1.Konfigurujemy karty sieciowe na maszynie z Ubuntu

* pierwsza karta sieciowa – zmostkowana

* druga karta sieciowa – wewnętrzna intnet

jeżeli posiadamy, możemy przywrócić odpowiednią migawkę z kartami już skonfigurowanymi

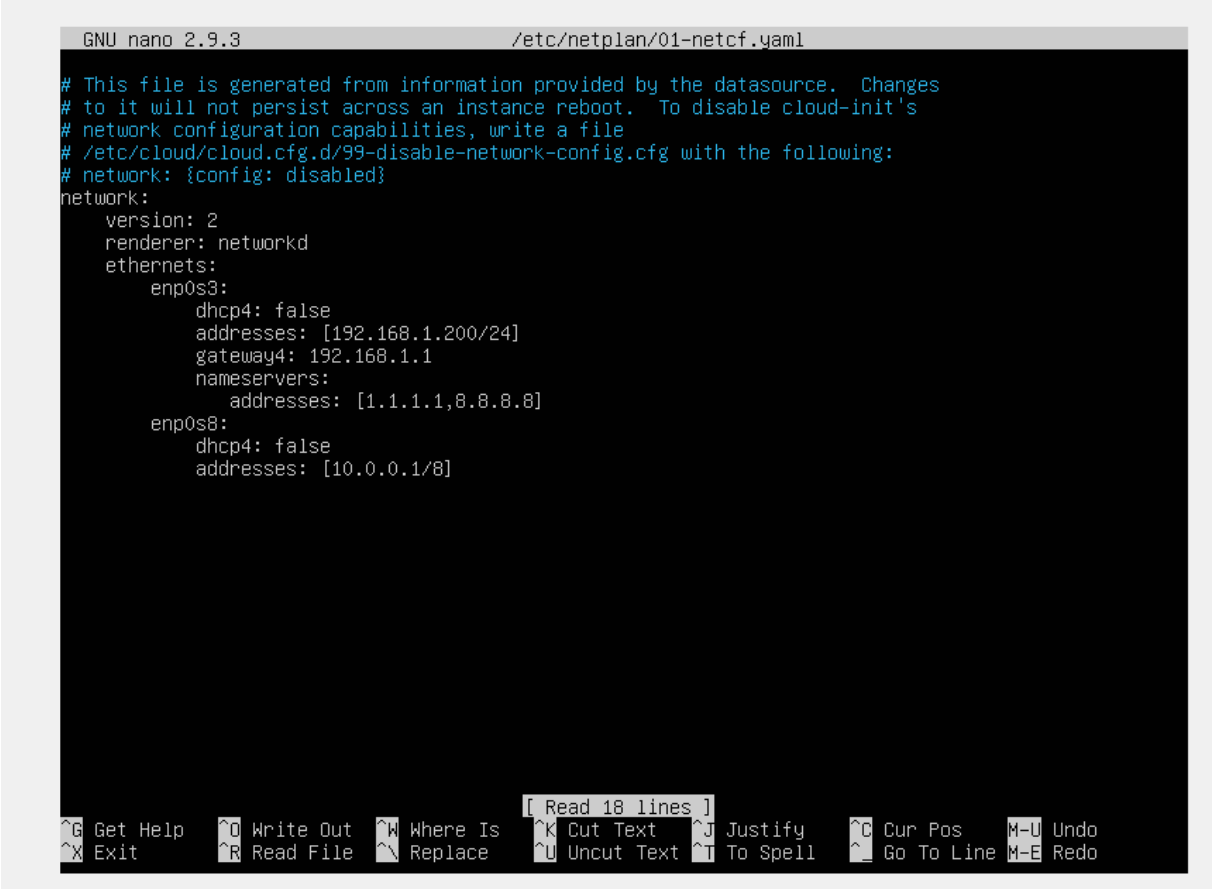
2. Konfigurujemy karty sieciowe na maszynie testowej z WinXP:

* pierwsza karta sieciowa - zmostkowana

3. Uruchamiamy maszynę z Ubuntu , sprawdzamy ustawienia adresów IP

nano /etc/netplan/01-netcf.yaml

ustawiamy na pierwszej sieciówce adres pasujący do adresacji w naszej sieci domowej, u mnie to:



po ustawieniu odpowiednich adresów zatwierdzamy ustawienia za pomocą **netplan apply** i sprawdzamy za pomocą **ip a,** spradzamy tez za pomocą polecenia ping czy mamy łączoność z internetem.

4. Aktualizujemy zawartość repozytoriów:

sudo apt update

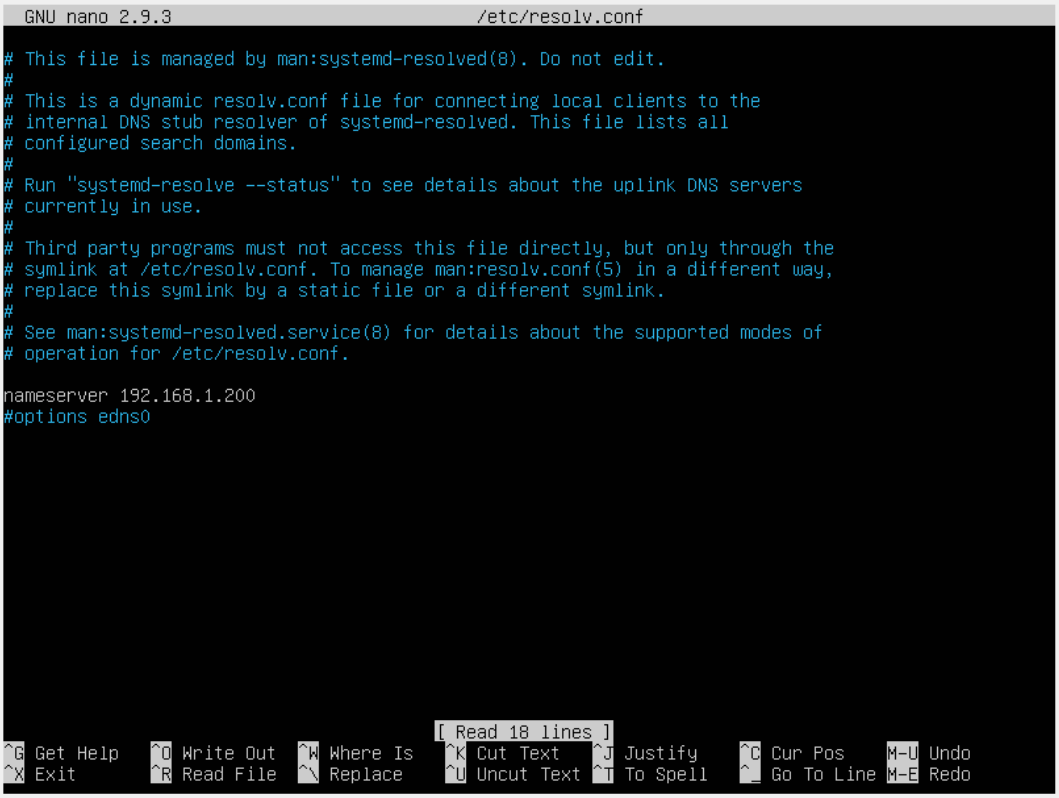
5. Instalujemy serwer DNS:

sudo apt install bind9

Folder gdzie serwer DNS będzie trzymał swoje pliki konfiguracyjne to:

