

## Instalujemy PHP:

```
sudo apt-get install php libapache2-mod-php
```

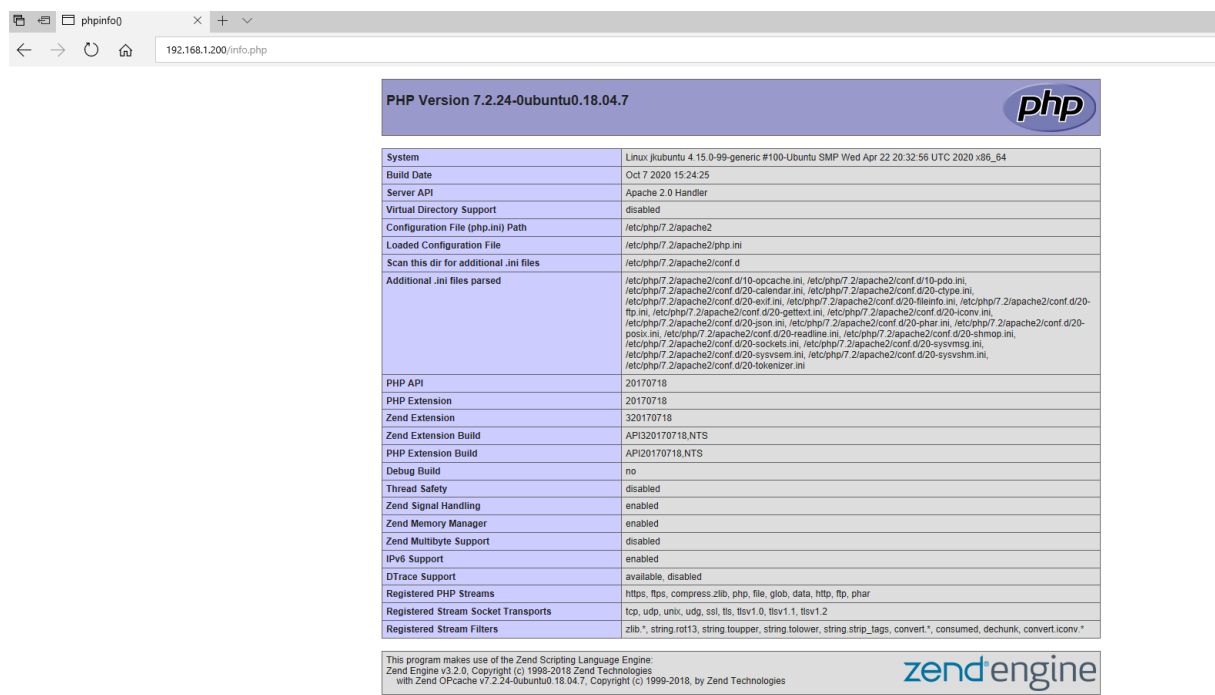
Robimy pliczek testowy PHP gdzie wywołujemy funkcje wyświetlającą informacje na temat PHP w naszym systemie:

```
sudo nano /var/www/html/info.php
```

```
<?php
    phpinfo();
?>
```

restartujemy Apache i testujemy:

```
sudo systemctl restart apache2
```



PHP Version 7.2.24-0ubuntu0.18.04.7	
System	Linux jkubuntu 4.15.0-99-generic #100-Ubuntu SMP Wed Apr 22 20:32:56 UTC 2020 x86_64
Build Date	Oct 7 2020 15:24:25
Server API	Apache 2.0 Handler
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc/php/7.2/apache2
Loaded Configuration File	/etc/php/7.2/apache2/php.ini
Scan this dir for additional .ini files	/etc/php/7.2/apache2/conf.d
Additional .ini files parsed	/etc/php/7.2/apache2/conf.d/10-opcache.ini, /etc/php/7.2/apache2/conf.d/10-pdo.ini, /etc/php/7.2/apache2/conf.d/20-calendar.ini, /etc/php/7.2/apache2/conf.d/20-ctype.ini, /etc/php/7.2/apache2/conf.d/20-enf.ini, /etc/php/7.2/apache2/conf.d/20-fileinfo.ini, /etc/php/7.2/apache2/conf.d/20-ftp.ini, /etc/php/7.2/apache2/conf.d/20-gettext.ini, /etc/php/7.2/apache2/conf.d/20-iconv.ini, /etc/php/7.2/apache2/conf.d/20-json.ini, /etc/php/7.2/apache2/conf.d/20-ldap.ini, /etc/php/7.2/apache2/conf.d/20-openssl.ini, /etc/php/7.2/apache2/conf.d/20-readline.ini, /etc/php/7.2/apache2/conf.d/20-shmop.ini, /etc/php/7.2/apache2/conf.d/20-sockets.ini, /etc/php/7.2/apache2/conf.d/20-sysmsg.ini, /etc/php/7.2/apache2/conf.d/20-syssem.ini, /etc/php/7.2/apache2/conf.d/20-sysvshm.ini, /etc/php/7.2/apache2/conf.d/20-tokenizer.ini
PHP API	20170718
PHP Extension	20170718
Zend Extension	320170718
Zend Extension Build	API320170718.NTS
PHP Extension Build	API20170718.NTS
Debug Build	no
Thread Safety	disabled
Zend Signal Handling	enabled
Zend Memory Manager	enabled
Zend Multibyte Support	disabled
IPv6 Support	enabled
DTrace Support	available, disabled
Registered PHP Streams	https, ftps, compress.zlib, php, file, glob, data, http, ftp, phar
Registered Stream Socket Transports	tcp, udp, unix, udg, ssl, tls, tlsv1.0, tlsv1.1, tlsv1.2
Registered Stream Filters	zlib.*, string.rot13, string.toupper, string.tolower, string.strip_tags, convert.*, consumed, dechunk, convert.iconv.*

This program makes use of the Zend Scripting Language Engine:  
Zend Engine v2.2.0, Copyright (c) 1998-2018 Zend Technologies  
with Zend OPcache v7.2.24-0ubuntu0.18.04.7, Copyright (c) 1999-2018, by Zend Technologies

## Ustawienie, żeby pierwszym otwieranym plikiem był index.php a nie index.html

```
sudo nano /etc/apache2/mods-enabled/dir.conf
```

przestawiamy index.php na pierwszą pozycję w tej sekcji:

```
<IfModule mod_dir.c>
    DirectoryIndex index.php index.html index.cgi index.pl index.xhtml index.htm
</IfModule>
```



```
Plik Maszyna Widok Wejście Urządzenia Pomoc
GNU nano 2.9.3 dir.conf
<IfModule mod_dir.c>
    DirectoryIndex index.php index.html index.cgi index.pl index.xhtml index.htm
</IfModule>
# vim: syntax=apache ts=4 sw=4 sts=4 sr noet
```

Tworzymy w katalogu strony domyślnej Apache pliczek index.php

```
nano /var/www/html/index.php
```

i wpisujemy taką zawartość:

```
<?php
echo "strona domyslna w PHP !!!!";
?>
```

Po zapisaniu możemy przetestować – mimo że w katalogu są teraz dwa pliki index:

```
root@jkubuntu:/var/www/html# ls -l
total 20
-rwxr-xr-x 1 root root 10904 lis 20 00:02 index.html
-rw-r--r-- 1 root root 46 lis 24 14:07 index.php
-rw-r--r-- 1 root root 44 lis 24 12:02 info.php
```

to po wpisaniu adresu serwera zamiast tej strony domyślnej Ubuntu powinien się wyświetlić nasz pliczek PHP



## Włączenie katalogów użytkowników – użytkownicy będą mogli umieszczać strony WWW w swoich katalogach

Włączamy mod do Apache2 odpowiedzialny za tą funkcję

```
sudo a2enmod userdir
```

Nie trudno się domyślić, że konfiguracja tego moda znajduje się w katalogu mods-enabled, jak zajrzemy tam, to widać, że strony użytkowników będą przechowywane w ich katalogach domowych w podkatalogu public\_html

```
nano /etc/apache2/mods-enabled/userdir.conf
```

```
GNU nano 2.9.3 userdir.conf
<IfModule mod_userdir.c>
  UserDir public_html
  UserDir disabled root

  <Directory /home/*/public_html>
    AllowOverride FileInfo AuthConfig Limit Indexes
    Options MultiViews Indexes SymLinksIfOwnerMatch IncludesNoExec
    Require method GET POST OPTIONS
  </Directory>
</IfModule>
```

Jeżeli byśmy chcieli żeby na przykład podkatalog na strony to był np.: user\_sites, to oczywiście w powyższy pliku można by to przekonfigurować – my zostawiamy tak jak jest

**Przelogujemy się** na zwykłego użytkownika, np. **zsme (stworzcie sobie usera inazwisko i robcie to z jego konta)** i stwórzmy katalog public\_html w jego katalogu domowym

```
mkdir ~/public_html
```

