

POLITECHNIKA WROCLAWSKA

SCR - SIECI KOMPUTEROWE
CZWARTEK TN 18.55 - 20.35

Projekt routera domowego

Autor:

Jakub KOZŁOWICZ

Jakub BACIK

Michał MENDELAK

Prowadzący:

dr inż. Jerzy GREBLICKI

11 czerwca 2021



Politechnika
Wroclawska

Spis treści

1	Wybór IP	2
2	Serwer DHCP	3
3	ROUTING	6
4	NAT	7
5	Firewall	8
6	Serwer http	10
6.1	Baza danych	10
6.2	PHP	11
6.3	Wordpress	11
6.4	Nginx	13
6.5	Rezultat konfiguracji http	14
7	Serwer ssh	15
8	Serwer plików Samba	16
9	Serwer DNS	19

1 Wybór IP

Dla sieci wirtualnej wybrano adres IP z zakresu adresów prywatnych. Maskę podsieci wybrano najpopularniejszą - 24 bitową.

IP Address:	192.168.9.1	IP Address:	192.168.10.1
Network Address:	192.168.9.0	Network Address:	192.168.10.0
Usable Host IP Range:	192.168.9.1 - 192.168.9.254	Usable Host IP Range:	192.168.10.1 - 192.168.10.254
Broadcast Address:	192.168.9.255	Broadcast Address:	192.168.10.255
Total Number of Hosts:	256	Total Number of Hosts:	256
Number of Usable Hosts:	254	Number of Usable Hosts:	254
Subnet Mask:	255.255.255.0	Subnet Mask:	255.255.255.0
Wildcard Mask:	0.0.0.255	Wildcard Mask:	0.0.0.255
Binary Subnet Mask:	11111111.11111111.11111111.00000000	Binary Subnet Mask:	11111111.11111111.11111111.00000000
IP Class:	C	IP Class:	C
CIDR Notation:	/24	CIDR Notation:	/24
IP Type:	Private	IP Type:	Private
Short:	192.168.9.1 /24	Short:	192.168.10.1 /24
Binary ID:	11000000101010000000100100000001	Binary ID:	11000000101010000000101000000001
Integer ID:	3232237825	Integer ID:	3232238081
Hex ID:	0xc0a80901	Hex ID:	0xc0a80a01
in-addr.arpa:	1.9.168.192.in-addr.arpa	in-addr.arpa:	1.10.168.192.in-addr.arpa
IPv4 Mapped Address:	::ffff:c0a8.0901	IPv4 Mapped Address:	::ffff:c0a8.0a01
6to4 Prefix:	2002:c0a8.0901::/48	6to4 Prefix:	2002:c0a8.0a01::/48

Rysunek 1.1: Obliczenie zakresu adresów IP.

Po obliczeniu zakresu adresów oraz adresu broadcast na routerze został manualnie ustawiony statyczny adres IP na interfejsie dla sieci wirtualnej.

```
jakub@ubuntu-vm:~$ sudo ifconfig enp0s8 inet 192.168.9.1 netmask 255.255.255.0 up
jakub@ubuntu-vm:~$ sudo ifconfig enp0s9 inet 192.168.10.1 netmask 255.255.255.0 up
jakub@ubuntu-vm:~$ ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.8.101 netmask 255.255.255.0 broadcast 192.168.8.255
    inet6 fe80::a00:27ff:fe53:77b6 prefixlen 64 scopeid 0x20<link>
    inet6 fdc0:f4e6:d4e7:2400:a00:27ff:fe53:77b6 prefixlen 64 scopeid 0x0<global>
    ether 08:00:27:53:77:b6 txqueuelen 1000 (Ethernet)
    RX packets 252 bytes 38806 (38.8 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 423 bytes 51064 (51.0 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

enp0s8: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.9.1 netmask 255.255.255.0 broadcast 192.168.9.255
    inet6 fe80::a00:27ff:fed9:5d14 prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:d9:5d:14 txqueuelen 1000 (Ethernet)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 180 bytes 26725 (26.7 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

enp0s9: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.10.1 netmask 255.255.255.0 broadcast 192.168.10.255
    inet6 fe80::9a8d:2c3e:39f3:bd85 prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:63:ec:2b txqueuelen 1000 (Ethernet)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 185 bytes 34745 (34.7 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 1176 bytes 113251 (113.2 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 1176 bytes 113251 (113.2 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

Rysunek 1.2: Skonfigurowanie adresu IP na routerze.

2 Serwer DHCP

Skonfigurowanie serwera DHCP zaczęto od zainstalowania odpowiedniego pakietu oraz edycji pliku `/etc/default/isc-dhcp-server`. W nim ustawiono interfejs sieci wirtualnej oraz ścieżkę do pliku konfiguracyjnego.

```
jakub@ubuntu-vm:~$ sudo apt install isc-dhcp-server
Reading package lists... Done
Building dependency tree
Reading state information... Done
isc-dhcp-server is already the newest version (4.4.1-2.1ubuntu5.20.04.1).
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
jakub@ubuntu-vm:~$ sudo vim /etc/default/isc-dhcp-server
jakub@ubuntu-vm:~$ cat /etc/default/isc-dhcp-server
# Defaults for isc-dhcp-server (sourced by /etc/init.d/isc-dhcp-server)

# Path to dhcpd's config file (default: /etc/dhcp/dhcpd.conf).
DHCPDv4_CONF=/etc/dhcp/dhcpd.conf
#DHCPDv6_CONF=/etc/dhcp/dhcpd6.conf

# Path to dhcpd's PID file (default: /var/run/dhcpd.pid).
#DHCPDv4_PID=/var/run/dhcpd.pid
#DHCPDv6_PID=/var/run/dhcpd6.pid

# Additional options to start dhcpd with.
# Don't use options -cf or -pf here; use DHCPD_CONF/ DHCPD_PID instead
#OPTIONS=""

# On what interfaces should the DHCP server (dhcpd) serve DHCP requests?
# Separate multiple interfaces with spaces, e.g. "eth0 eth1".
INTERFACESv4="enp0s8 enp0s9"
INTERFACESv6=""
jakub@ubuntu-vm:~$
```

Rysunek 2.1: Instalacja serwera DHCP na routerze.

Następnie edytując plik `/etc/dhcp/dhcpd.conf` skonfigurowano podsieć, zakresy dynamicznej puli IP, oraz serwery DNS.

