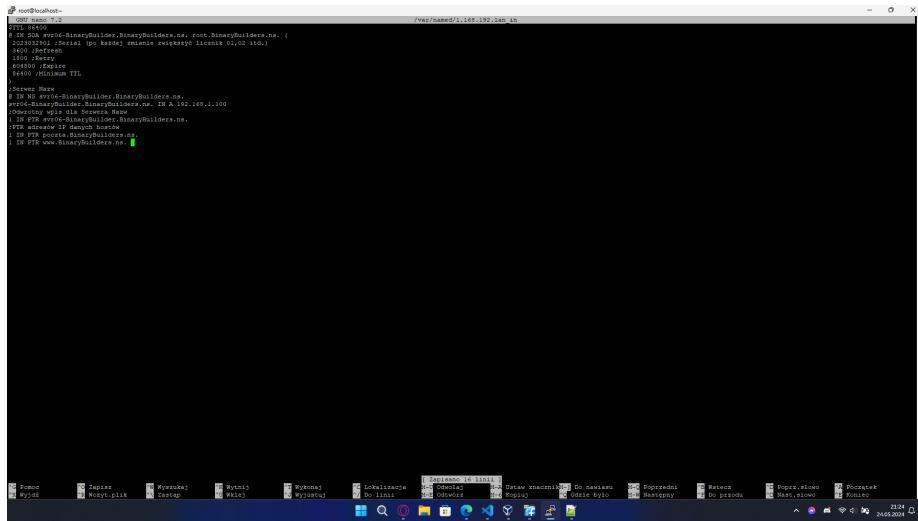
```

Następnym krokiem jest uruchomienie kilku komend:

```

systemctl start named
systemctl enable named
firewall-cmd --add-service=dns --permanent
firewall-cmd --reload
nmcli con mod enp0s9 ipv4.dns 192.168.230.1
nmcli con down enp0s9 && nmcli con up enp0s9
rndc reload
rndc status

```



Rysunek 41: zawartość pliku strefy dla przeszukiwania wstecznego

Wytłumaczenie powyższych poleceń:

- `systemctl start named`

## Komenda ta uruchomi usługe

- `systemctl enable named`

Polecenie to spowoduje że usługa będzie uruchamiana automatycznie przy włączeniu serwera.

- `firewall-cmd --add-service=dns --permanent`

Dodaje reguły zapory sieciowej, aby na stałe zezwalać na ruch DNS.

- `firewall-cmd --reload`

Przeładowuje ustawienia zapory sieciowej, aby zastosować wprowadzone zmiany.

- nmcli con mod enp0s9 ipv4 dns 192.168.230.1

Modyfikuj połaczenie `enp0s9`, aby używało serwera DNS o adresie 192.168.230.1

- nmcli con down enp0s9 & & nmcli con up enp0s9

Dezaktywuje i ponownie aktywuje połączenie sieciowe enp0s9

- ## • undo/reload

Przeładowuję konfigurację serwera serwera DNS

- ### • `rndc status`

[www.virtla.status.sorvare.sorvare.DNS](#)

Jak widać na zrzucie ekranu powyżej miałem problemy z błędna konfiguracją jednego z pliku, jednakże udało mi się naprawić problem i uruchomić usługę DNS. Kolejnym i ostatnim krokiem jest test usługi DNS. Wykonać go można korzystając z drugiej maszyny wirtualnej. Przykładowy test DNS możesz zobaczyć tutaj.

Rysunek 42: Uruchomienie usługi DNS

#### 4.6 DHCP – instalacja i konfiguracja

Aby zainstalować oprogramowanie do stworzenia serwera DHCP należy wydać polecenie:

```
sudo dnf install -y dhcp-server
```

Rysunek 43: Instalacja DHCP

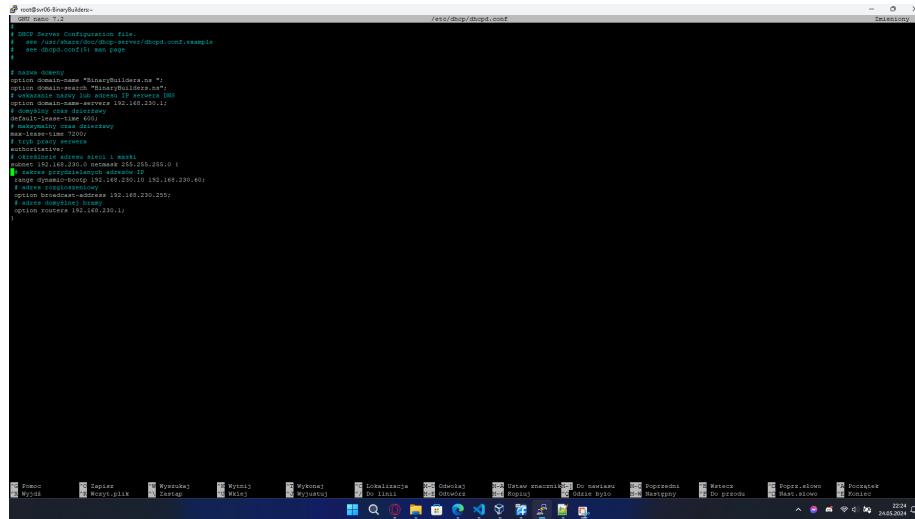
Po zainstalowaniu odpowiednich pakietów dobrze jest wykonać kopię zapasową oryginalnego pliku konfiguracji, co widać na zrucie powyżej. Można to zrobić następującą komendą:

```
cp /etc/dhcp/dhcpd.conf /etc/dhcp/dhcpd.conf.org
```

W kolejnym kroku należy wprowadzić zmiany w pliku konfiguracyjnym DHCP tj. /etc/dhcp/dhcpd.conf. W moim przypadku:

```
# nazwa domeny
option domain-name "BinaryBuilders.ns ";
option domain-search "BinaryBuilders.ns";
# wskazanie nazwy lub adresu IP serwera DNS
option domain-name-servers 192.168.230.1;
# domyślny czas dzierżawy
default-lease-time 600;
# maksymalny czas dzierżawy
max-lease-time 7200;
# tryb pracy serwera
authoritative;
# określne adresu sieci i maski
subnet 192.168.230.0 netmask 255.255.255.0 {
    # zakres przydzielanych adresów IP
    range dynamic-bootp 192.168.230.10 192.168.230.60;
    # adres rozgłoszeniowy
    option broadcast-address 192.168.230.255;
    # adres domyślnej bramy
    option routers 192.168.230.1;
}
```

Wprowadzoną treść widać na zrzucie poniżej.



Rysunek 44: Konfiguracja DHCP – edycja pliku /etc/dhcp/dhcpd.conf

Następnym krokiem jest wprowadzenie kilku poleceń:

```
systemctl start dhcpcd  
systemctl enable dhcpcd  
firewall-cmd --add-service=dhcp --permanent  
firewall-cmd --reload
```

Wytłumaczenie powyższych poleceń:

- systemctl start dhcpcd

Komenda ta uruchomi usługę DHCP

- systemctl enable dhcpcd

Polecenie to spowoduje że usługa DHCP będzie uruchamiana automatycznie przy włączeniu serwera

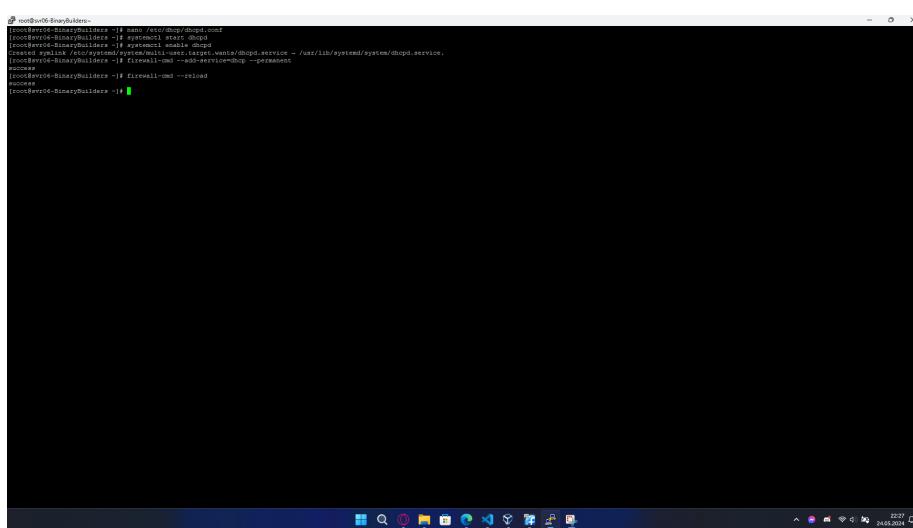
- irewall-cmd --add-service=dhcp --permanent

Dodaje regułę zapory sieciowej, aby na stałe zezwalać na ruch DHCP.

- firewall-cmd --reload

Przeładowuje ustawienia zapory sieciowej, aby zastosować wprowadzone zmiany.

Zrzut ekranu poniżej przedstawia wykonanie tych komend.