Zróbmy index.html i sprawdzamy czy działa (ukradnijmy z któregoś hosta wirtualnego z poprzedniego ćwiczenia ):

```
cp /var/www/zsme.local/html/index.html /home/zsme/public_html/index.html
```

Edytujemy ten pliczek do takiej postaci:

```
GNU nano 2.9.3 /home/zsme/public_html/index.html
<html>
<head></head>
<body>
<h1>Strona uzytkownika zsme</h1>
</body>
</html>
```

teraz po wpisaniu adresu naszego serwera w ten sposób

```
http://192.168.1.200/~zsme/index.html
```

powinna się zgłosić strona danego usera (w naszym przypadku to będzie user zsme)



## Strona uzytkownika zsme

### PHP z katalogów użytkowników:

Domyślnie obsługa PHP w katalogach użytkowników jest wyłączona – jeżeli użytkownik zrobi jakiś plik PHP w swoim katalogu public\_html i spróbuje go wywołać, ten po prostu nie zadziała. Żeby włączyć tą obsługę musimy wyedytować ustawienia moda php, jak wszystkie inne mody znajdują się one oczywiście w katalogu mods-enabled w ustawieniach Apache:

```
nano /etc/apache2/mods-enabled/php7.2.conf
```

```
# Running PHP scripts in user directories is disabled by default
#
# To re-enable PHP in user directories comment the following lines
# (from <IfModule ...> to </IfModule>.) Do NOT set it to On as it
# prevents .htaccess files from disabling it.
<IfModule mod_userdir.c>
  <Directory /home/*/public_html>
    php_admin_flag engine Off
  </Directory>
</IfModule>
```

Zgodnie z informacją z komentarza komentujemy 5 linijek, zapisujemy plik i restartujemy Apache.

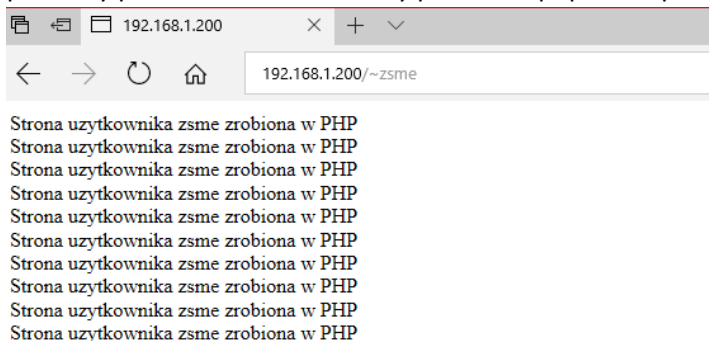
```
# Running PHP scripts in user directories is disabled by default
#
# To re-enable PHP in user directories comment the following lines
# (from <IfModule ...> to </IfModule>.) Do NOT set it to On as it
# prevents .htaccess files from disabling it.
#<IfModule mod_userdir.c>
#  <Directory /home/*/public_html>
#    php_admin_flag engine Off
#  </Directory>
#</IfModule>
```

Teraz otwieramy sobie drugą konsolę (Ctrl+Alt+F2) i logujemy się na zsmc (wy oczywiście na swojego usera inazwisko) i robimy pliczek testowy php o nazwie index.php z pętlą:

```
nano ~/public_html/index.php
```

```
<?php
for($i=0;$i<10;$i++){
echo "Strona uzytkownika zsmc zrobiona w PHP<br>";
}
?>
```

Po wpisaniu w przeglądarkę adresu strony użytkownika zsmc zgodnie z wcześniejszą konfiguracją pierwszy powinien zostać otwarty plik index.php i interpreter PHP powinien wykonać naszą pętlę:



```
192.168.1.200/~zsmc
Strona uzytkownika zsmc zrobiona w PHP
Strona uzytkownika zsmc zrobiona w PHP
Strona uzytkownika zsmc zrobiona w PHP
Strona uzytkownika zsmc zrobiona w PHP
Strona uzytkownika zsmc zrobiona w PHP
Strona uzytkownika zsmc zrobiona w PHP
Strona uzytkownika zsmc zrobiona w PHP
Strona uzytkownika zsmc zrobiona w PHP
Strona uzytkownika zsmc zrobiona w PHP
```

## Wyświetlanie błędów w plikach PHP

Domyślnie PHP ma wyłączone wyświetlanie komunikatów o błędach które zrobimy w plikach php, zamiast nich wyświetla pustą stronę - na serwerze produkcyjnym to podejście jest jak najbardziej w porządku ze względu na bezpieczeństwo naszej strony (po co ktoś ma znać nazwy naszych zmiennych itd.).

Jako użytkownik zsmę skopiuję sobie nasz plik index.php i nazwę kopie blad.php

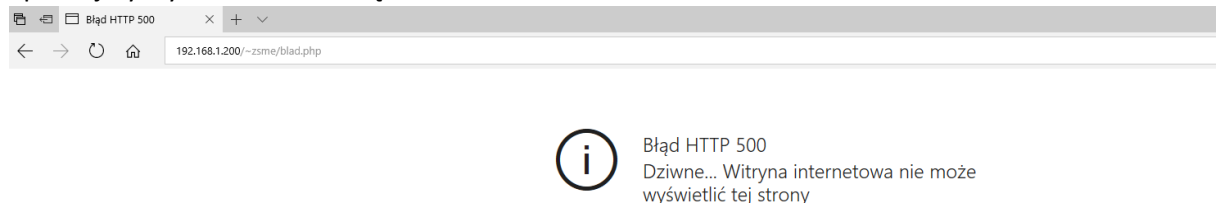
```
cp ~/public_html/index.php ~/public_html/blad.php
```

```
zsmę@jkubuntu:~/public_html$ ls -l
total 12
-rw-r--r-- 1 zsmę zsmę 87 lis 24 16:26 blad.php
-rwxr-xr-x 1 zsmę zsmę 88 lis 24 15:05 index.html
-rw-r--r-- 1 zsmę zsmę 87 lis 24 15:17 index.php
zsmę@jkubuntu:~/public_html$
```

Zróbmy teraz w naszym pliku blad.php specjalnego popularnego błędem – usuńmy średnik

```
GNU nano 2.9.3 blad.php
<?php
for($i=0;$i<10;$i++){
    echo "Strona uzytkownika zsmę zrobiona w PHP</br>"
}
?>
```

Spróbujmy wyświetlić ta stronę:



Strona po prostu się nie wyświetla (w innych przeglądarkach może to być pusta strona). Natomiast dla naszej wygody, jako że to środowisko służące tylko do ćwiczeń, dobrze sobie włączyć obsługę wyświetlania błędów.

Musimy włączyć wyświetlanie komunikatów przez PHP w pliku php.ini

```
nano /etc/php/7.2/apache2/php.ini
```

Szukamy w pliku sekcji dotyczące błędów (sam plik jest mocno rozbudowany)

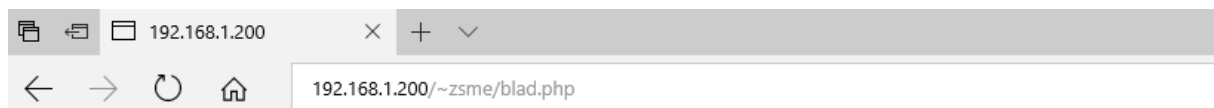
```
GNU nano 2.9.3 /etc/php/7.2/apache2/php.ini
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
; Error handling and logging ;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

; This directive informs PHP of which errors, warnings and notices you would like
; it to take action for. The recommended way of setting values for this
; directive is through the use of the error level constants and bitwise
; operators. The error level constants are below here for convenience as well as
; some common settings and their meanings.
; By default, PHP is set to take action on all errors, notices and warnings EXCEPT
; those related to E_NOTICE and E_STRICT, which together cover best practices and
; recommended coding standards in PHP. For performance reasons, this is the
; recommend error reporting setting. Your production server shouldn't be wasting
; resources complaining about best practices and coding standards. That's what
; development servers and development settings are for.
; Note: The php.ini-development file has this setting as E_ALL. This
; means it pretty much reports everything which is exactly what you want during
; development and early testing.
;
; Error Level Constants:
; E_ALL          - All errors and warnings (includes E_STRICT as of PHP 5.4.0)
; E_ERROR        - fatal run-time errors
; E_RECOVERABLE_ERROR  - almost fatal run-time errors
; E_WARNING      - run-time warnings (non-fatal errors)
; E_PARSE        - compile-time parse errors
; E_NOTICE       - run-time notices (these are warnings which often result
;                  from a bug in your code, but it's possible that it was
;                  intentional (e.g., using an uninitialized variable and
;                  relying on the fact it is automatically initialized to an
;                  empty string)
;
```

I poniżej ustawiamy zmienną **display\_errors = On**

```
; stderr = Display errors to STDERR (affects only CGI/CLI binaries!)
; On or stdout = Display errors to STDOUT
; Default Value: On
; Development Value: On
; Production Value: Off
; http://php.net/display-errors
display_errors = On
```

Zapisujemy, restartujemy Apache i sprawdzamy jak teraz działa



**Parse error: syntax error, unexpected '}', expecting ',' or ';' in /home/zsme/public\_html/blad.php on line 5**

Jak widać, teraz widzimy dokładną informację o błędzie – w naszym przypadku braku średnika na końcu 5 linijki.