```
jakub@ubuntu-vm:~$ sudo vim /etc/dhcp/dhcpd.conf
jakub@ubuntu-vm:~$ cat /etc/dhcp/dhcpd.conf
# dhcpd.conf
#

ddns-update-style none;
authoritative;

subnet 192.168.9.0 netmask 255.255.255.0 {
    range 192.168.9.100 192.168.9.200;
    option domain-name-servers 192.168.9.1;
    option domain-name "ubuntu.local";
    option subnet-mask 255.255.255.0;
    option routers 192.168.9.1;
    option broadcast-address 192.168.9.255;
    default-lease-time 600;
    max-lease-time 7200;
}

subnet 192.168.10.0 netmask 255.255.255.0 {
    range 192.168.10.100 192.168.10.200;
    option domain-name-servers 192.168.10.1;
    option domain-name "ubuntu.local";
    option subnet-mask 255.255.255.0;
    option routers 192.168.10.1;
    option broadcast-address 192.168.10.255;
    default-lease-time 600;
    max-lease-time 7200;
}

jakub@ubuntu-vm:~$ sudo systemctl restart isc-dhcp-server
jakub@ubuntu-vm:~$ |
```

Rysunek 2.2: Konfiguracja serwera DHCP na routerze.

3 ROUTING

Do skonfigurowania routingu wykorzystano netplan dostępny w ubuntu od wersji 18.04. Przypisano w nim statyczne IP dla obu interfejsów oraz dla interfejsu zewnętrznego przypisano adresy serwerów DNS jak i domyślną bramę.

```
jakub@ubuntu-vm:~$ sudo vim /etc/netplan/00-installer-config.yaml
jakub@ubuntu-vm:~$ cat /etc/netplan/00-installer-config.yaml
network:
  version: 2
  renderer: networkd
  ethernets:
    enp0s3:
      dhcp4: false
      addresses:
        - 192.168.8.101/24
      gateway4: 192.168.8.1
      nameservers:
        addresses:
          - 192.168.8.1
          - 8.8.8.8
    enp0s8:
      dhcp4: false
      addresses:
        - 192.168.9.1/24
    enp0s9:
      dhcp4: false
      addresses:
        - 192.168.10.1/24
jakub@ubuntu-vm:~$ sudo netplan apply
jakub@ubuntu-vm:~$ |
```

Rysunek 3.1: Konfiguracja statycznych IP na routerze.

W następnym kroku edytując plik `/etc/sysctl.conf` oraz usuwając komentarz dla linii `net.ipv4.ip_forward=1` włączono przekierowywanie adresów IP. Na koniec zastosowano zmiany wprowadzone w tym pliku komendą `sudo sysctl -p`.

```
jakub@ubuntu-vm:~$ sudo vim /etc/sysctl.conf |
```

Rysunek 3.2: Włączenie przekierowywania IP - 1.

```
# Uncomment the next line to enable packet forwarding for IPv4
net.ipv4.ip_forward=1
```

Rysunek 3.3: Włączenie przekierowywania IP - 2.

```
jakub@ubuntu-vm:~$ sudo sysctl -p
net.ipv4.ip_forward = 1
jakub@ubuntu-vm:~$ |
```

Rysunek 3.4: Włączenie przekierowywania IP - 3.

4 NAT

Konfigurując NAT umożliwiono przekazywanie pakietów między odpowiednimi interfejsami sieciowymi oraz włączono maskaradę wewnętrznych adresów IP, aby nie były one widoczne na zewnątrz.

```
jakub@ubuntu-vm:~$ sudo iptables -A FORWARD -i enp0s8 -o enp0s3 -j ACCEPT
jakub@ubuntu-vm:~$ sudo iptables -A FORWARD -i enp0s9 -o enp0s3 -j ACCEPT
jakub@ubuntu-vm:~$ sudo iptables -A FORWARD -i enp0s3 -o enp0s8 -m state --state RELATED,ESTABLISHED -
j ACCEPT
jakub@ubuntu-vm:~$ sudo iptables -A FORWARD -i enp0s3 -o enp0s9 -m state --state RELATED,ESTABLISHED -
j ACCEPT
jakub@ubuntu-vm:~$ sudo iptables -t nat -A POSTROUTING -o enp0s3 -j MASQUERADE
jakub@ubuntu-vm:~$ sudo su
root@ubuntu-vm:/home/jakub# sudo iptables-save > /etc/iptables/rules.v4
root@ubuntu-vm:/home/jakub# exit
exit
jakub@ubuntu-vm:~$ route
Kernel IP routing table
Destination Gateway Genmask Flags Metric Ref Use Iface
default _gateway 0.0.0.0 UG 100 0 0 enp0s3
link-local 0.0.0.0 255.255.0.0 U 1000 0 0 enp0s3
192.168.8.0 0.0.0.0 255.255.255.0 U 100 0 0 enp0s3
192.168.9.0 0.0.0.0 255.255.255.0 U 101 0 0 enp0s8
192.168.10.0 0.0.0.0 255.255.255.0 U 102 0 0 enp0s9
jakub@ubuntu-vm:~$ |
```

Rysunek 4.1: Konfiguracja przekierowań tabel IP.

Następnie sprawdzono czy komputer klienta ma dostęp do internetu.