/etc/bind/

6. Ustawiamy w pliku /etc/resolv.conf nasz adres IP jako adres serwera DNS:



7. Konfiguracja serwera DNS:

Serwer DNS ma następujące pliki konfiguracyjne:

- ***/etc/bind/named.conf*** - główny plik konfiguracyjny zawierający instrukcje dołączania pozostałych plików konfiguracyjnych
- ***/etc/bind/named.conf.options*** - opcje serwera DNS
- ***/etc/bind/named.conf.local*** - definicje stref lokalnych
- ***/etc/bind/named.conf.default-zones*** - konfiguracja domyślnych stref (localhost, serwery główne - root servers)

Uwaga: w plikach konfiguracyjnych ***named.conf.**** znaki // oznaczają komentarz jednolinijkowy.

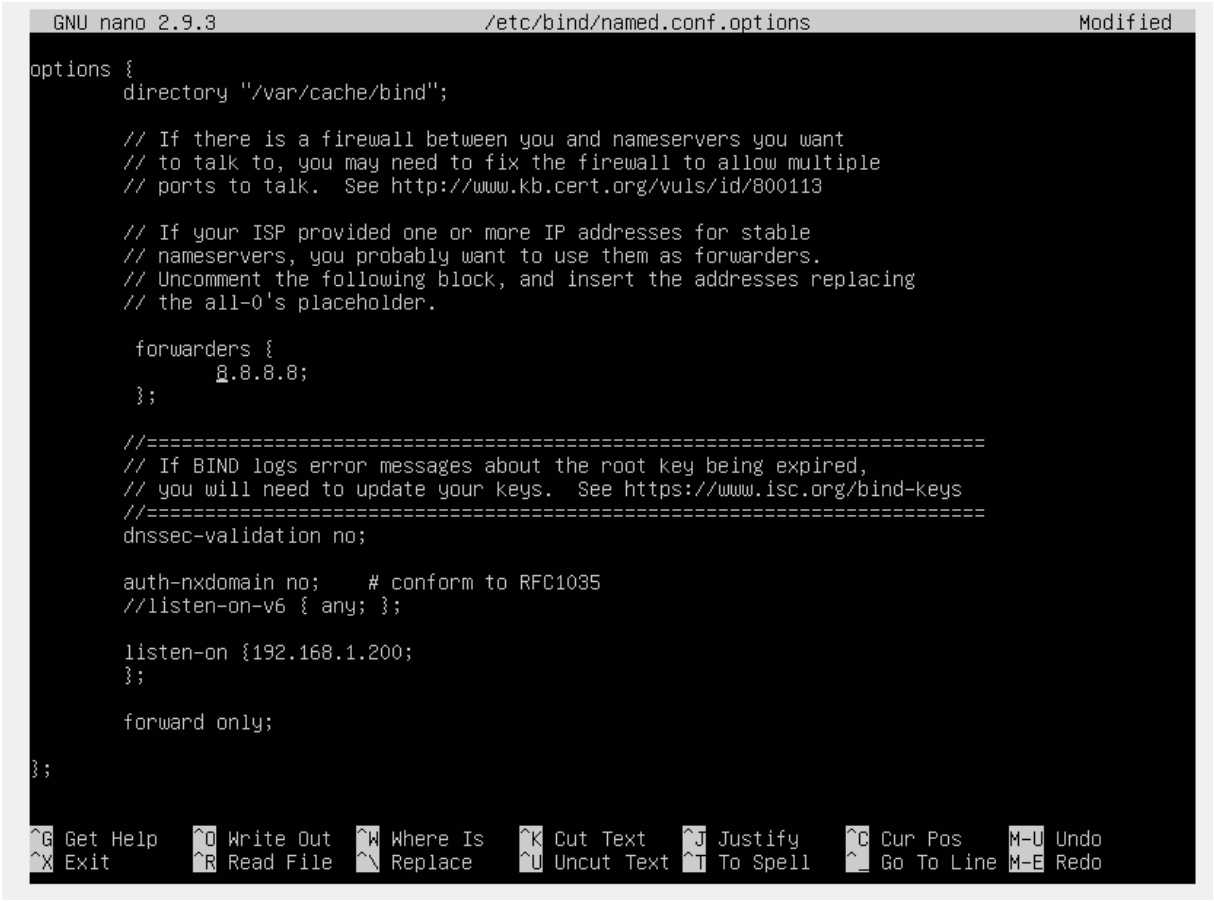
Na początku trzeba zacząć od zrobienia kopii zapasowych tych plików:

cp /etc/bind/named.conf /etc/bind/named.conf.bac
cp /etc/bind/named.conf.options /etc/bind/named.conf.options.bac
cp /etc/bind/named.conf.local /etc/bind/named.conf.local.bac
cp /etc/bind/named.conf.default.zones /etc/bind/named.conf.default.zones.bac

Teraz możemy przystąpić do faktycznej konfiguracji:

W pliku ***/etc/bind/named.conf.options*** określa się globalną konfigurację serwera DNS.

ustawiamy tutaj na którym interfejsie ma działać nasz serwer DNS, adresy serwerów DNS do których będzie przekazywał zapytania których sam nie potrafi rozwiązać, czy też reguły ACL kto może się odpytywać naszego serwera DNS – ja ustawiłem że serwer ma działac na adresie 192.168.1.200 i ma się odpytywać serwer DNS googla, jeżeli czego nie będzie wiedział.



Inna przykładowa konfiguracja (z kursu na moodle) z opisem parametrów – oczywiście jeżeli będziemy jej chcieli użyć to musimy dopasować do naszej adresacji:

Plik named.conf.options

```
#utworzenie grupy ZaufaniKlienci mogącej odpytywać ten serwer DNS
acl ZaufaniKlienci {
        192.168.0.0/24;
        localhost;
        localnets; #lokalne sieci serwera (wszystkie interfejsy)
};

options {
#lokalizacja pamięci podręcznej serwera DNS:
        directory "/var/cache/bind";

#zezwolenie na odpytywanie tego serwera DNS tylko przez określonych klientów (np. z utworzonej grupy):
        allow-query {
         ZaufaniKlienci;
         10.0.0.0/8;
        };

#Adresy IP serwerów DNS, do których będą wysyłane dalsze zapytania.
        forwarders {
          8.8.8.8;
          8.8.4.4;
        };

#wybranie adresów IP, na których serwer DNS będzie nasłuchiwał:
        listen-on {
         192.168.0.1;
        };

#forward {only | first}
#forward only   - zapytania przesyłane są wyłącznie do adresów określonych w sekcji forwarders
#forward first  - (domyślnie) jeżeli serwery z sekcji forwarders nie są wstanie udzielić odpowiedzi,
#                 serwer podejmie próbę rozwiązania nazwy np. za pomocą serwerów najwyższego poziomu (root)
        forward only;
};
```

Sprawdzanie poprawności konfiguracji.