Ściągamy ISO z systemem z adresu:

<https://releases.ubuntu.com/18.04/ubuntu-18.04.4-live-server-amd64.iso>

lub z Sharepointa:

[https://zsmetarnow.sharepoint.com/:u:/s/PP\\_2i\\_projektowanie\\_sieci\\_komputerowych/EapMGOVI5ZFPLIAvW6XAp7cBpqpZ7BwpfnoD9blTADGPtQ?e=rXFybc](https://zsmetarnow.sharepoint.com/:u:/s/PP_2i_projektowanie_sieci_komputerowych/EapMGOVI5ZFPLIAvW6XAp7cBpqpZ7BwpfnoD9blTADGPtQ?e=rXFybc)

Tworzymy nową maszynę w Virtual Box z takimi ustawieniami:

Name:

Typ:

Wersja:

Name:

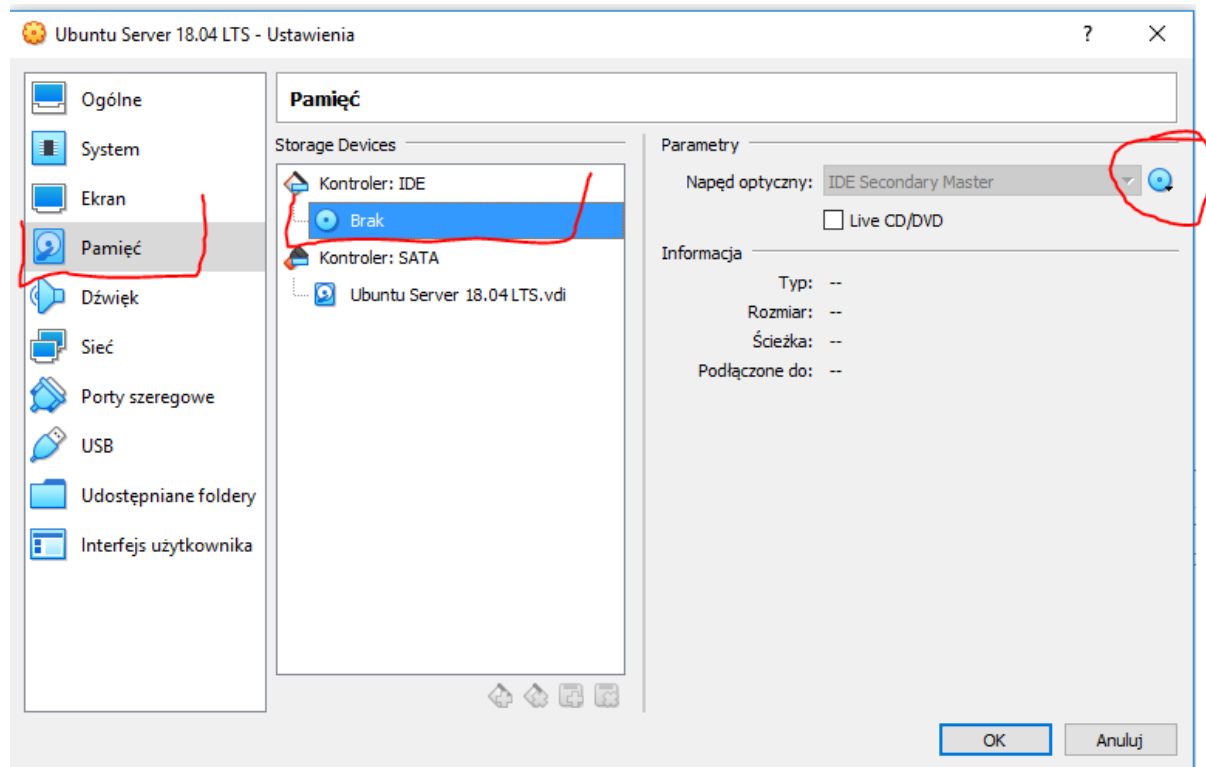
Typ:

Wersja:

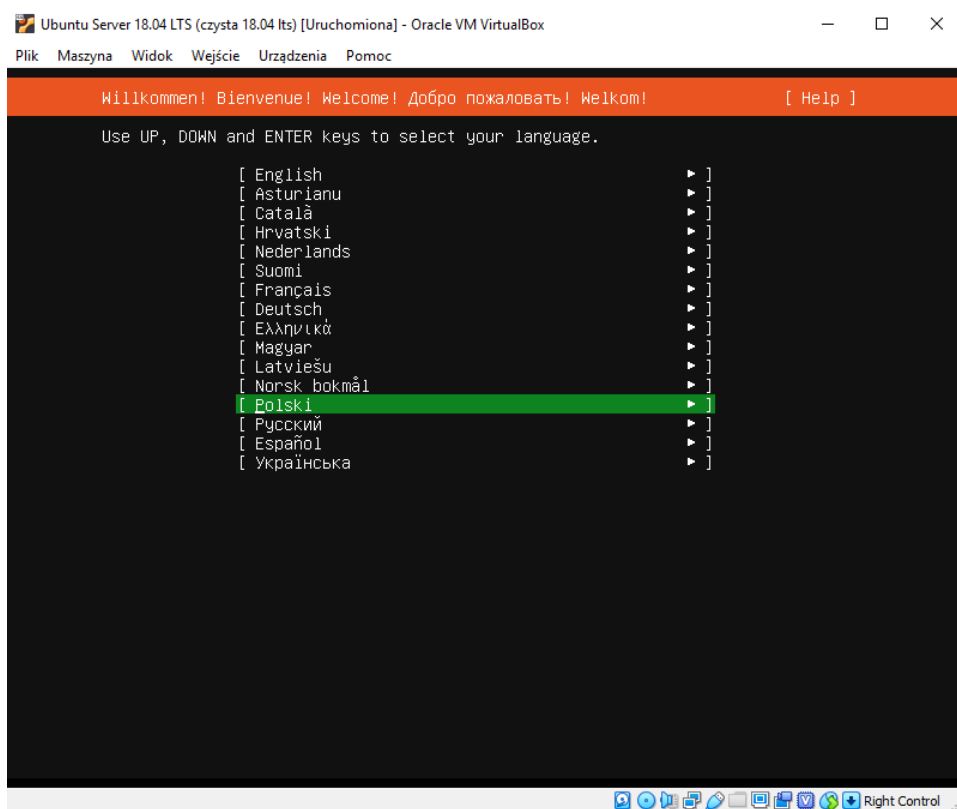
Sieciówka

może zostać w trybie NAT

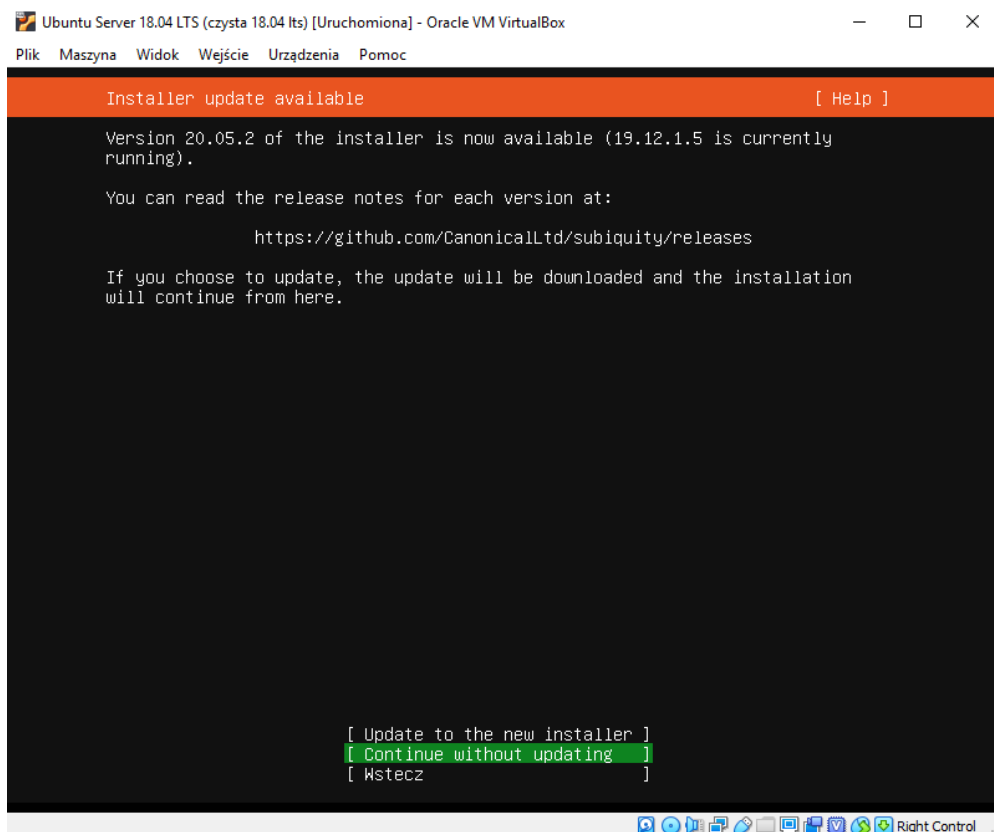
Wsadzamy plik ISO do wirtualnego napędu w maszynie:



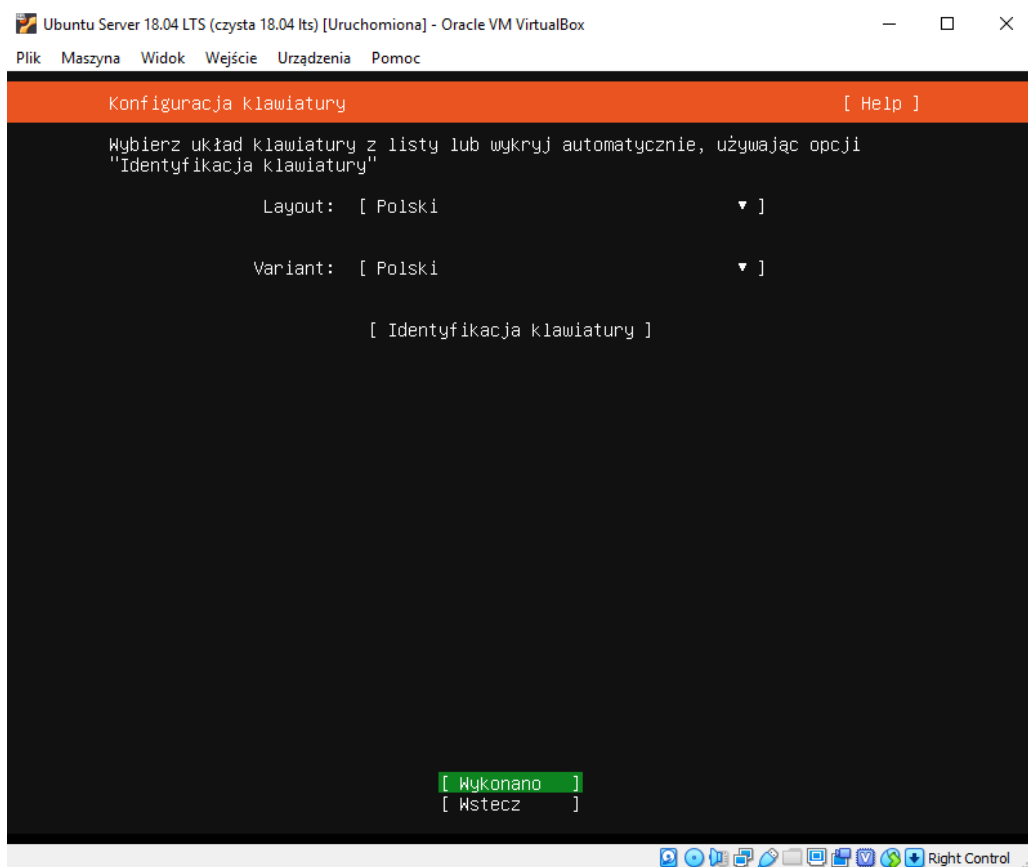
## Wybieramy język:



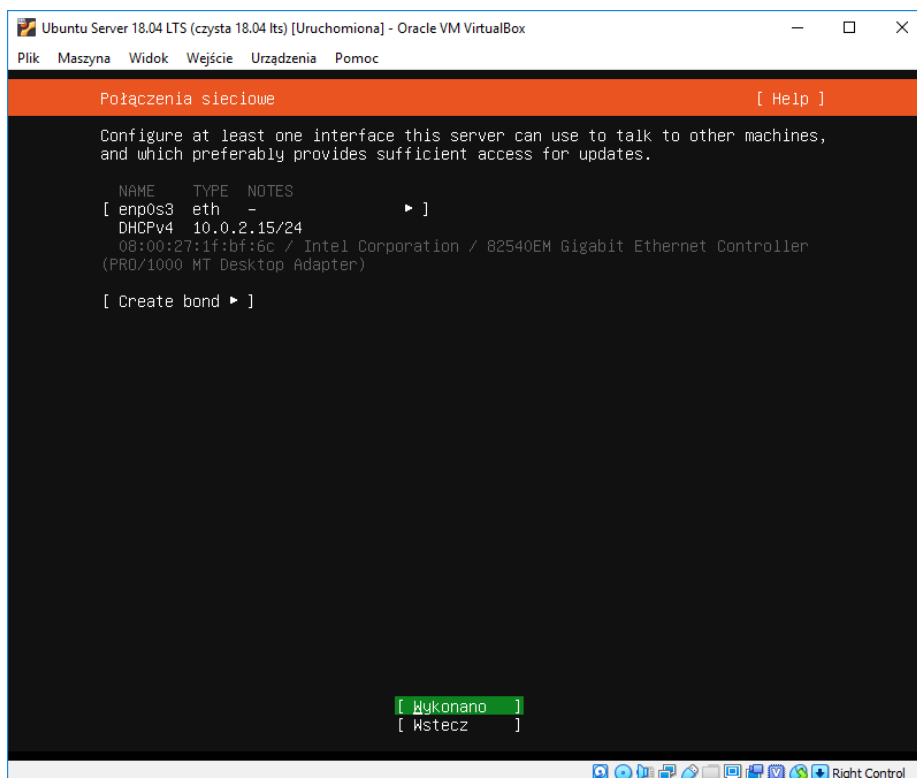
## Nie robimy update podczas instalacji:



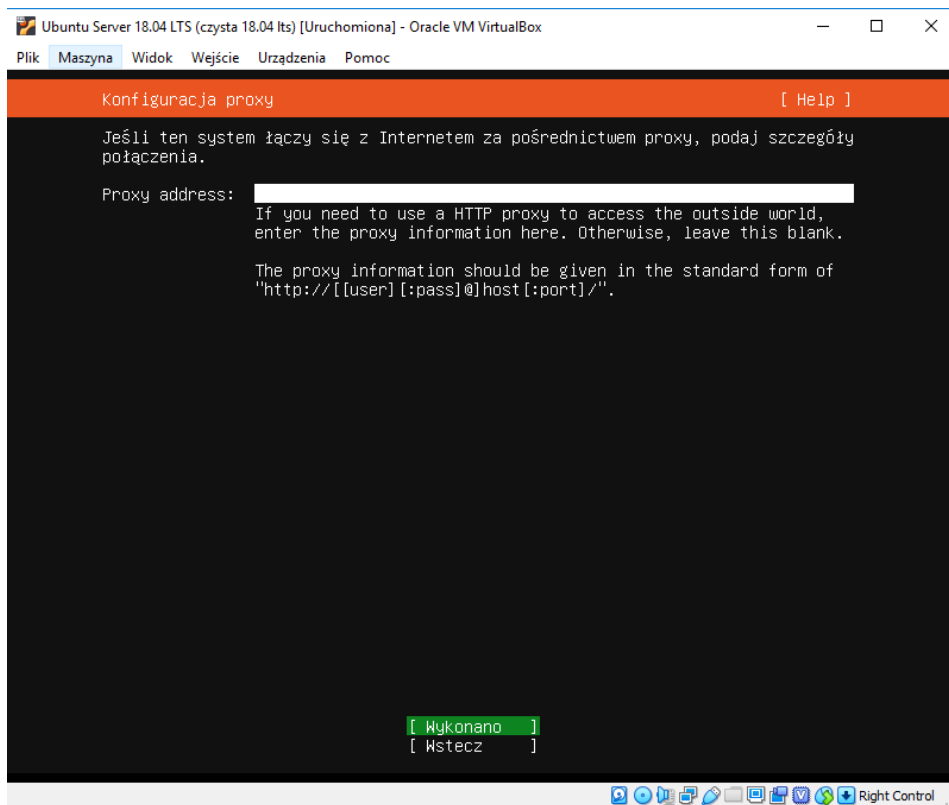
## Konfigurujemy klawiaturę:



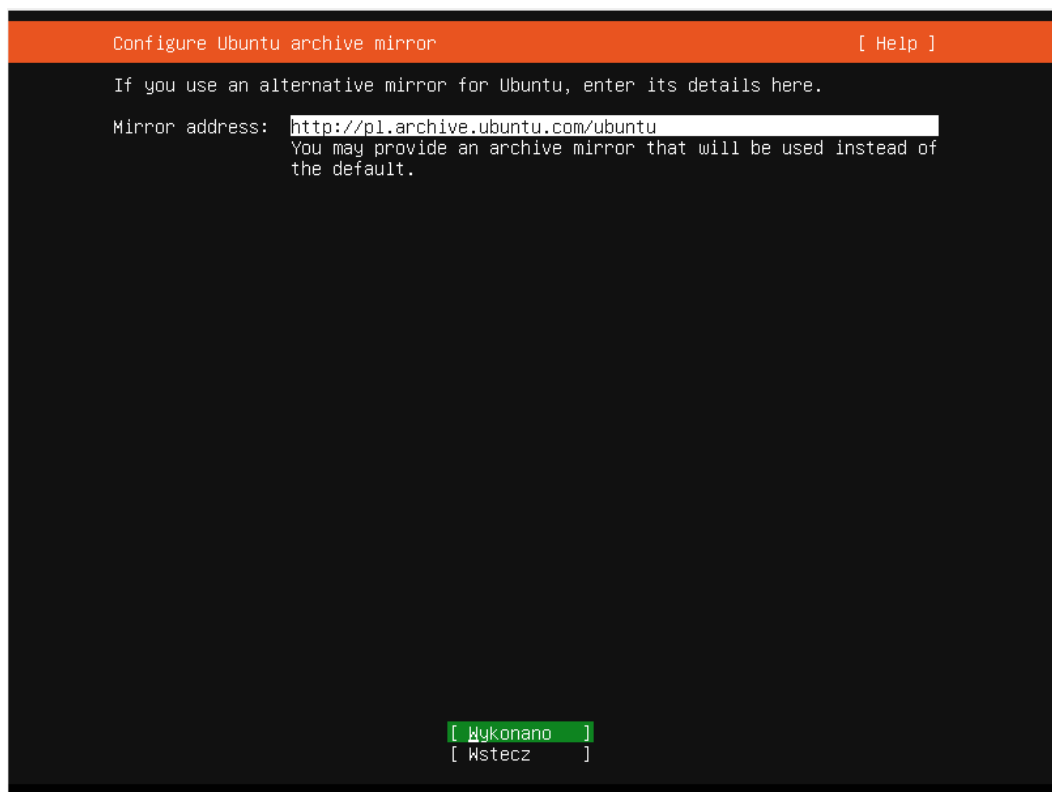
## Sieciówki zostawiamy tak jak są:



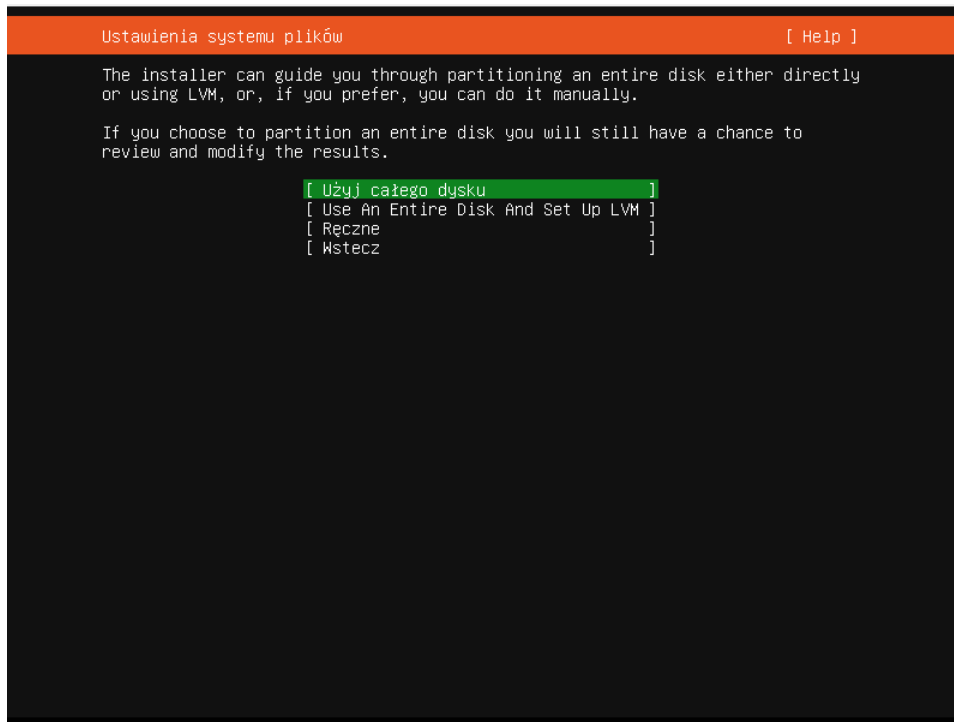
## Nie wpisujemy PROXY



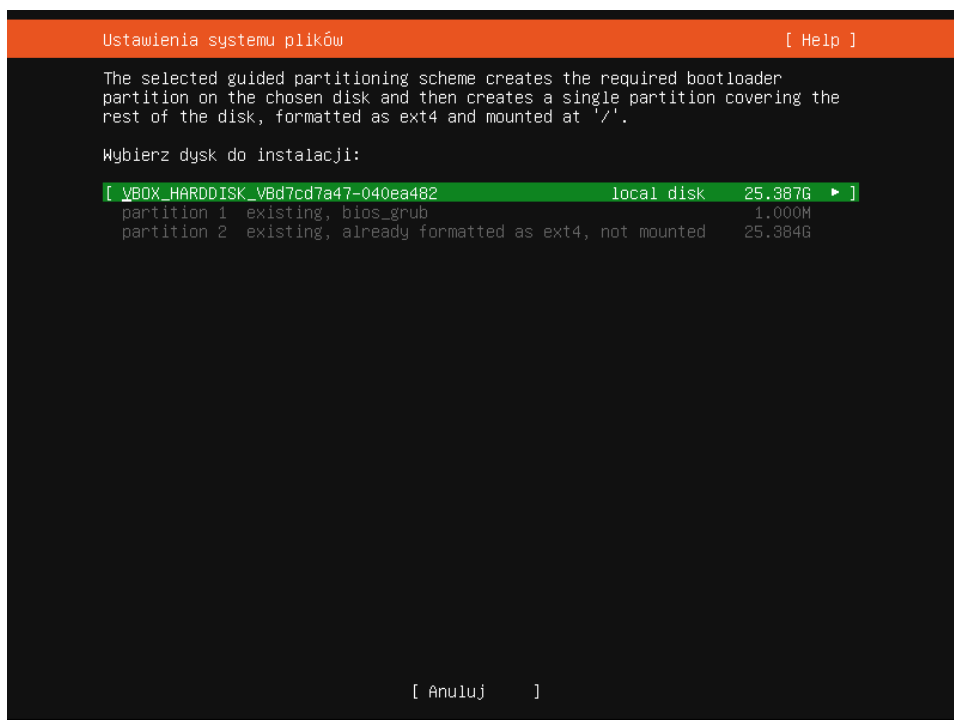
## Pozostawiamy bez zmian Ubuntu archive mirror



Do zainstalowania używamy całej powierzchni dysku



Wybieramy nasz dysk:



Potwierdzamy:

```
Ustawienia systemu plików [ Help ]
PODSUMOWANIE SYSTEMU PLIKÓW
MOUNT POINT   Rozmiar   Typ      Typ urządzenia
[ /           25.384G  new ext4 new partition of local disk ▶ ]

DOSTĘPNE URZĄDZENIA
No available devices

[ Create software RAID (md) ▶ ]
[ Create volume group (LVM) ▶ ]

USED DEVICES
Urządzenie                               Typ      Rozmiar
[ VBOX_HARDDISK_VBd7cd7a47-040ea482      local disk 25.387G ▶ ]
partition 1 new, bios_grub                          1.000M ▶ ]
partition 2 new, to be formatted as ext4, mounted at / 25.384G ▶ ]

[ Wykonano ]
[ Przywróć ]
[ Wstecz   ]
```

```
Ustawienia systemu plików [ Help ]
PODSUMOWANIE SYSTEMU PLIKÓW
MOUNT POINT   Rozmiar   Typ      Typ urządzenia
[ /           25.384G  new ext4 new partition of local disk ▶ ]

DOSTĘPNE URZĄDZENIA
No available devices

Potwierdź destrukcyjne działanie

Selecting Continue below will begin the installation process and
result in the loss of data on the disks selected to be formatted.

You will not be able to return to this or a previous screen once the
installation has started.

Are you sure you want to continue?

[ Nie       ]
[ Kontynuuj ]

[ Wykonano ]
[ Przywróć ]
[ Wstecz   ]
```

Konfigurujemy nazwę komputera i loginy/hasła

Imię: wpiszcie swoje

Nazwa serwera: **in**ubuntu czyli dla **Jan Kowalski** nazwa to **jk**ubuntu (pierwsze litery imienia i nazwiska plus ubuntu)

Login/hasło : zsme/zsme

Ustawienia profilu [ Help ]

Enter the username and password you will use to log in to the system. You can configure SSH access on the next screen but a password is still needed for sudo.