```
jakub@ubuntu-vm:~$ route
Kernel IP routing table
Destination      Gateway          Genmask          Flags Metric Ref    Use Iface
default          _gateway        0.0.0.0          UG    100    0      0 enp0s3
link-local       0.0.0.0         255.255.0.0     U     1000   0      0 enp0s3
192.168.9.0     0.0.0.0         255.255.255.0   U     100    0      0 enp0s3
jakub@ubuntu-vm:~$ ping 8.8.8.8 -c 5
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=118 time=433 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=118 time=144 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=118 time=153 ms
64 bytes from 8.8.8.8: icmp_seq=4 ttl=118 time=127 ms
64 bytes from 8.8.8.8: icmp_seq=5 ttl=118 time=33.2 ms

--- 8.8.8.8 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4037ms
rtt min/avg/max/mdev = 33.208/178.069/432.632/134.264 ms
jakub@ubuntu-vm:~$ ping google.com -c 5
PING google.com (142.250.75.14) 56(84) bytes of data.
64 bytes from waw07s03-in-f14.1e100.net (142.250.75.14): icmp_seq=1 ttl=117 time=595 ms
64 bytes from waw07s03-in-f14.1e100.net (142.250.75.14): icmp_seq=2 ttl=117 time=471 ms
64 bytes from waw07s03-in-f14.1e100.net (142.250.75.14): icmp_seq=3 ttl=117 time=532 ms
64 bytes from waw07s03-in-f14.1e100.net (142.250.75.14): icmp_seq=4 ttl=117 time=493 ms
64 bytes from waw07s03-in-f14.1e100.net (142.250.75.14): icmp_seq=5 ttl=117 time=415 ms

--- google.com ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 5404ms
rtt min/avg/max/mdev = 414.824/501.310/594.962/60.282 ms
jakub@ubuntu-vm:~$ |
```

Rysunek 4.2: Sprawdzenie poprawności konfiguracji sieci routera.

5 Firewall

Do konfiguracji firewall'a użyto narzędzia ufw. Pozwolono w nim na komunikację usług takich jak

1. OpenSSH - serwer ssh,
2. Nginx HTTP - serwer stron internetowych,
3. Samba - serwer plików,
4. Bind9 - serwer DNS.

```
jakub@ubuntu-vm:~$ sudo ufw app list
Available applications:
  Bind9
  CUPS
  Nginx Full
  Nginx HTTP
  Nginx HTTPS
  OpenSSH
  Samba
jakub@ubuntu-vm:~$ sudo ufw allow OpenSSH
sudo: ufw: command not found
jakub@ubuntu-vm:~$ sudo ufw allow OpenSSH
Rule added
Rule added (v6)
jakub@ubuntu-vm:~$ sudo ufw allow Samba
Rule added
Rule added (v6)
jakub@ubuntu-vm:~$ sudo ufw allow Nginx HTTP
ERROR: Need 'to' or 'from' clause
jakub@ubuntu-vm:~$ sudo ufw allow Nginx
ERROR: Could not find a profile matching 'Nginx'
jakub@ubuntu-vm:~$ sudo ufw allow 'Nginx HTTP'
Rule added
Rule added (v6)
jakub@ubuntu-vm:~$ sudo ufw allow Bind9
Rule added
Rule added (v6)
jakub@ubuntu-vm:~$ sudo ufw status
Status: active

To Action From
--
OpenSSH ALLOW Anywhere
Samba ALLOW Anywhere
Nginx HTTP ALLOW Anywhere
Bind9 ALLOW Anywhere
OpenSSH (v6) ALLOW Anywhere (v6)
Samba (v6) ALLOW Anywhere (v6)
Nginx HTTP (v6) ALLOW Anywhere (v6)
Bind9 (v6) ALLOW Anywhere (v6)

jakub@ubuntu-vm:~$ sudo ufw enable
Firewall is active and enabled on system startup
jakub@ubuntu-vm:~$ |
```

Rysunek 5.1: Konfiguracja firewall na routerze.

6 Serwer http

Stronę na serwerze wybrano jako stronę wordpress. Do tego będzie wymagana baza danych jak i również php.

6.1 Baza danych

Zainstalowano odpowiedni serwer bazy danych.

```
jakub@ubuntu-vm:~$ sudo apt install mysql
mysql-client          mysql-client-core-8.0  mysql-router          mysql-server
mysql-client-8.0     mysql-common           mysql-sandbox         mysql-server-8.0
jakub@ubuntu-vm:~$ sudo apt install mysql-server
```

Rysunek 6.1: Instalacja bazy danych na routerze.

W kolejnym kroku utworzono bazę danych pod stronę wordpress oraz użytkownika z uprawnieniami do edytowania tej bazy danych.

```
jakub@ubuntu-vm:~$ sudo mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 9
Server version: 8.0.25-0ubuntu0.20.04.1 (Ubuntu)

Copyright (c) 2000, 2021, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> CREATE DATABASE wordpress;
Query OK, 1 row affected (0,03 sec)

mysql> CREATE USER 'admin'@'localhost' IDENTIFIED BY PASSWORD 'wordpress';
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'PASSWORD 'wordpress''
at line 1
mysql> CREATE USER 'admin'@'localhost' IDENTIFIED BY 'wordpress';
Query OK, 0 rows affected (0,03 sec)

mysql> GRANT ALL PRIVILEGES ON wordpress.* TO 'admin'@'localhost';
Query OK, 0 rows affected (0,01 sec)

mysql> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0,01 sec)

mysql> EXIT;
Bye
jakub@ubuntu-vm:~$
```

Rysunek 6.2: Utworzenie bazy danych pod stronę wordpress.

6.2 PHP

```
jakub@ubuntu-vm:~$ sudo apt install php-fpm php-mysql
[sudo] password for jakub:
Reading package lists... Done
Building dependency tree
```

Rysunek 6.3: Instalacja php wymaganego przez wordpress.

6.3 Wordpress

Do skonfigurowania wordpressa ściągnięto jego najnowszą wersję. Wypakowano go do folderu /tmp aby po wylogowaniu niepotrzebne pliki zostały samoczynnie usunięte.

```
jakub@ubuntu-vm:~$ cd tmp
bash: cd: tmp: No such file or directory
jakub@ubuntu-vm:~$ cd /tmp
jakub@ubuntu-vm:/tmp$ wget -c http://wordpress.org/latest.tar.gz
--2021-05-17 19:57:13-- http://wordpress.org/latest.tar.gz
Resolving wordpress.org (wordpress.org)... 198.143.164.252
Connecting to wordpress.org (wordpress.org)|198.143.164.252|:80... connected.
HTTP request sent, awaiting response... 301 Moved Permanently
Location: https://wordpress.org/latest.tar.gz [following]
--2021-05-17 19:57:13-- https://wordpress.org/latest.tar.gz
Connecting to wordpress.org (wordpress.org)|198.143.164.252|:443... connected.
HTTP request sent, awaiting response... 416 Requested Range Not Satisfiable

The file is already fully retrieved; nothing to do.
jakub@ubuntu-vm:/tmp$ tar -xzf latest.tar.gz
```

Rysunek 6.4: Pobranie plików wordpress.

W kolejnym kroku przekopiowano pliki do odpowiedniego katalogu oraz zmieniono uprawnienia aby później można było poprzez przeglądarkę wysyłać pliki.