```
[root@srv06-BinaryBuilders ~]# nano /etc/dhcp/dhcpd.conf  
[root@srv06-BinaryBuilders ~]# systemctl start dhcpcd  
[root@srv06-BinaryBuilders ~]# systemctl enable dhcpcd  
[root@srv06-BinaryBuilders ~]# useradd -r -u 1000 -g 1000 -s /bin/false -c 'dhcp' ventus/dhcpd.service - /usr/lib/systemd/system/dhcpd.service  
[root@srv06-BinaryBuilders ~]# firewall-cmd --add-service=dhcp --permanent  
[root@srv06-BinaryBuilders ~]# firewall-cmd --reload  
[root@srv06-BinaryBuilders ~]#
```

Rysunek 45: Instalacja DHCP

Po powyższym kroku nie pozostaje nic innego jak przetestować działanie DHCP. Wyniki testu dostępne są [tutaj](#).

## 5 Testy działania wdrożonych usług

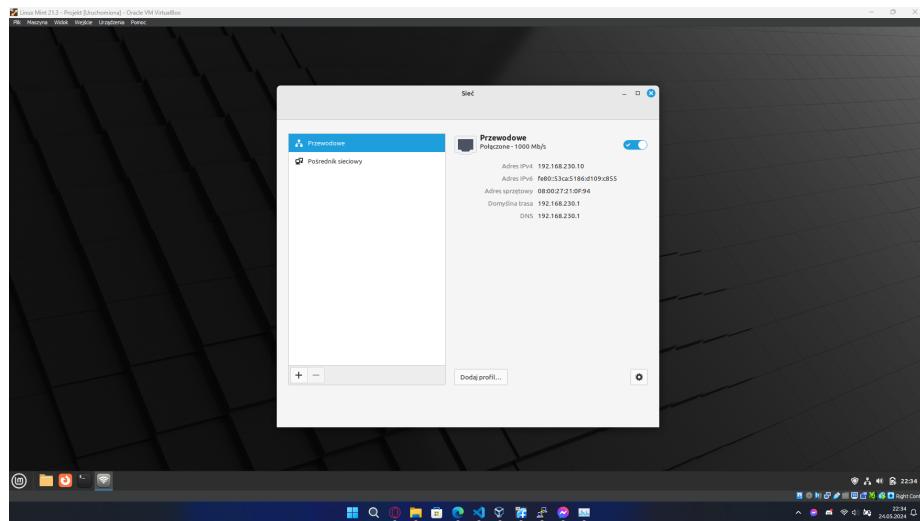
### 5.1 DNS

```
maciej@maciejs-VirtualBox:~$ dig www.BinaryBuilders.ns
; <>> SIG 9.18.18-ubuntu#22.04.2-ubuntu <>> poczta.BinaryBuilders.ns
;; options: +cmd
;; Got answer:
;; ->>> QUERY: poczta.BinaryBuilders.ns, status: NOERROR, id: 60723
;; Flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1
;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags: udp: 65494
;poczta.BinaryBuilders.ns. IN A
;; ANSWER SECTION:
poczta.BinaryBuilders.ns. 7120 IN A 192.168.230.1
;; Query time: 4 msec
;; SERVER: 127.0.0.53#53 (127.0.0.53) (UDP)
;; WHEN: Fr 24 May 24 22:51:59 CEST 2024
;; MSG SIZE rcvd: 69
maciej@maciejs-VirtualBox:~$ dig www.BinaryBuilders.ns
; <>> SIG 9.18.18-ubuntu#22.04.2-ubuntu <>> www.BinaryBuilders.ns
;; options: +cmd
;; Got answer:
;; ->>> QUERY: www.BinaryBuilders.ns, status: NOERROR, id: 47558
;; Flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1
;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags: udp: 65494
;QUESTION SECTION:
www.BinaryBuilders.ns. IN A
;; ANSWER SECTION:
www.BinaryBuilders.ns. 7120 IN A 192.168.230.1
;; Query time: 0 msec
;; SERVER: 127.0.0.53#53 (127.0.0.53) (UDP)
;; WHEN: Fr 24 May 24 22:51:59 CEST 2024
;; MSG SIZE rcvd: 66
maciej@maciejs-VirtualBox:~$ ping -c 2 poczta
PING poczta.BinaryBuilders.ns (192.168.230.1): 56(84) bytes of data.
64 bytes from 192.168.230.1: icmp_seq=1 ttl=64 time=1.46 ms
64 bytes from 192.168.230.1: icmp_seq=2 ttl=64 time=1.26 ms
--- poczta.BinaryBuilders.ns ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 100ms
rtt min/avg/max/mdev = 1.229/1.359/1.469/0.392 ms
maciej@maciejs-VirtualBox:~$
```

Rysunek 46: Test DNS

Jak widać na powyższym zdjęciu system w sieci wewnętrznej dostaje odpowiedzi od serwera na zapytania, co sugeruje że usługa DNS została skonfigurowana poprawnie.

### 5.2 DHCP



Rysunek 47: Instalacja DHCP

Jak widać na powyższym zdjęciu karta w systemie klienta ustawiona na

sieć wewnętrzna dostała poprawny adres IP, adres bramy domyślnej i DNS. Na zdjęciu również widać że pula DHCP działa poprawnie.

## **6 Kod skryptu BASH, oraz tablica crontab**

FFFFFFFFFFFFn

## **7 Wnioski**

EEEEEEEEE

## 8 Literatura

- [1] *Kubernetes Blog*. Dostęp: 2024-01-19. URL: <https://kubernetes.io/blog/>.
- [2] *Kubernetes Documentation*. Dostęp: 2024-01-19. URL: <https://kubernetes.io/docs/>.
- [3] *Kubernetes GitHub Repository*. Dostęp: 2024-01-19. URL: <https://github.com/kubernetes/kubernetes>.