Polecenie ***named-checkconf*** sprawdza plik konfiguracyjny serwera DNS bind9 (***/etc/bind9/named.conf.options***) i wyświetla ewentualne błędy.

8. Utworzenie lokalnych stref wyszukiwania

Serwer DNS potrafi wyszukiwać do przodu – zamieniać nazwy domenowe na adresu IP oraz wyszukiwać wstecz – zamieniać IP na nazwy domenowe. Oczywiście, żeby potrafił to zrobić, musimy zdefiniować strefy wyszukiwania:

nano /etc/bind/named.conf.local

W naszym przypadku zdefiniujemy strefę wyszukiwania w przód dla domeny **zsme.local**, oraz strefę wyszukiwania wstecz dla adresów należących do sieci **192.168.1.0/24**

W pliku ***named.conf.local*** definiowane są strefy lokalne: wyszukiwania w przód ***(ang. forward zone)*** i wstecz ***(ang. reverse zone)***. Serwer bind9 definiuje całe strefy (a nie domeny / nazwy domenowe) czyli grupy nazw domen, subdomen i hostów.

Przykładowa konfiguracja stref z opisem parametrów (będziemy musieli ją przerobić tak żeby pasowała do naszej adresacji:)

Definicja strefy wyszukiwania w przód i wstecz.

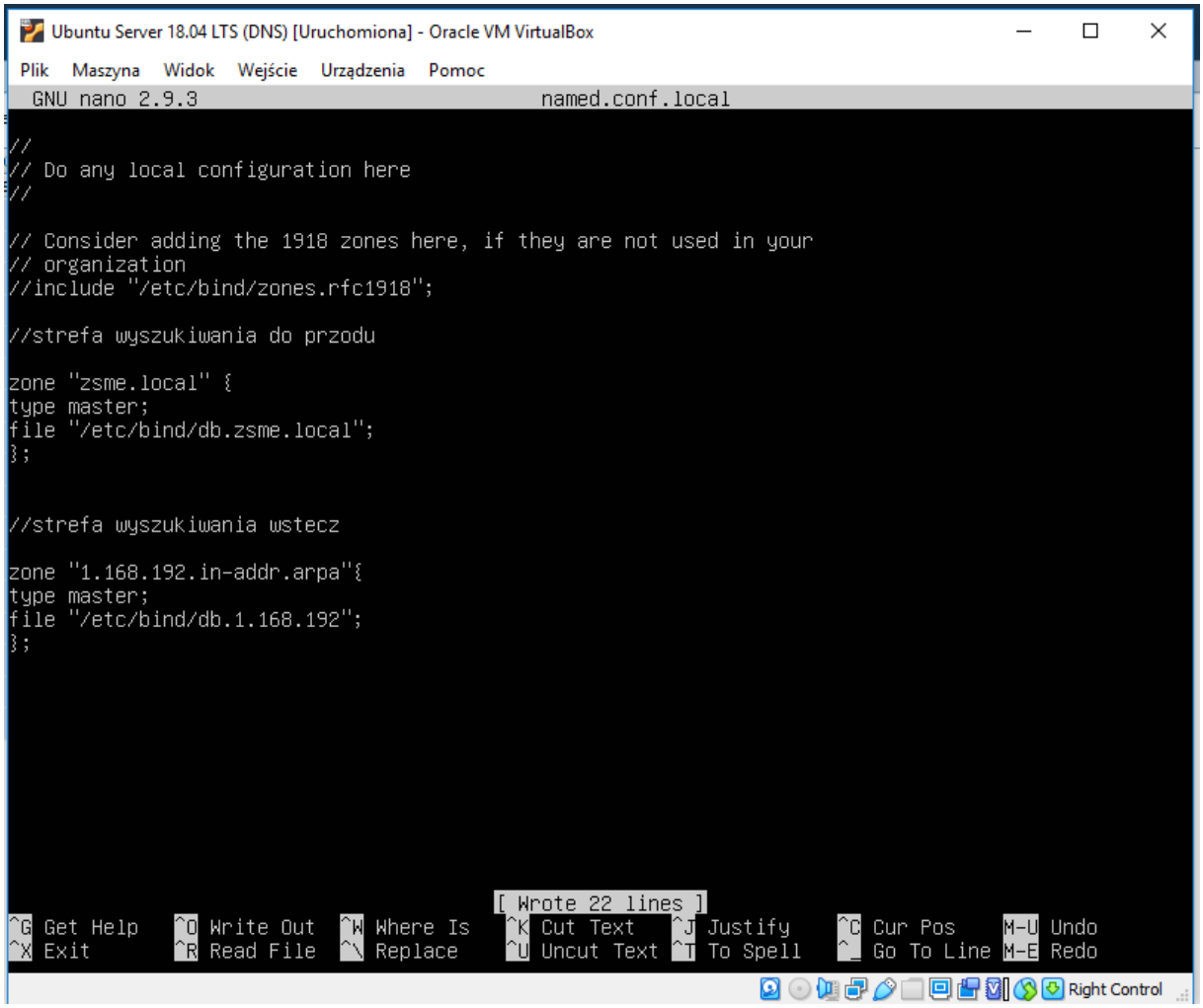
Strefa wyszukiwania w przód: named.conf.local CodeHL 1.7 **[Command Prompt]**

```
#definicja strefy wyszukiwania w przód np.: zsme.local - 192.168.0.1

zone "zsme.local" {
  #typ strefy: master | slave
  type master;

  #lokalizacja pliku z definicją danej strefy
  file "/etc/bind/db.zsme.local";

  #zezwolenie na wysyłanie danych do dodatkowego serwera DNS
  #allow-transfer { 192.168.0.2; };

  #serwer dodatkowy - slave - powinien mieć zdefiniowany adres serwera głównego
  #masters { 192.168.0.1; };
};

#definicja strefy wyszukiwania wstecz np.: 192.168.0.1 - zsme.local
zone "0.168.192.in-addr.arpa" {
  #typ strefy: master | slave
  type master;

  #lokalizacja pliku z definicją danej strefy
  file "/etc/bind/db.0.168.192";

  #zezwolenie na wysyłanie danych do dodatkowego serwera DNS
  #allow-transfer { 192.168.0.2; };
};
```

Typ strefy definiowany parametrem ***type*** określa czy dany serwer jest serwerem głównym ***(master)*** czy pomocniczym ***(slave)***. Serwery pomocnicze stosowane są w celu zapewnienia ciągłej dostępności usługi DNS np. w sytuacji gdy serwer główny ulegnie awarii lub jest przeciążony. Serwer główny wysyła aktualizacje danych do serwerów dodatkowych.