```
jakub@ubuntu-vm:/tmp$ sudo cp -R wordpress/ /var/www/html/router
[sudo] password for jakub:
jakub@ubuntu-vm:/tmp$ sudo chown -R www-data:www-data /var/www/html/router/
jakub@ubuntu-vm:/tmp$ sudo chmod -R 775 /var/www/html/router/
jakub@ubuntu-vm:/tmp$ |
```

Rysunek 6.5: Przeniesienie plików wordpress w odpowiednie miejsce oraz zmiana ich uprawnień.

Ostatnią rzeczą było podłączenie bazy danych do wordpressa. W tym celu zmieniono nazwę pliku `wp-config-sample.php` na `wp-config.php`.

```
jakub@ubuntu-vm:~$ sudo mv /var/www/html/router/wp-config-sample.php /var/www/html/router/wp-config.php
jakub@ubuntu-vm:~$ sudo vim /var/www/html/router/wp-config.php
```

Rysunek 6.6: Konfiguracja wordpress - 1.

Następnie edytowano ten plik wpisując odpowiednie informacje o wcześniej utworzonej bazie danych.

```
// ** MySQL settings - You can get this info from your web host ** //
/** The name of the database for WordPress */
define( 'DB_NAME', 'wordpress' );

/** MySQL database username */
define( 'DB_USER', 'admin' );

/** MySQL database password */
define( 'DB_PASSWORD', 'wordpress' );

/** MySQL hostname */
define( 'DB_HOST', 'localhost' );

/** Database Charset to use in creating database tables. */
define( 'DB_CHARSET', 'utf8' );

/** The Database Collate type. Don't change this if in doubt. */
define( 'DB_COLLATE', '' );

/**#@+
 * Authentication Unique Keys and Salts.
 */
```

Rysunek 6.7: Konfiguracja wordpress - 2.

Na koniec w tym pliku dodano klucze autoryzacyjne i zapisano plik.

```
*/
define( 'AUTH_KEY',         '=) _C#.k8=(Zl)DN+2=sp*(c)}$s-g|C,30v{Hjg@4*ZJZp`D^0=nb~4ykVAjYl<');
define( 'SECURE_AUTH_KEY', 'G@/,|~gtR5u(;X&V@-U)+0w8u`<+;+Q&Ja1 #T!tm+:L-uvzQ[-nvsCo*2{4Re0h');
define( 'LOGGED_IN_KEY',   'Ow^X*+-60!^sTsy4+[V<q|d,|K?XZh7@_{d$^1_{5|}$br`YQA7R9yA5|>^<@<z|');
define( 'NONCE_KEY',       '6((#mM5%QZ+e=MX|?)1/7CsMA,N2zg1*6Cc<q _-CJY1&0a(Q:$EmT,3e|*2{?oz');
define( 'AUTH_SALT',       'ks+cT0Mpsu{n[+SK&[-Y0>Fz9r$ç;1BR|lfnh/OaT_<a{5ALNB~)-=qFB,f~q>');
define( 'SECURE_AUTH_SALT', '/-h{cz2;+DOM{c7JzgSCMX`=IJSpg?-5;9^kFI4hD(_n+g(o4/OU@TW?h;>JgHkw');
define( 'LOGGED_IN_SALT',  'bgb-ec>5pq=ENC|V%gUG2hs C<Su1R|-7),zz:l9>F GRC. A[TjG(jR8Q@(sf-');
define( 'NONCE_SALT',      'n0/1nTD#^]:v3W/X0=C+`pYA9zT!N6pf&Wm3 S?m%RD,U;18%Jqum0cF:uKh]79s');
/**#@-*/
```

Rysunek 6.8: Konfiguracja wordpress - 3.

6.4 Nginx

Na początku zainstalowano serwer stron internetowych - Nginx.

```
jakub@ubuntu-vm:~$ sudo apt install nginx
```

Rysunek 6.9: Instalacja serwera internetowego.

Aby strona wordpress była dostępna z zewnątrz skonfigurowano serwer stron internetowych Nginx. W katalogu `/etc/nginx/sites-available/` utworzono plik `router` oraz wpisano w niego odpowiednią konfigurację. Utworzono również symboliczny link do katalogu `/etc/nginx/sites-enabled/`.

```
jakub@ubuntu-vm:~$ cat /etc/nginx/sites-available/router
server {
    listen 80;
    server_name router;
    root /var/www/html/router;

    index index.html index.htm index.php;

    location / {
        try_files $uri $uri/ =404;
    }

    location ~ /\.php$ {
        include snippets/fastcgi-php.conf;
        fastcgi_pass unix:/var/run/php/php7.4-fpm.sock;
    }