Your name:

Your server's name:   
The name it uses when it talks to other computers.

Pick a username:

Choose a password:

Confirm your password:

[ Wykonano ]

## Nie instalujemy OpenSSH

```
SSH Setup [ Help ]

You can choose to install the OpenSSH server package to enable secure remote
access to your server.

[ ] Install OpenSSH server

Import SSH identity: [ No ▼ ]
You can import your SSH keys from Github or Launchpad.

Import Username:

[X] Allow password authentication over SSH

[ Wykonano ]
[ Wstecz ]
```

## Nie instalujemy dodatkowych pakietów:

```
Featured Server Snaps [ Help ]

These are popular snaps in server environments. Select or deselect with SPACE,
press ENTER to see more details of the package, publisher and versions
available.

( ) microk8s      Kubernetes for workstations and appliances ▶
( ) nextcloud     Nextcloud Server - A safe home for all your data ▶
( ) wekan         Open-Source kanban ▶
( ) kata-containers Lightweight virtual machines that seamlessly plug int ▶
( ) docker        Docker container runtime ▶
( ) canonical-livepatch Canonical Livepatch Client ▶
( ) rocketchat-server Group chat server for 100s, installed in seconds. ▶
( ) mosquitto     Eclipse Mosquitto MQTT broker ▶
( ) etcd          Resilient key-value store by CoreOS ▶
( ) powershell   PowerShell for every system! ▶
( ) stress-ng     A tool to load, stress test and benchmark a computer ▶
( ) sabnzbd       SABnzbd ▶
( ) wormhole      get things from one computer to another, safely ▶
( ) aws-cli       Universal Command Line Interface for Amazon Web Servi ▶
( ) google-cloud-sdk Command-line interface for Google Cloud Platform prod ▶
( ) slcli         Python based SoftLayer API Tool. ▶
( ) doctl         DigitalOcean command line tool ▶
( ) conjure-up    Package runtime for conjure-up spells ▶
( ) minidlna-escoand server software with the aim of being fully compliant ▶
( ) postgresql10 PostgreSQL is a powerful, open source object-relation ▶
( ) heroku        CLI client for Heroku ▶
( ) keepalived    High availability VRRP/BFD and load-balancing for Lin ▶
( ) prometheus    The Prometheus monitoring system and time series data ▶
( ) juju          Simple, secure and stable devops. Juju keeps complexi ▶

[ Wykonano ]
[ Wstecz ]
```

Czekamy aż instalacja się skończy :

```
Instalacja ukończona! [ Help ]

----- Instalacja ukończona! -----
running '/snap/bin/subiquity.subiquity-configure-run'
running '/snap/bin/subiquity.subiquity-configure-apt'
/snap/subiquity/1459/usr/bin/python3 true'
curtin command apt-config
curtin command in-target
running 'curtin curthooks'
curtin command curthooks
configuring apt configuring apt
installing missing packages
configuring iscsi service
configuring raid (mdadm) service
installing kernel
setting up swap
apply networking config
writing etc/fstab
configuring multipath
updating packages on target system
configuring pollinate user-agent on target
updating initramfs configuration
finalizing installation
running 'curtin hook'
curtin command hook
executing late commands
final system configuration
configuring cloud-init
restoring apt configuration
downloading and installing security updates
copying logs to installed system

[ Wyświetl pełny log ]
[ Reboot ]
```

Robimy reboot i sprawdzamy czy możemy się zalogować:

tutaj trzeba kliknąć Enter:

```
[ 25.882970] cloud-init[1597]: Generating locales (this might take a while)...
[ 26.576372] cloud-init[1597]: pl_PL.UTF-8... done
[ 26.577101] cloud-init[1597]: Generation complete.
[ 27.094824] cloud-init[1597]: Cloud-init v. 19.4-33-gbb4131a2-0ubuntu1~18.04.1 running 'modules:
onfig' at Tue, 12 May 2020 13:16:05 +0000. Up 25.73 seconds.
[ OK ] Started Apply the settings specified in cloud-config.
Starting Execute cloud user/final scripts...
ci-info: no authorized SSH keys fingerprints found for user zsme.
<14>May 12 13:16:07 ec2:
<14>May 12 13:16:07 ec2: #####
<14>May 12 13:16:07 ec2: ----BEGIN SSH HOST KEY FINGERPRINTS----
<14>May 12 13:16:07 ec2: 1024 SHA256:dJaxhpZKL+QqBQmgEpAdd6c8X10EVBSXTntLbW4Eg root@jkubuntu (DSA)
<14>May 12 13:16:07 ec2: 256 SHA256:002DuI2vyx7aUyJDREvzPVBnxM6dyIM2r0UTU6uIusw root@jkubuntu (ECDSA)
)
<14>May 12 13:16:07 ec2: 256 SHA256:5zu7YGDB+vZ2oosr8wI/E7o81pcBPM1u1HIBq8rzmUu root@jkubuntu (ED255
19)
<14>May 12 13:16:07 ec2: 2048 SHA256:7tsAz6uKCh2SA/GVswqJMFrxX2RubMrOnhJJ4rJLmg3M root@jkubuntu (RSA)
<14>May 12 13:16:07 ec2: ----END SSH HOST KEY FINGERPRINTS----
<14>May 12 13:16:07 ec2: #####
----BEGIN SSH HOST KEY KEYS----
ecdsa-sha2-nistp256 AAAAE2VjZHNhLXNoYTItbmlzdHAuNTYAAAAIbmlzdHAuNTYAAABBBLdn36ai2Yjl1aLPK0XJ8ofn4ux
u9kH89e0R7BoN698uWlikmVnkviJ3tR0wpp2bLQnDSAA08P4/KPDnzzF++ root@jkubuntu
ssh-ed25519 AAAAC3NzaC1l2D1lNTE5AAAAIKaClol.enTULdtFG77KEXKuh8ThuuS411/8u8JmXriyl root@jkubuntu
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQCVtPDA9IooTAsI/4c9P0HVp3WkTdaXe42HNQz94K2m4MN2J5W/3P1QhBXSpFr+
eDBXjIq3+XhekKzW1e10tpai0NBGxfad1axQoI+L3a1ALigZ6JfDw1fpAsa2h3s0BRad/ju9NNkmbiJiLRO3GHQ/inv3DRX1
IC3zTYM+vGpadqkKpnm2t08kNE212BmtaADROAVKqkQzok4EqzuuHQAcB4gYKwX6/miyZKF68H3M1uAp32A7Wmm19wE2W46P
SduH160n4Smm0stmXCFuRN7zba90o41LQNH/MokJcuW59rrGxX43udCl/e8B8vZQX9cU7/zJ/CgUsKLD6 root@jkubuntu
----END SSH HOST KEY KEYS----
[ 27.602406] cloud-init[1662]: Cloud-init v. 19.4-33-gbb4131a2-0ubuntu1~18.04.1 running 'modules:f
inal' at Tue, 12 May 2020 13:16:07 +0000. Up 27.45 seconds.
[ 27.603428] cloud-init[1662]: ci-info: no authorized SSH keys fingerprints found for user zsme.
[ 27.604044] cloud-init[1662]: Cloud-init v. 19.4-33-gbb4131a2-0ubuntu1~18.04.1 finished at Tue, 1
2 May 2020 13:16:07 +0000. DataSource DataSourceNoCloud [seed=/var/lib/cloud/seed/nocloud-net] [dsmod
e=net]. Up 27.59 seconds
[ OK ] Started Execute cloud user/final scripts.
[ OK ] Reached target Cloud-init target.
```

Jeżeli wszystko poszło OK to powinniśmy być w stanie się zalogować:

```
[ OK ] Started Execute cloud user/final scripts.
[ OK ] Reached target Cloud-init target.

Ubuntu 18.04.4 LTS jkubuntu tty1

jkubuntu login: zsme
Password:
Welcome to Ubuntu 18.04.4 LTS (GNU/Linux 4.15.0-99-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

System information as of Tue May 12 13:17:39 UTC 2020

System load:  0.28          Processes:            86
Usage of /:   15.3% of 24.86GB   Users logged in:    0
Memory usage: 15%          IP address for enp0s3: 10.0.2.15
Swap usage:   0%

21 packages can be updated.
0 updates are security updates.

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

zsme@jkubuntu:~$ _
```

W Ubuntu 18.04 konfiguracja interfejsów sieciowych uległa zmianie.