Parametr ***allow-transfer*** definiuje adresy IP dodatkowych serwerów DNS przechowujących kopie danej strefy.

W naszym przypadku plik z definicją stref będzie wyglądał tak:



Zwróćcie uwagę na sposób definiowania strefy wstecz – nasza przykładowa strefa wyszukiwania wstecz będzie pozwalała na wyszukiwanie rekordów dla adresów należących do sieci **192.168.1.0/24** – w definicji są wpisane pierwsze trzy oktety, więc dodając rekordy do strefy będziemy wpisywać tylko ostatni oktet adresu. Jeżeli na przykład chcielibyśmy zrobić strefę wyszukiwania wstecz dla adresów z sieci **10.0.0.0/8**, to w definicja strfy wstecz wyglądała by następująco:

```
zone "10.in-addr.arpa" {
      #typ strefy: master | slave
       type master;
       file "/etc/bind/db.10";
};
```

9. Utworzenie plików z konfiguracja stref

Musimy stworzyć pliki w których będziemy dodawać wpisy do stref, nazwy i ich położenie muszą być takie jak zdefiniowaliśmy dla poszczególnych stref w pliku named.conf.local. Żeby nie musieć tworzyć plików od zera, możemy skopiować pliki z przykładową konfiguracją stref które są już w katalogu /etc/bind. Plik konfiguracji strefy zsme.local stworzymy na podstawie db.local

cp /etc/bind/db.local /etc/bind/db.zsme.local

a plik konfiguracji dla strefy wyszukiwania wstecz stworzymy na podstawie pliku db.127

cp /etc/bind/db.127 /etc/bind/db.1.168.192

Teraz możemy przejść do konfiguracji poszczególnych opcji w plikach:

Konfiguracja **strefy wyszukiwania do przodu** dla domeny **zsme.local**:

zdefiniujemy wpisy dotyczące hostów:

zsme.local ma pokazywać na adres 192.168.1.200
ns1.zsme.local ma pokazywać na adres 192.168.1.200 i jest też serwerem DNS (rekord typu NS)
xp.zsme.local ma pokazywać na adres 1.150 – nasz testowy komputer z XP
komp1.zsme.local ma pokazywać na to samo co xp.zsme.local – rekord typu CNAM (alias)

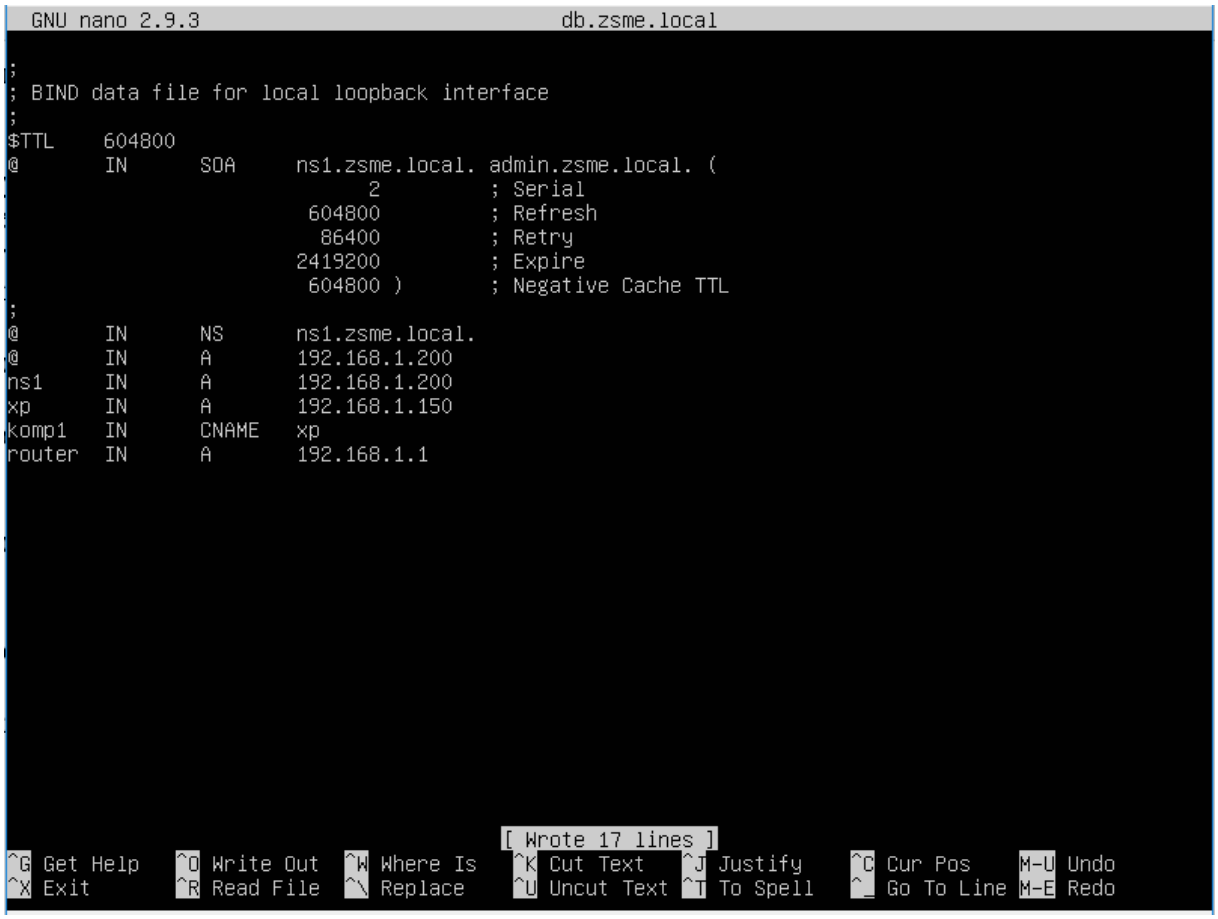
router.zsme.local – ma pokazywać na adres naszego routera – 192.168.1.1

W plikach z definicjami stref mogą być stosowane znaki o specjalnym znaczeniu:

Znak średnika ; oznacza komentarz jednolinijkowy

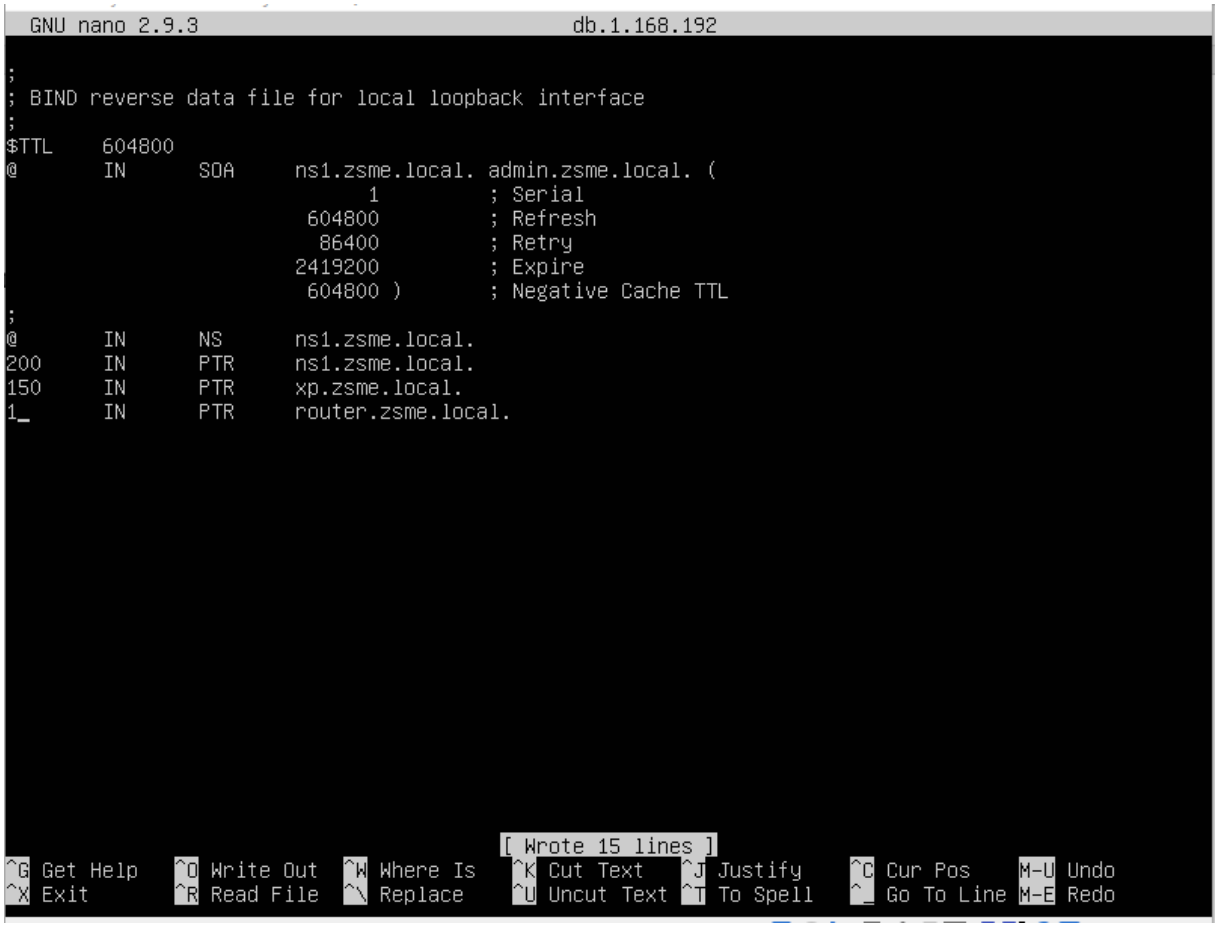
Symbol @ - oznacza aktualnie definiowaną strefę (w tym przypadku zsme.local.)

Kropka - na końcu nazwy rekordu oznacza koniec dołączania. Jeżeli jej nie będzie, serwer dołączy do niej nazwę strefy np.: abcd jest równoznaczne z: abcd.zsme.local.



Konfiguracja **strefy wyszukiwania wstecz** dla sieci **192.168.1.0/24**:

Definiujemy rekordy typu PTR czyli wskaźniki (czyli dany adres IP pokazuje na podana nazwę), zwróćcie uwagę, ze podajemy tylko brakujący oktet (bo deklarując strefę w named.conf.locale deklarowaliśmy
zone "1.168.192.in-addr.arpa" - jak widać, **brakuje** nam tylko **jednego oktetu**, stąd dodając rekordy wpisujemy tylko ten oktet – **serwer sklei to w całość**.



tak jak wcześniej pisałem – jeżeli zadeklarowalibyśmy strefę wyszukiwania wstecz dla sieci 10.0.0.0/8:

```
zone "10.in-addr.arpa" {
      #typ strefy: master | slave
       type master;
       file "/etc/bind/db.10";
};
```

dla takiej strefy dodając rekordy musielibyśmy podać 3 brakujące oktety, czyli przykładowy rekord: **adres 10.0.0.100** pokazuje na nazwę **pc1.zsme.local** wyglądał by następująco:

```
100.0.0  IN        PTR      pc1.zsme.local. ; 10.0.0.100
```

Znaczenie poszczególnych parametrów:

- ***Czas ważności***- każdy rekord może mieć określony indywidualny czas ważności (TTL) czyli okres przez jaki systemy rozwiązywania nazw (pomocnicze serwery DNS, usługi cache DNS systemów operacyjnych) powinny przechowywać o nim dane.

CodeHL 1.7 **[Command Prompt]**

```
pc1     1800    IN      A       192.168.0.100 ; Czas ważności rekordu A
```

- ***$TTL*** - parametr globalny określający czas życia dla wszystkich rekordów (pod warunkiem, że nie mają określonego swojego indywidualnego czasu). Parametr ten musi znajdować się na początku linii (w pierwszej kolumnie, nie może być poprzedzony nawet spacją!). Dla każdego rekordu musi być zdefiniowany parametr TTL (globalny lub indywidualny),
- ***IN*** - typ sieci (IN - Internet),
- ***SOA*** - definicja głównego rekordu (ang. start of authority), który określa:
  - ***ns1.zsme.local.*** - nazwę głównego serwera ***DNS*** dla tej strefy,
  - ***root.zsme.local.*** - adres email osoby odpowiedzialnej za administrację strefą (***root@zsme.local*** - znak **@** zastępowany jest kropką),
  - ***serial*** - numer seryjny rekordu ***SOA***, po każdej aktualizacji powinien być zwiększany. Dostępny zakres: ***1 - 4294967295,*** liczba typu unsigned int,
  - ***refresh*** - czas kiedy serwery dodatkowe (***slave***) powinny odświeżyć informacje o strefie z serwera głównego (***master***),
  - ***retry*** - czas po jakim serwery ***slave*** ponowią próbę odświeżenia informacji jeżeli wcześniejsza próba połączenia z serwerem ***master*** się nie powiodła,
  - ***expire*** - czas po jakim serwery ***slave*** przestaną traktować dane o strefie jako autorytatywne (pewne),
  - ***negative cache TTL*** - czas przez jaki powinna być przechowywana informacja, że dany rekord (cała strefa) nie istnieje (max 10800 sekund).

Reprezentacja czasu w usłudze bind9:

Parametry czasu dla rekordów domyślnie podawane są w sekundach - 32 bitowe liczby ze znakiem. Możliwe jest użycie innych jednostek takich jak:

- ***s*** - sekundy
- ***m*** - minuty
- ***h*** - godziny
- ***d*** - dni
- ***w*** - tygodnie

Wielkość liter nie ma znaczenia.