    location ~ /\.ht {
        deny all;
    }
}
jakub@ubuntu-vm:~$ sudo ln -s /etc/nginx/sites-available/router /etc/nginx/sites-enabled/
```

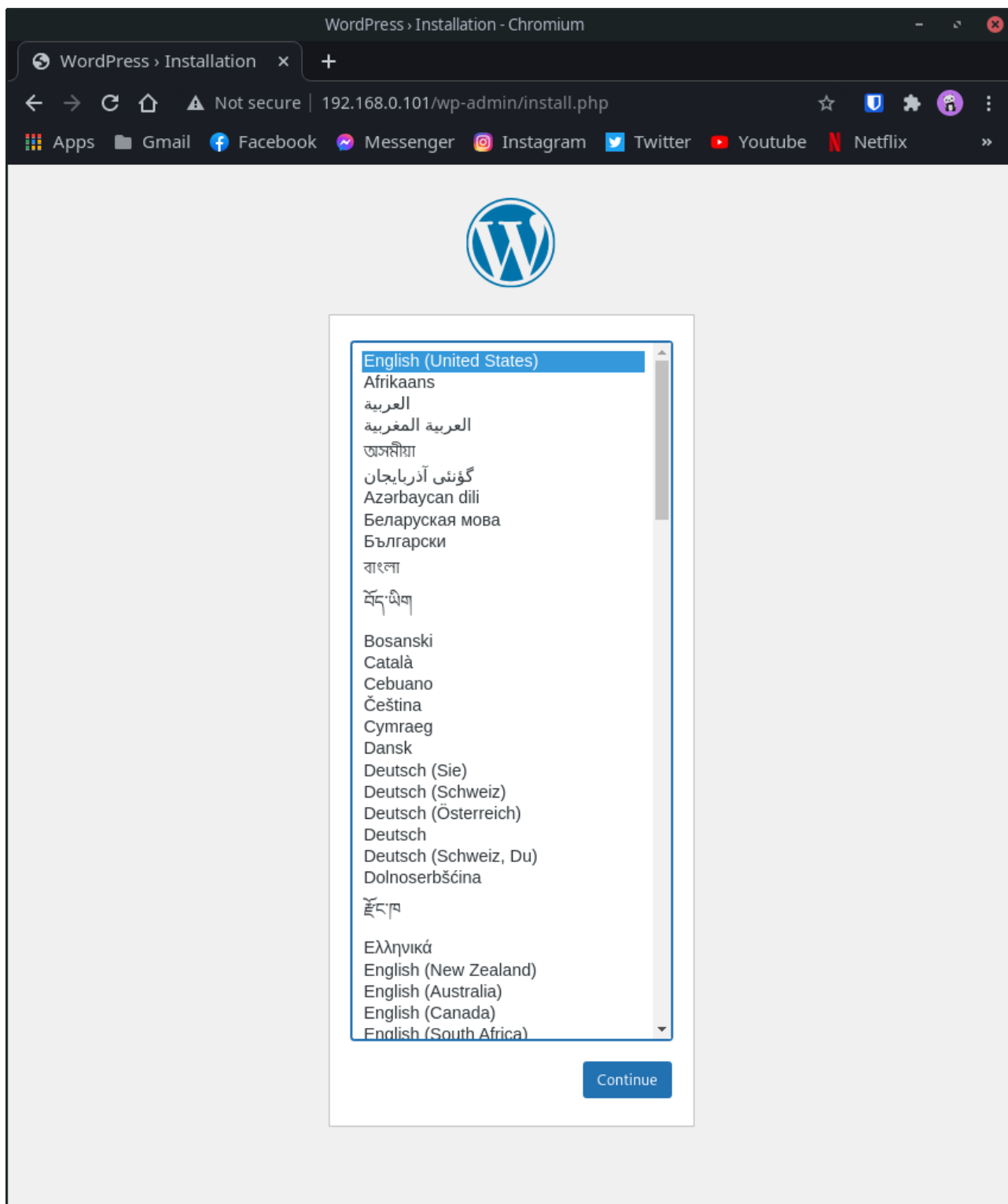
Rysunek 6.10: Konfiguracja nginx do wyświetlania strony wordpress.

Na koniec wystarczyło zrestartować nginx aby strona zaczęła być wyświetlana po wpisaniu adresu IP routera.

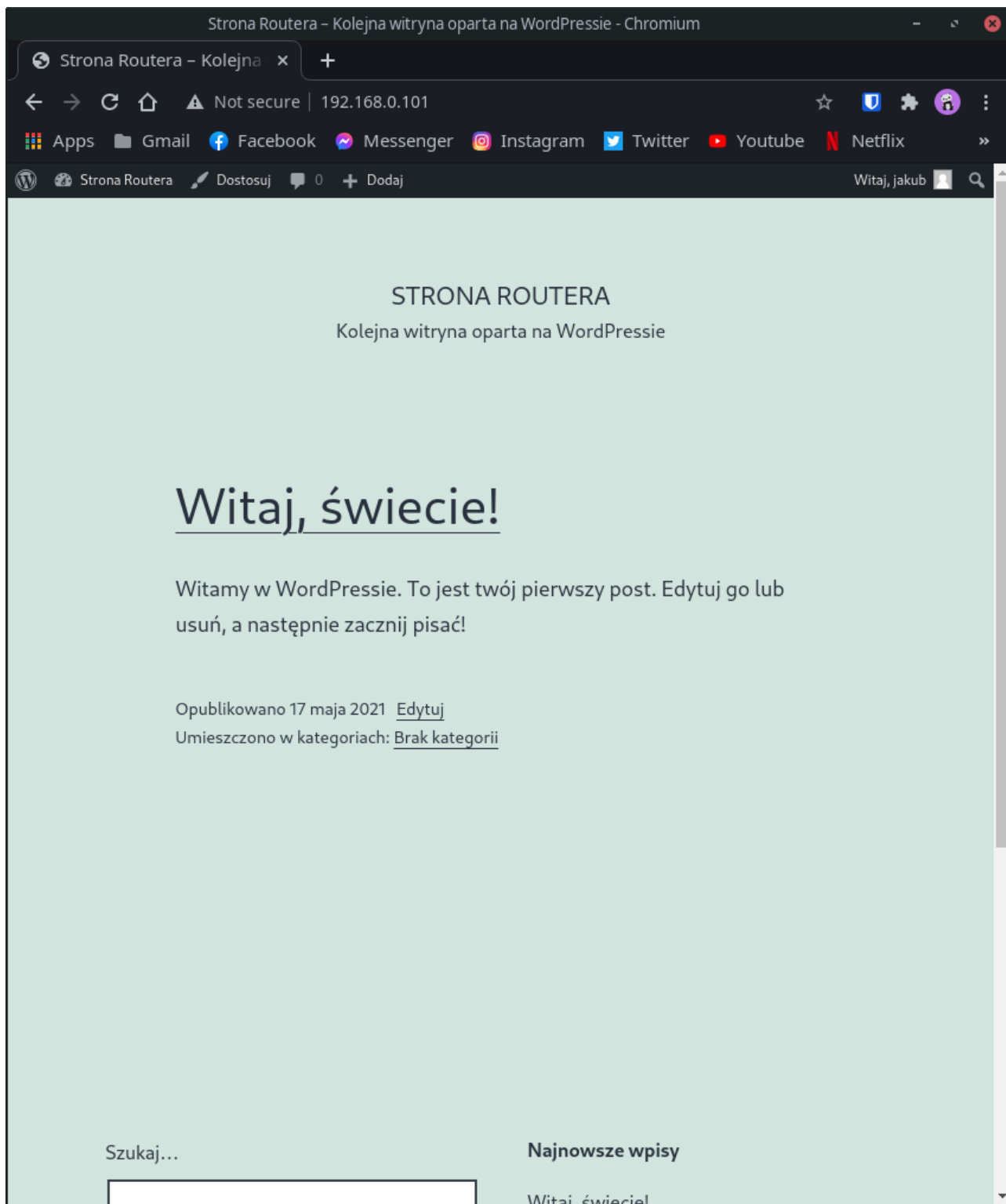
```
jakub@ubuntu-vm:~$ sudo systemctl restart nginx.service
jakub@ubuntu-vm:~$ |
```

Rysunek 6.11: Zrestartowanie serwisu nginx do wprowadzenia zmian.

6.5 Rezultat konfiguracji http



Rysunek 6.12: Ekran początkowej konfiguracji wordpress.



Rysunek 6.13: Strona wordpress po konfiguracji wyświetlana z routera.

7 Serwer ssh

Na początek zainstalowany został serwer SSH.