Nie używamy już jak dawniej pliku `/etc/network/interfaces`

Teraz do konfiguracji używa się `/etc/netplan/` i plików z rozszerzeniem `.yaml`

I tu zaczynają się schody 😊

W teorii konfiguracja jest banalna, w praktyce może nas czasem nerw złapać 😊

Pierwsze co trzeba umieć to sprawdzenie adresu ip:

**ip a**

```
root@jkubuntu:/etc/netplan# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:1f:bf:6c brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.200/24 brd 192.168.1.255 scope global enp0s3
        valid_lft forever preferred_lft forever
    inet6 fe80::a00:27ff:fe1f:bf6c/64 scope link
        valid_lft forever preferred_lft forever
```

Już widzimy, że zamiast prostych nazw interfejsów są jakieś takie dziwne.

Jak sprawdzić jak nazywają się nasze karty sieciowe:

**lshw -c network**

i poszukać nazw:

```

root@jkubuntu:/etc/netplan# lshw -c network
*-network:0
  description: Ethernet interface
  product: 82540EM Gigabit Ethernet Controller
  vendor: Intel Corporation
  physical id: 3
  bus info: pci@0000:00:03.0
  logical name: enp0s3
  version: 02
  serial: 08:00:27:1f:bf:6c
  size: 1Gbit/s
  capacity: 1Gbit/s
  width: 32 bits
  clock: 66MHz
  capabilities: pm pci_x bus_master cap_list ethernet physical tp 10bt 10bt-fd 100bt 100bt-fd 1000bt-fd autonegotiation
  configuration: autonegotiation=on broadcast=yes driver=e1000 driverversion=7.3.21-k8-NAPI duplex=full ip=192.168.1.200 latency=64 link=yes mingnt=255 multicast=yes port=twisted pair speed=1Gbit/s
  resources: irq:19 memory:f1200000-f121ffff ioport:d020(size=8)

```

Teraz tak :

Domyślny plik konfiguracyjny Netplan znajduje się w katalogu /etc/netplan. Możesz to sprawdzić za pomocą następującego polecenia:

```
ls /etc/netplan/
```

W teorii powinien być tam plik **01-netcfg.yaml** lub o podobnej nazwie a widzimy że znajduje się plik **50-cloud-init.yaml** - domyślnie w Ubuntu 18.04 instaluje się usługa cloud-init która zarządza wieloma rzeczami w tym sieciówkami - twórcy założyli że jak będziemy stawiać maszynę, to pewnie w chmurze typu Azure gdzie ta usługa ułatwi wstępną konfigurację - jeżeli odpalimy maszynę normalnie lokalnie, np. w Virtual Box, to może się okazać, mimo że zmienimy zawartość pliku yaml z konfiguracją sieciówki, to po restarcie usługa cloud-init i tak go sobie zmieni. Sam plik wygląda zasadniczo podobnie do typowego pliku z konfiguracją sieciówki z netplan

```

root@jkubuntu:/home/zsme# cat /etc/netplan/50-cloud-init.yaml
# This file is generated from information provided by the datasource. Changes
# to it will not persist across an instance reboot. To disable cloud-init's
# network configuration capabilities, write a file
# /etc/cloud/cloud.cfg.d/99-disable-network-config.cfg with the following:
# network: {config: disabled}
network:
  ethernets:
    enp0s3:
      dhcp4: true
  version: 2
root@jkubuntu:/home/zsme# _

```

Jak widać, mamy tutaj ustawioną jedną sieciówkę jako klienta dhcp.

**Pierwsze co musimy zrobić, to wyłączyć konfigurację sieci dla usługi cloud-init.**

Tworzymy nowy plik:

```
sudo nano /etc/cloud/cloud.cfg.d/99-disable-network-config.cfg
```

I wpisujemy do niego:

```
network: {config: disabled}
```

(oczywiście zapisujemy plik)

Teraz zmieniamy nazwę pliku konfiguracji sieci na taką jak powinna być bez usługi cloud-init

```
sudo mv /etc/netplan/50-cloud-init.yaml /etc/netplan/01-netcfg.yaml
```

Możemy zrestartować maszynę poleceniem reboot.

Musimy jeszcze zmienić delikatnie kolejność zmiennych w pliku.

**#####UWAGA#####**

Netplan jest bardzo czuły na wcięcia - wszystkie wcięcia musimy robić przy pomocy spacji - jak zrobicie z rozpędu Tab to przy próbie uruchomienia configa wywali błąd.

Ważna też jest kolejność zmiennych - też potrafi wywalić błąd.

**#####UWAGA#####**

nasz plik yaml wygląda tak:

```
root@jkubuntu:/home/zsme# cat /etc/netplan/01-netcfg.yaml
# This file is generated from information provided by the datasource. Changes
# to it will not persist across an instance reboot. To disable cloud-init's
# network configuration capabilities, write a file
# /etc/cloud/cloud.cfg.d/99-disable-network-config.cfg with the following:
# network: {config: disabled}
network:
  ethernets:
    enp0s3:
      dhcp4: true
  version: 2
```

A ma wyglądać tak:

```
GNU nano 2.9.3 /etc/netplan/01-netcfg.yaml
# This file is generated from information provided by the datasource. Changes
# to it will not persist across an instance reboot. To disable cloud-init's
# network configuration capabilities, write a file
# /etc/cloud/cloud.cfg.d/99-disable-network-config.cfg with the following:
# network: {config: disabled}
network:
  version: 2
  renderer: networkd
  ethernets:
    enp0s3:
      dhcp4: true
```

wszystkie wcięcia robimy spacją, muszą być na określona głębokość 😊

Po zapisaniu configu żeby ustawić go na karcie sieciowej musimy użyć polecenia:

**sudo netplan apply**

Sprawdzamy jaki jest adres przy użyciu

**ip a**

## KONFIGURACJA JEDNEJ SIECIÓWKI ZMOSTKOWANEJ pobierającej adres po DHCP z routera domowego

1. Ustawiamy w maszynie sieć jako zmostkowaną
2. Włączamy maszynę
2. Edytujemy plik yaml z konfiguracją w katalogu /etc/netplan do postaci:

```
GNU nano 2.9.3 /etc/netplan/01-netcfg.yaml
# This file is generated from information provided by the datasource. Changes
# to it will not persist across an instance reboot. To disable cloud-init's
# network configuration capabilities, write a file
# /etc/cloud/cloud.cfg.d/99-disable-network-config.cfg with the following:
# network: {config: disabled}
network:
  version: 2
  renderer: networkd
  ethernets:
    enp0s3:
      dhcp4: true
```

Jak wyskakuje ostrzeżenie że plik jest edytowany to po prostu potwierdzamy Wszystkie wcięcia robimy spacją!! Przy tabie wywali błąd.

3. Zapisujemy

4. Stosujemy zmiany przy użyciu **netplan apply**

5. Sprawdzamy czy adres został pobrany poprawnie przy użyciu **ip a**

```
root@jkubuntu:/home/zsme# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:1f:bf:6c brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.111/24 brd 192.168.1.255 scope global dynamic enp0s3
        valid_lft 86079sec preferred_lft 86079sec
    inet6 fe80::a00:27ff:fe1f:bf6c/64 scope link
        valid_lft forever preferred_lft forever
```

## KONFIGURACJA JEDNEJ SIECIÓWKI ZMOSTKOWANEJ z adresem statycznym

1. Ustawiamy w maszynie sieć jako zmostkowaną
2. Włączamy maszynę
2. Edytujemy plik yml z konfiguracją w katalogu /etc/netplan do postaci:

```
GNU nano 2.9.3 /etc/netplan/01-netcfg.yaml
# This file is generated from information provided by the datasource. Changes
# to it will not persist across an instance reboot. To disable cloud-init's
# network configuration capabilities, write a file
# /etc/cloud/cloud.cfg.d/99-disable-network-config.cfg with the following:
# network: {config: disabled}
network:
  version: 2
  renderer: networkd
  ethernets:
    enp0s3:
      dhcp4: false
      addresses: [192.168.1.200/24]
      gateway4: 192.168.1.1
      nameservers:
        addresses: [1.1.1.1,8.8.8.8]
```

Jak wyskakuje ostrzeżenie że plik jest edytowany to po prostu potwierdzamy. Wszystkie wcięcia robimy spacją!! Przy tabie wywali błąd.