Przykład rekordu SOA CodeHL 1.7 **[Command Prompt]**

```
@       20H     IN      SOA   ns1.zsme.local. root.zsme.local (
                        1234    ; serial
                        10800s  ; refresh = 3 godziny
                        15M     ; retry = 15 minut
                        2W12h   ; expiry = 2 tygodnie i 12 godzin
                        2h30M   ; neg. cache = 2 i pół godziny
                        )
```

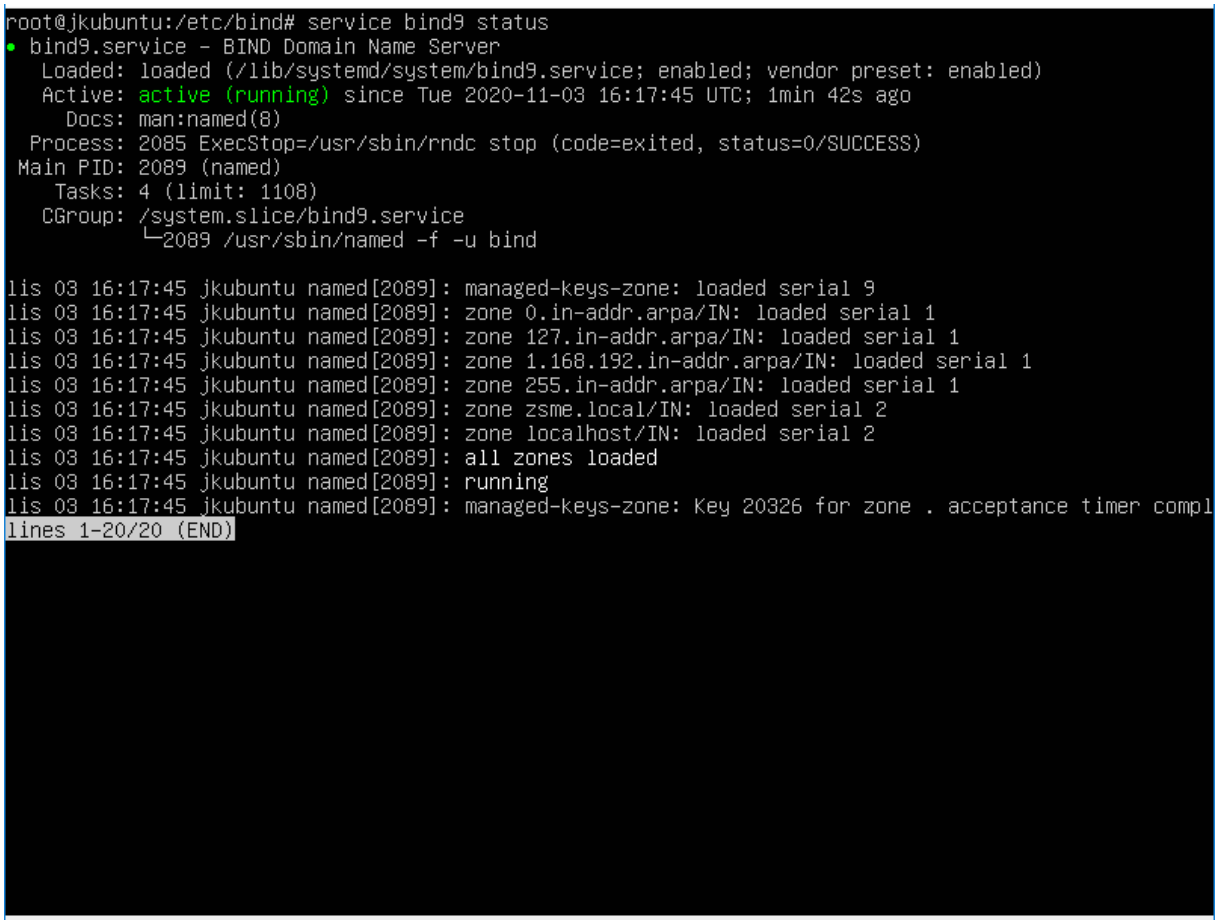
10. Restart usługi i testy

Po skonfigurowaniu stref restartujemy nasz serwer DNS:

service bind9 restart

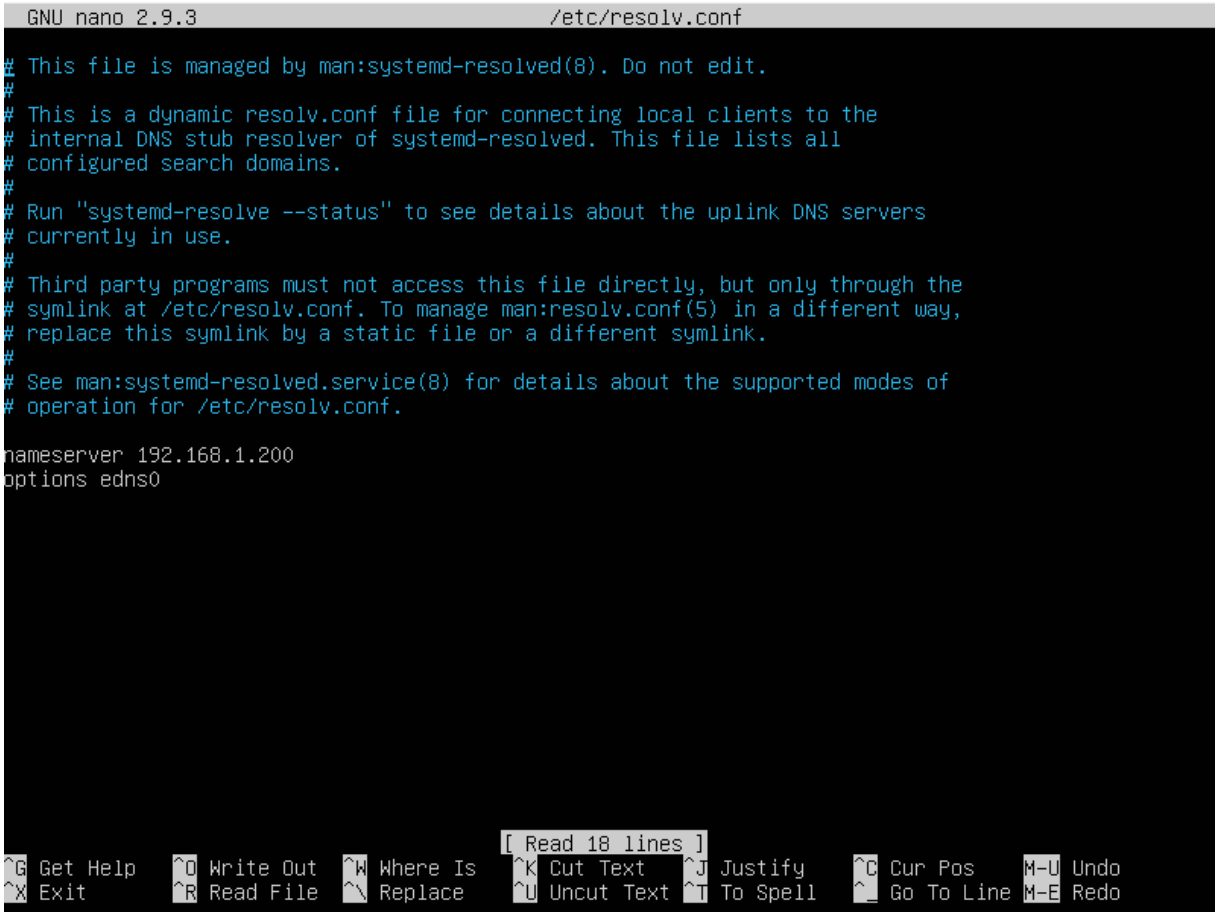
Możemy sprawdzić czy usługa (deamon) działa poprawnie używając polecenia:

service bind9 status



Ustawmy teraz żeby nasz Linux korzystał z właśnie przez nas skonfigurowanego serwera DNS. W tym celu musimy edytować plik **/etc/resolv.conf** i wpisać w nim adres naszego serwera DNS:

nano /etc/resolv.conf



Do testów użyjemy narzędzia **nslookup** – jest ono zarówno w Linuxie jak i w Windowsie. Jako że ustawiliśmy, żeby naszym DNS był nasz lokalny serwer DNS, to wystarczy wpisać poleceni nslookup a po uruchomieniu programu wpisać nazwę domenową, o która chcemy odpytać serwer DNS:

Widzimy że serwerem który odpytujemy jest nasz komputer i zna naszą domenę zsme.local

Zapytajmy się o nasze rekordy, które konfigurowaliśmy:

```
root@jkubuntu:/etc/bind# nslookup
> ns1.zsme.local
Server:         192.168.1.200
Address:        192.168.1.200#53

Name:   ns1.zsme.local
Address: 192.168.1.200
> xp.zsme.local
Server:         192.168.1.200
Address:        192.168.1.200#53

Name:   xp.zsme.local
Address: 192.168.1.150
> komp1.zsme.local
Server:         192.168.1.200
Address:        192.168.1.200#53

komp1.zsme.local        canonical name = xp.zsme.local.
Name:   xp.zsme.local
Address: 192.168.1.150
> router.zsme.local
Server:         192.168.1.200
Address:        192.168.1.200#53

Name:   router.zsme.local
Address: 192.168.1.1
>
```

Widzimy, że wszystkie rekordy które ustawiliśmy w strefie wyszukiwania do przodu działają poprawnie.
Spróbujmy teraz wykonać zapytania wstecz, czyli wpisujemy adres IP a serwer powinien nam odpowiedzieć nazwą domenową:

```
zsme@jkubuntu:~$ nslookup
> 192.168.1.200
200.1.168.192.in-addr.arpa      name = ns1.zsme.local.
> 192.168.1.1
1.1.168.192.in-addr.arpa        name = router.zsme.local.
> 192.168.1.150
150.1.168.192.in-addr.arpa      name = xp.zsme.local.
>
```

Serwer tez poprawnie odpowiada na zapytania.

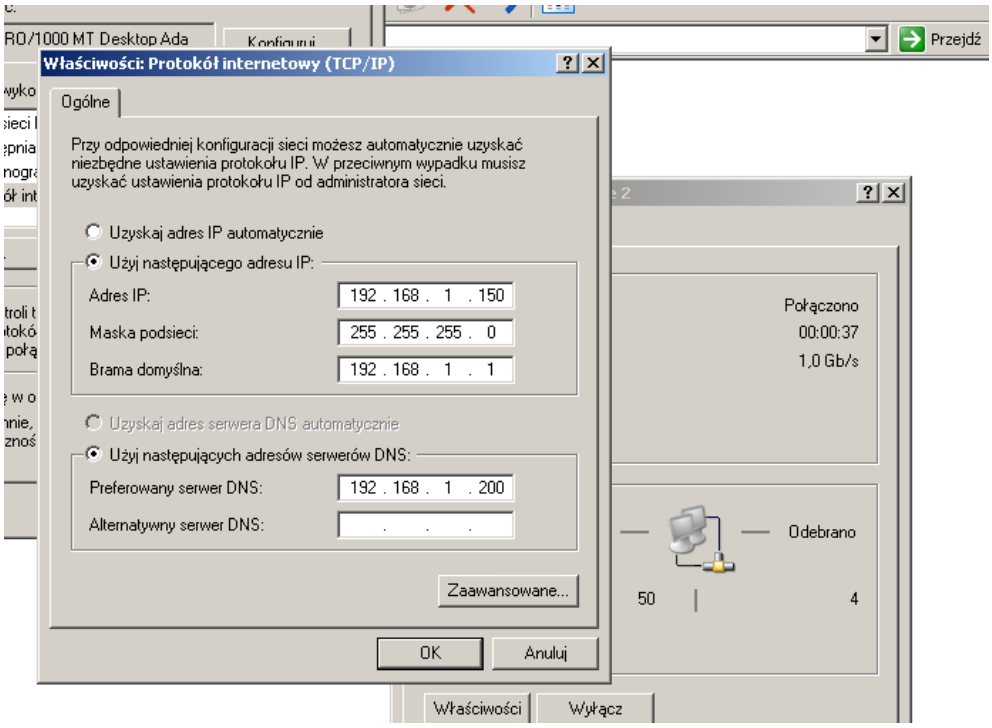
przetestujmy, czy serwet będzie potrafił przekazać zapytanie do forwardera (zdefiniowanego przez nas serwera DNS: 8.8.8.8)

```
zsme@jkubuntu:~$ nslookup
> interia.pl
Server:         192.168.1.200
Address:        192.168.1.200#53

Non-authoritative answer:
Name:   interia.pl
Address: 217.74.65.23
> _
```

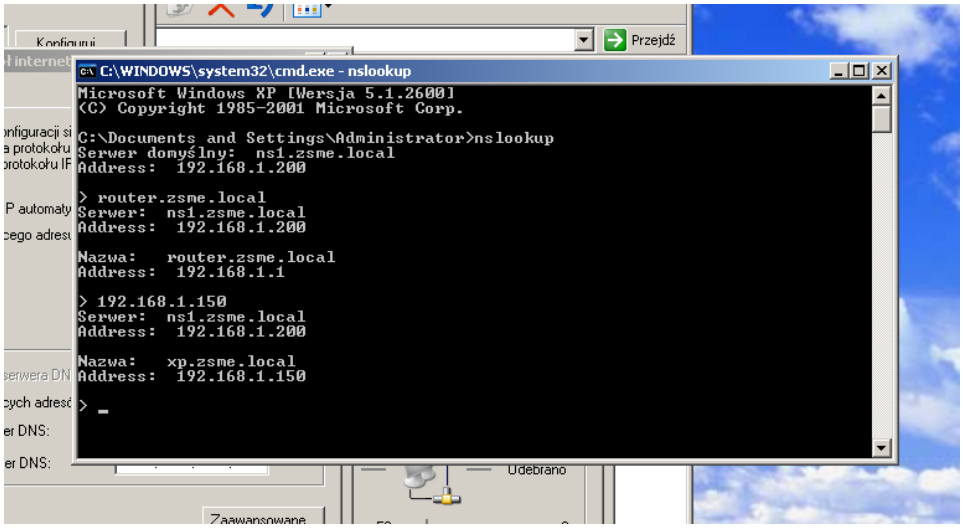
Jak widać, serwer odpytany o domenę, której przecież nie ma skonfigurowanej, przekazuje zapytanie do forwardera a ten je rozwiązuje i zwraca odpowiedź.