```
jakub@ubuntu-vm:~$ sudo apt install openssh-server
```

Rysunek 7.1: Instalacja serwera ssh na routerze.

Serwer SSH nie wymaga dodatkowej konfiguracji aby zaczął działać. Wykonana jednak została dodatkowa konfiguracja aby zwiększyć trochę bezpieczeństwo.

```
jakub@ubuntu-vm:~$ cat /etc/ssh/sshd_config
#      $OpenBSD: sshd_config,v 1.103 2018/04/09 20:41:22 tj Exp $

# This is the sshd server system-wide configuration file.  See
# sshd_config(5) for more information.

# This sshd was compiled with PATH=/usr/bin:/bin:/usr/sbin:/sbin

# The strategy used for options in the default sshd_config shipped with
# OpenSSH is to specify options with their default value where
# possible, but leave them commented.  Uncommented options override the
# default value.

Include /etc/ssh/sshd_config.d/*.conf

Port 22
LogLevel VERBOSE
PermitRootLogin no
ChallengeResponseAuthentication no
UsePAM yes
AllowTcpForwarding no
X11Forwarding no
PrintMotd no
AcceptEnv LANG LC_*
Subsystem      sftp      /usr/lib/openssh/sftp-server
```

Rysunek 7.2: Konfiguracja serwera ssh na routerze.

8 Serwer plików Samba

Na początku zainstalowano serwer plików samba oraz utworzono odpowiednie foldery. Tym folderom zmieniono uprawnienia dostępu aby każdy w wirtualnej sieci miał do nich dostęp.

```
jakub@ubuntu-vm:~$ sudo apt-get install samba
Reading package lists... Done
Building dependency tree
Reading state information... Done
samba is already the newest version (2:4.11.6+dfsg-0ubuntu1.8).
The following package was automatically installed and is no longer required:
  distro-info
Use 'sudo apt autoremove' to remove it.
0 upgraded, 0 newly installed, 0 to remove and 1 not upgraded.
jakub@ubuntu-vm:~$ sudo mkdir -p /data/public
jakub@ubuntu-vm:~$ sudo chmod -R 777 /data/public
jakub@ubuntu-vm:~$ sudo chown -R nobody.nogroup /data/public
jakub@ubuntu-vm:~$ sudo chmod -R 777 /data
jakub@ubuntu-vm:~$ |
```

Rysunek 8.1: Instalacja serwera plików oraz utworzenie odpowiednich folderów.

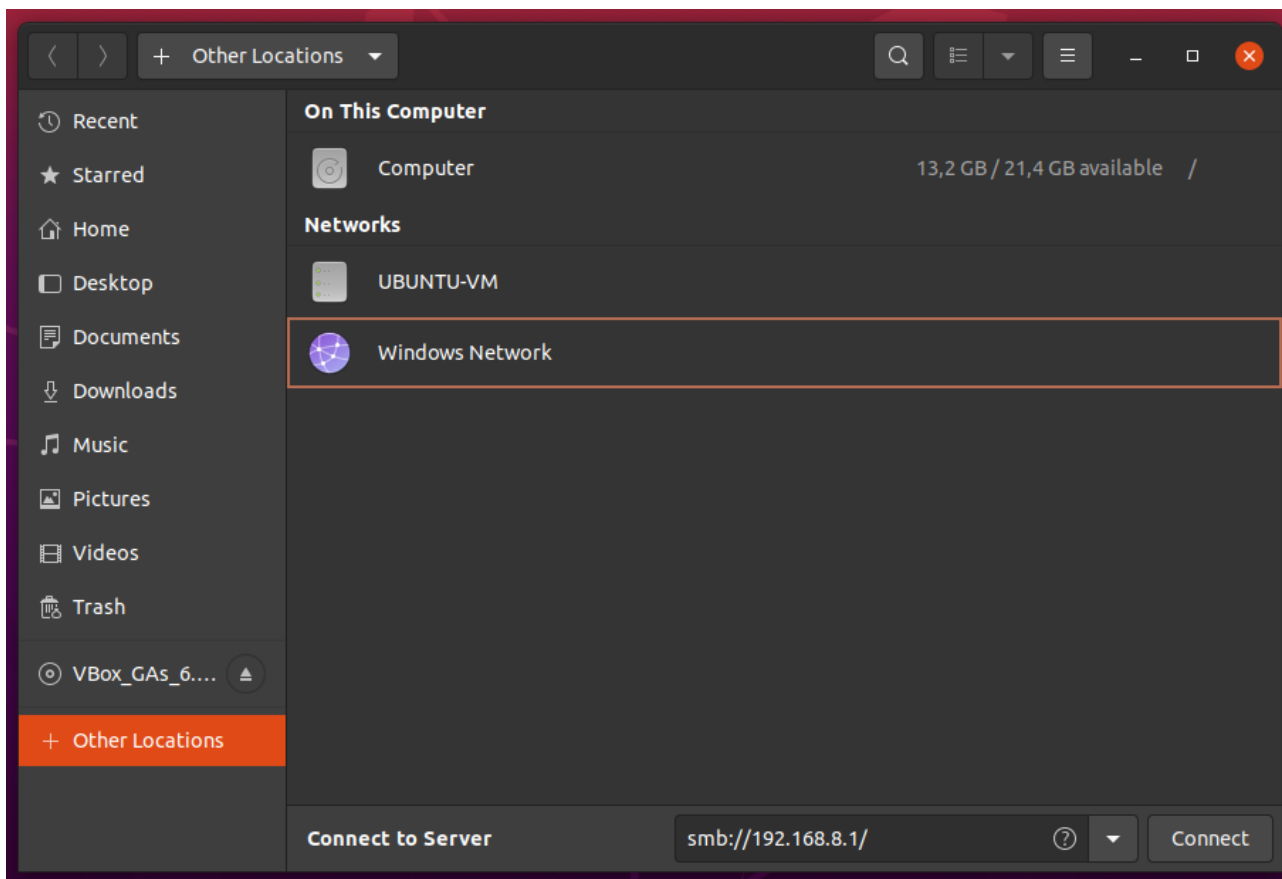
W kolejnym kroku edytowano plik `/etc/samba/smb.conf`. Na końcu pliku dopisano konfigurację, która udostępni wcześniej utworzony folder. Na koniec zrestartowano usługę.

```
jakub@ubuntu-vm:~$ sudo vim /etc/samba/smb.conf
jakub@ubuntu-vm:~$ sed -n 240,250p /etc/samba/smb.conf
# to the drivers directory for these users to have write rights in it
; write list = root, @lpadmin

[Public]
  path = /data/public
  browseable = yes
  guest ok = yes
  read only = no
  create mask = 777
jakub@ubuntu-vm:~$ sudo service smb restart
jakub@ubuntu-vm:~$ sudo service nmb restart
jakub@ubuntu-vm:~$
```

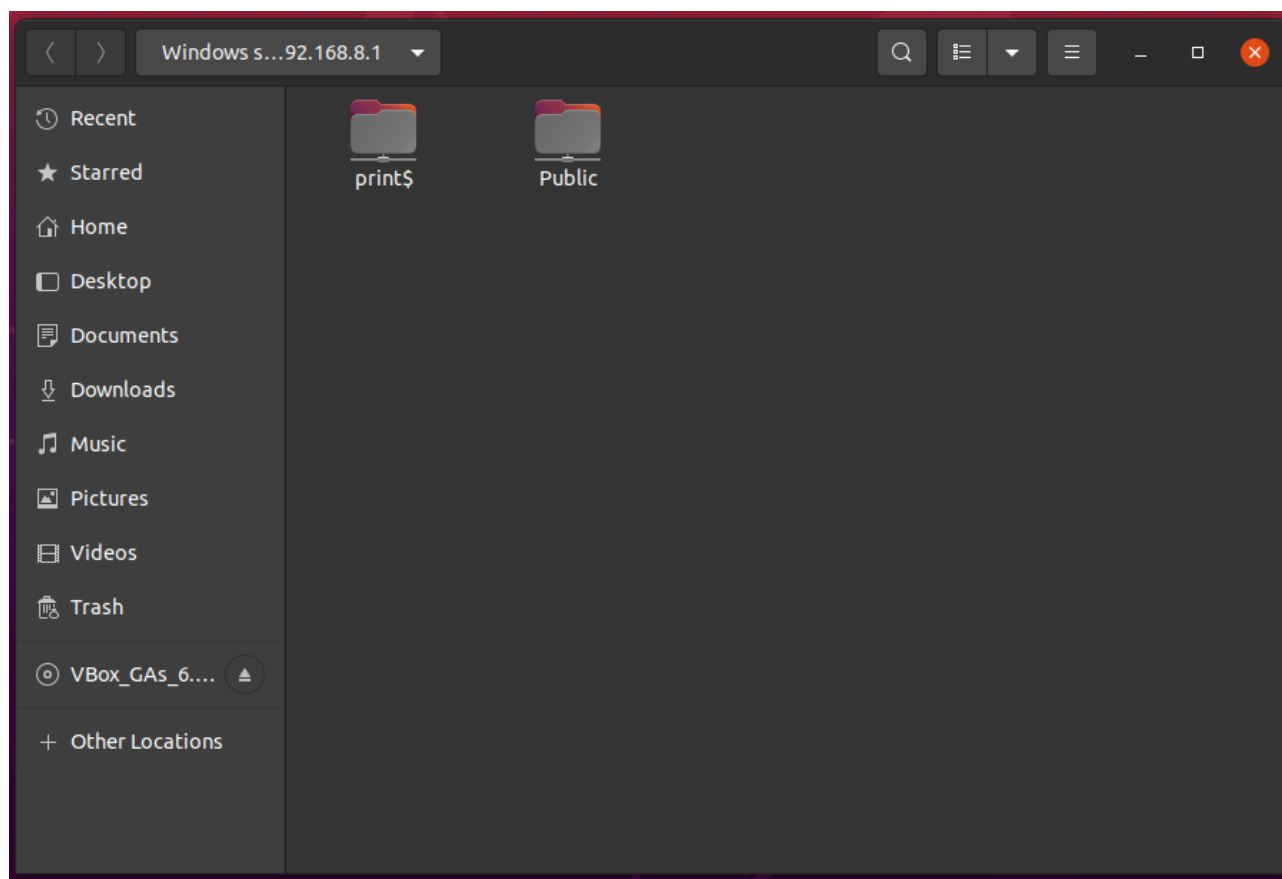
Rysunek 8.2: Konfiguracja serwera plików.

Na kliencie następnie wykonano próbę połączenia z serwerem plików. W tym celu otworzono inne lokalizacje w eksploratorze plików i wpisano `smb://192.168.8.1/`.



Rysunek 8.3: Próba nawiązania połączenia z serwerem plików z klienta.

Po połączeniu ukazał się utworzony folder publiczny.



Rysunek 8.4: Rezultat końcowy konfiguracji serwera plików.

9 Serwer DNS

Na początku zainstalowano odpowiednie pakiety. Kolejnym krokiem było ustawienie w pliku `/etc/bind/named.conf.options` router jako cache serwera DNS 8.8.8.8.