Odpalmy teraz maszynę testową z XP, skonfigurujmy ja do pracy w tej samej sieci z pasująca adresacją:

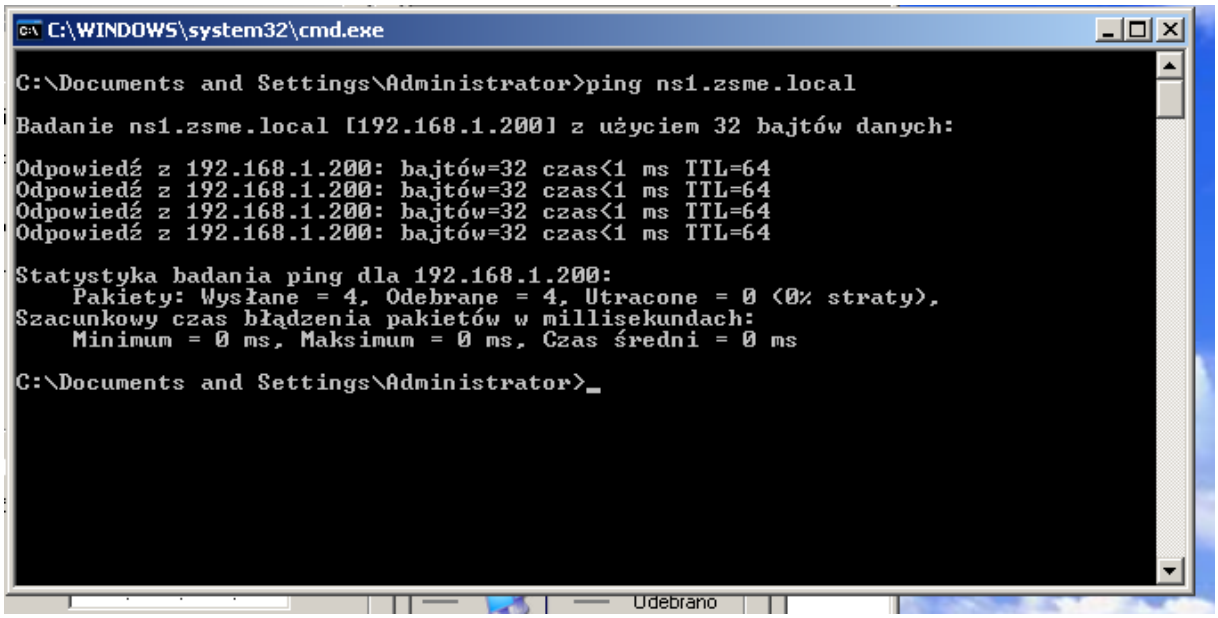
karta sieciowa: zmostkowana
IP:192.168.1.150/24
B:192.168.1.1
DNS:192.168.1.200 (adres naszego Ubuntu Server z uruchomionym serwerem DNS)



Użyjmy tez narzędzia nslookup i odpytajmy serwer o parę rekordów:



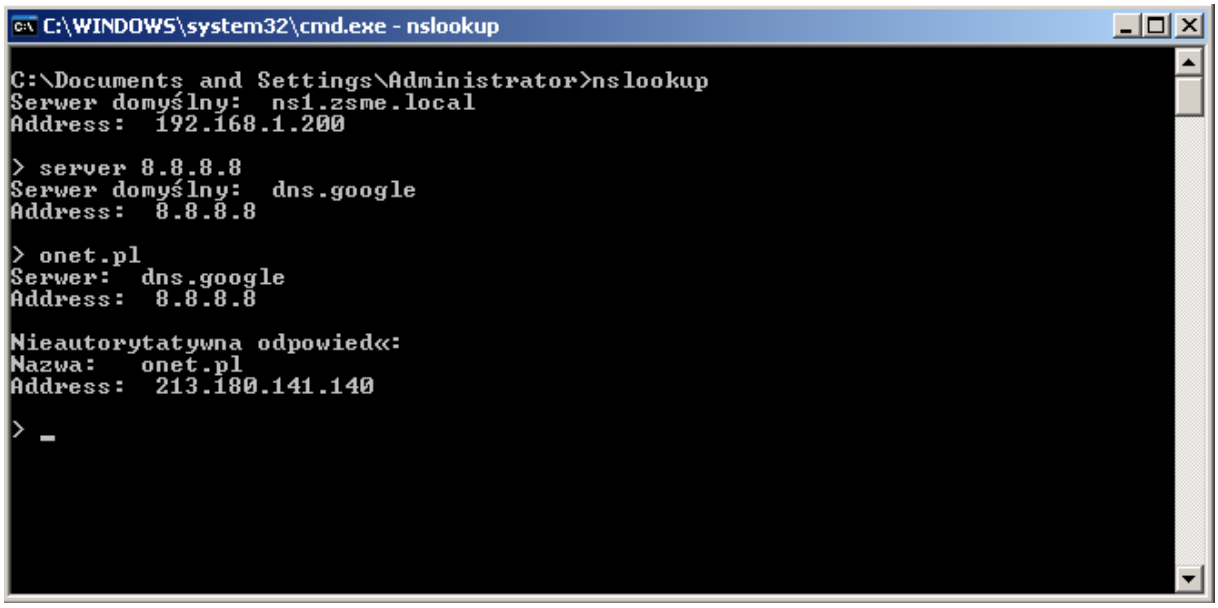
Jak widać, działa poprawnie. Oczywiście możemy też po prostu puścić ping na nazwę domenową



11. NSLOOKUP – dodatek

W momencie, gdy chcemy żeby nslookup nie odpytywał z domyślnego serwera DNS (tego który jest skonfigurowany w systemie) to pod systemem Windows, po uruchomieniu nslookup musimy wpisać polecenie:

server 8.8.8.8 (serwer będzie odpytywał DNS googla zamiast naszego)



W Linuxie musimy uruchomić nslookup z podaniem adresu serwera który ma odpytywać, zamiast domyślnego:

nslookup – 8.8.8.8

Czyszczenie pamięci podręcznej serwera DNS:

rndc flush

Czyszczenie pamięci podręcznej DNS w Windows:

ipconfig /flushdns