```
jakub@ubuntu-vm:~$ sudo apt-get install bind9 dnstools
Reading package lists... Done
Building dependency tree
Reading state information... Done
bind9 is already the newest version (1:9.16.1-0ubuntu2.8).
dnstools is already the newest version (1:9.16.1-0ubuntu2.8).
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
jakub@ubuntu-vm:~$ sudo vim /etc/bind/named.conf.options
jakub@ubuntu-vm:~$ cat /etc/bind/named.conf.options
options {
    directory "/var/cache/bind";

    // If there is a firewall between you and nameservers you want
    // to talk to, you may need to fix the firewall to allow multiple
    // ports to talk.  See http://www.kb.cert.org/vuls/id/800113

    // If your ISP provided one or more IP addresses for stable
    // nameservers, you probably want to use them as forwarders.
    // Uncomment the following block, and insert the addresses replacing
    // the all-0's placeholder.

    forwarders {
        8.8.8.8;
    };

    //=====
    // If BIND logs error messages about the root key being expired,
    // you will need to update your keys.  See https://www.isc.org/bind-keys
    //=====
    dnssec-validation auto;

    listen-on-v6 { any; };
};
jakub@ubuntu-vm:~$ sudo systemctl restart bind9
jakub@ubuntu-vm:~$ |
```

Rysunek 9.1: Instalacja pakietu dns oraz ustawienie serwera jako cache.

Na początku skonfigurowano aby domeny były przypisane do IP tzw. Forward zone (domena → IP). Dla przykładu wybrano tutaj domenę „router.com”.

```
jakub@ubuntu-vm:~$ cat /etc/bind/named.conf
// This is the primary configuration file for the BIND DNS server named.
//
// Please read /usr/share/doc/bind9/README.Debian.gz for information on the
// structure of BIND configuration files in Debian, *BEFORE* you customize
// this configuration file.
//
// If you are just adding zones, please do that in /etc/bind/named.conf.local

include "/etc/bind/named.conf.options";
include "/etc/bind/named.conf.local";
include "/etc/bind/named.conf.default-zones";
jakub@ubuntu-vm:~$ |
```

Rysunek 9.2: Główny plik konfiguracyjny serwera DNS.

```
jakub@ubuntu-vm:~$ sudo vim /etc/bind/named.conf.local
[sudo] password for jakub:
jakub@ubuntu-vm:~$ sudo cp /etc/bind/db.local /etc/bind/db.router.com
jakub@ubuntu-vm:~$ cat /etc/bind/named.conf.local
//
// Do any local configuration here
//
// Consider adding the 1918 zones here, if they are not used in your
// organization
//include "/etc/bind/zones.rfc1918";

zone "router.com" {
    type master;
    file "/etc/bind/db.router.com";
}

jakub@ubuntu-vm:~$ sudo vim /etc/bind/db.router.com
jakub@ubuntu-vm:~$ sudo vim /etc/bind/db.router.com
jakub@ubuntu-vm:~$ cat /etc/bind/db.router.com
;
; BIND data file for local loopback interface
;
$TTL      604800
@         IN      SOA      router. root.router.com. (
                        2          ; Serial
                        604800     ; Refresh
                        86400      ; Retry
                        2419200    ; Expire
                        604800 )   ; Negative Cache TTL
;
@         IN      A        192.168.8.1
;
@         IN      NS       ns.router.com.
@         IN      A        192.168.8.1
@         IN      AAAA     ::1
ns        IN      A        192.168.8.1
jakub@ubuntu-vm:~$ sudo systemctl restart bind9
jakub@ubuntu-vm:~$
```

Rysunek 9.3: Skonfigurowanie „forward zone”.

W kolejnym kroku skonfigurowano tzw. Reverse zone. Oznacza to przypisanie IP do domeny (IP → domena).

```
jakub@ubuntu-vm:~$ sudo vim /etc/bind/named.conf.local
jakub@ubuntu-vm:~$ cat /etc/bind/named.conf.local
//
// Do any local configuration here
//
// Consider adding the 1918 zones here, if they are not used in your
// organization
//include "/etc/bind/zones.rfc1918";

zone "router.com" {
    type master;
    file "/etc/bind/db.router.com";
};

zone "192.168.8.in-addr.arpa" {
    type master;
    file "/etc/bind/db.10";
};
jakub@ubuntu-vm:~$ sudo cp /etc/bind/db.127 /etc/bind/db.10
jakub@ubuntu-vm:~$ sudo vim /etc/bind/db.10
jakub@ubuntu-vm:~$ cat /etc/bind/db.10
;
; BIND reverse data file for local loopback interface
;
$TTL      604800
@         IN      SOA      ns.router.com. root.router.com. (
                        1          ; Serial
                        604800     ; Refresh
                        86400      ; Retry
                        2419200    ; Expire
                        604800 )   ; Negative Cache TTL
;
@         IN      NS       ns.
1.0.0    IN      PTR      ns.router.com.
jakub@ubuntu-vm:~$
```

Rysunek 9.4: Skonfigurowanie „reverse zone”.

Na koniec sprawdzono poprawność konfiguracji serwera DNS.


```
jakub@ubuntu-vm:~$ named-checkzone router.com /etc/bind/db.router.com
zone router.com/IN: loaded serial 2
OK
jakub@ubuntu-vm:~$ named-checkzone 192.168.8.0/24 /etc/bind/db.10
zone 192.168.8.0/24/IN: loaded serial 1
OK
jakub@ubuntu-vm:~$ named-checkconf /etc/bind/named.conf.local
jakub@ubuntu-vm:~$ named-checkconf /etc/bind/named.conf
jakub@ubuntu-vm:~$
```

Rysunek 9.5: Sprawdzenie poprawności konfiguracji serwera DNS.