3. Zapisujemy
4. Stosujemy zmiany przy użyciu **netplan apply**
5. Sprawdzamy, czy adres ustawił się poprawnie przy użyciu **ip a**

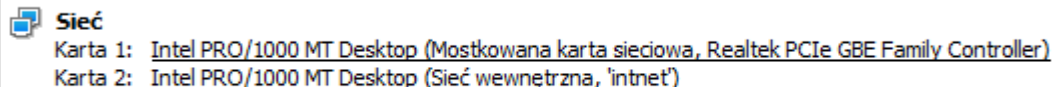
```
root@jkubuntu:/home/zsme# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:1f:bf:6c brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.200/24 brd 192.168.1.255 scope global enp0s3
        valid_lft forever preferred_lft forever
    inet6 fe80::a00:27ff:fe1f:bf6c/64 scope link
        valid_lft forever preferred_lft forever
root@jkubuntu:/home/zsme# _
```

6. Możemy sprawdzić pingiem czy mamy łączność ze światem:

```
root@jkubuntu:/home/zsme# ping onet.pl
PING onet.pl (213.180.141.140) 56(84) bytes of data:
64 bytes from sg1.any.onet.pl (213.180.141.140): icmp_seq=1 ttl=58 time=13.6 ms
64 bytes from sg1.any.onet.pl (213.180.141.140): icmp_seq=2 ttl=58 time=14.5 ms
64 bytes from sg1.any.onet.pl (213.180.141.140): icmp_seq=3 ttl=58 time=12.8 ms
64 bytes from sg1.any.onet.pl (213.180.141.140): icmp_seq=4 ttl=58 time=13.9 ms
64 bytes from sg1.any.onet.pl (213.180.141.140): icmp_seq=5 ttl=58 time=13.3 ms
64 bytes from sg1.any.onet.pl (213.180.141.140): icmp_seq=6 ttl=58 time=14.0 ms
64 bytes from sg1.any.onet.pl (213.180.141.140): icmp_seq=7 ttl=58 time=14.4 ms
^C
--- onet.pl ping statistics ---
7 packets transmitted, 7 received, 0% packet loss, time 6010ms
rtt min/avg/max/mdev = 12.876/13.850/14.593/0.573 ms
```

## KONFIGURACJA DWÓCH SIECIÓWEK jednej ZMOSTKOWANEJ z adresem statycznym i drugiej WEWNETRZNEJ z adresem statycznym

1. Ustawiamy w maszynie sieć jako zmostkowaną, drugą sieć jako wewnętrzną o nazwie intnet



Sieć  
Karta 1: Intel PRO/1000 MT Desktop (Mostkowana karta sieciowa, Realtek PCIe GBE Family Controller)  
Karta 2: Intel PRO/1000 MT Desktop (Sieć wewnętrzna, 'intnet')

2. Włączamy maszynę

3. Sprawdzamy nazwę naszej drugiej sieciówki przy użyciu `lshw -c network`

```
resources: irq:19 memory:f1200000-f121ffff ioport:d020(size=8)
*-network:1 DISABLED
description: Ethernet interface
product: 82540EM Gigabit Ethernet Controller
vendor: Intel Corporation
physical id: 8
bus info: pci@0000:00:08.0
logical name: enp0s8
version: 02
serial: 08:00:27:7b:3e:10
size: 1Gbit/s
capacity: 1Gbit/s
width: 32 bits
clock: 66MHz
capabilities: pm pci_x bus_master cap_list ethernet physical tp 10bt 10bt-fd 100bt 100bt-fd 100bt-fd autonegotiation
configuration: autonegotiation=on broadcast=yes driver=e1000 driverversion=7.3.21-k8-NAPI duplex=full latency=64 link=no mingnt=255 multicast=yes port=twisted pair speed=1Gbit/s
resources: irq:16 memory:f1820000-f183ffff ioport:d240(size=8)
root@jkubuntu:/home/zsme# _
```

U mnie to `enp0s8` (zero a nie duże o)

2. Edytujemy plik yaml z konfiguracją w katalogu /etc/netplan do postaci:

```
GNU nano 2.9.3 /etc/netplan/01-netcfg.yaml
# This file is generated from information provided by the datasource. Changes
# to it will not persist across an instance reboot. To disable cloud-init's
# network configuration capabilities, write a file
# /etc/cloud/cloud.cfg.d/99-disable-network-config.cfg with the following:
# network: {config: disabled}
network:
  version: 2
  renderer: networkd
  ethernets:
    enp0s3:
      dhcp4: false
      addresses: [192.168.1.200/24]
      gateway4: 192.168.1.1
      nameservers:
        addresses: [1.1.1.1,8.8.8.8]
    enp0s8:
      dhcp4: false
      addresses: [10.0.0.1/8]
```

Jak wyskakuje ostrzeżenie że plik jest edytowany to po prostu potwierdzamy. Wszystkie wcięcia robimy spacją!! Przy tabie wywali błąd. Jak gdzieś zabraknie jakiejś spacji też wywali błąd.

3. Zapisujemy

4. Stosujemy zmiany przy użyciu `netplan apply`

5. Sprawdzamy czy adres ustawił się poprawnie przy użyciu `ip a`

```
root@jkubuntu:/home/zsme# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:1f:bf:6c brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.200/24 brd 192.168.1.255 scope global enp0s3
        valid_lft forever preferred_lft forever
    inet6 fe80::a00:27ff:fe1f:bf6c/64 scope link
        valid_lft forever preferred_lft forever
3: enp0s8: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:7b:3e:10 brd ff:ff:ff:ff:ff:ff
    inet 10.0.0.1/8 brd 10.255.255.255 scope global enp0s8
        valid_lft forever preferred_lft forever
    inet6 fe80::a00:27ff:fe7b:3e10/64 scope link
        valid_lft forever preferred_lft forever
root@jkubuntu:/home/zsme# _
```

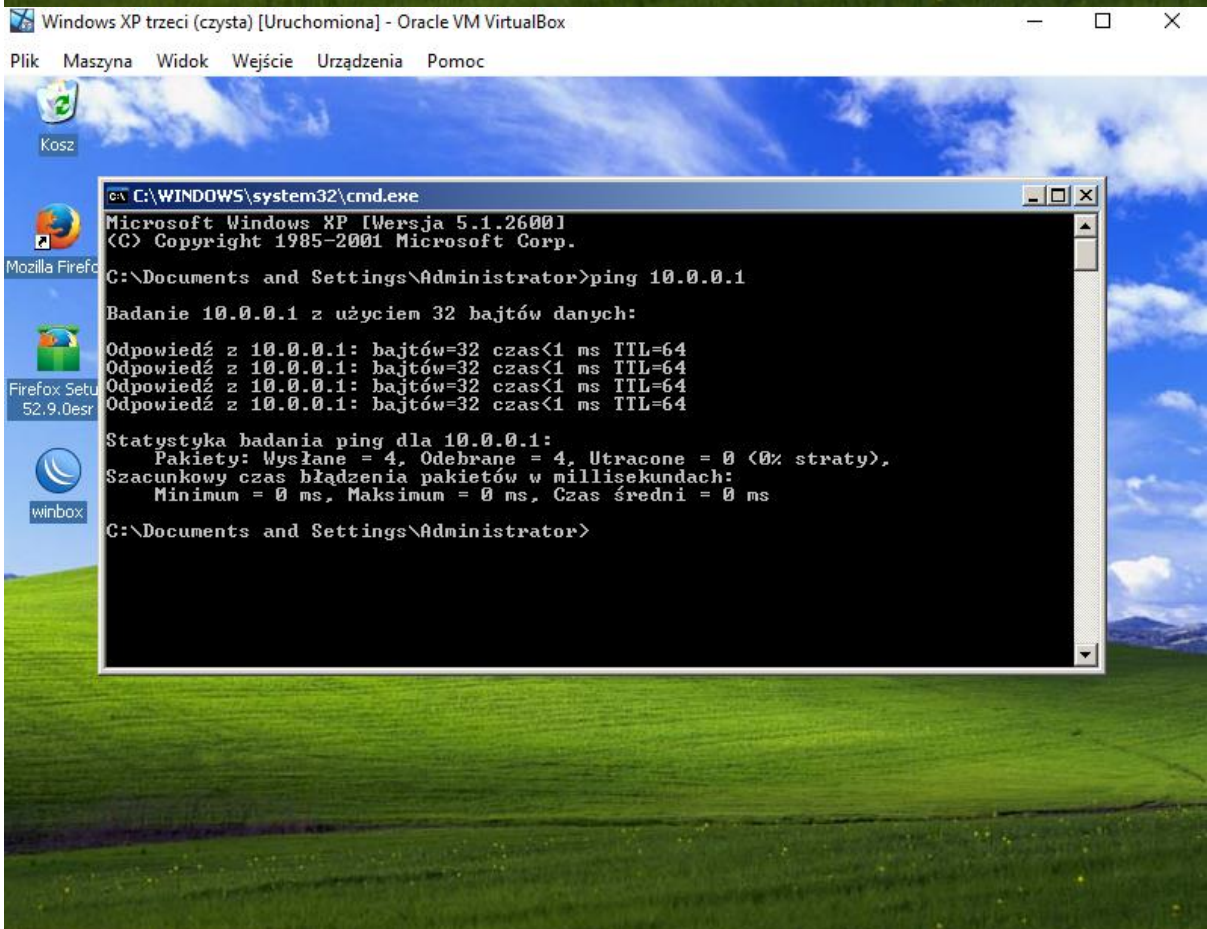
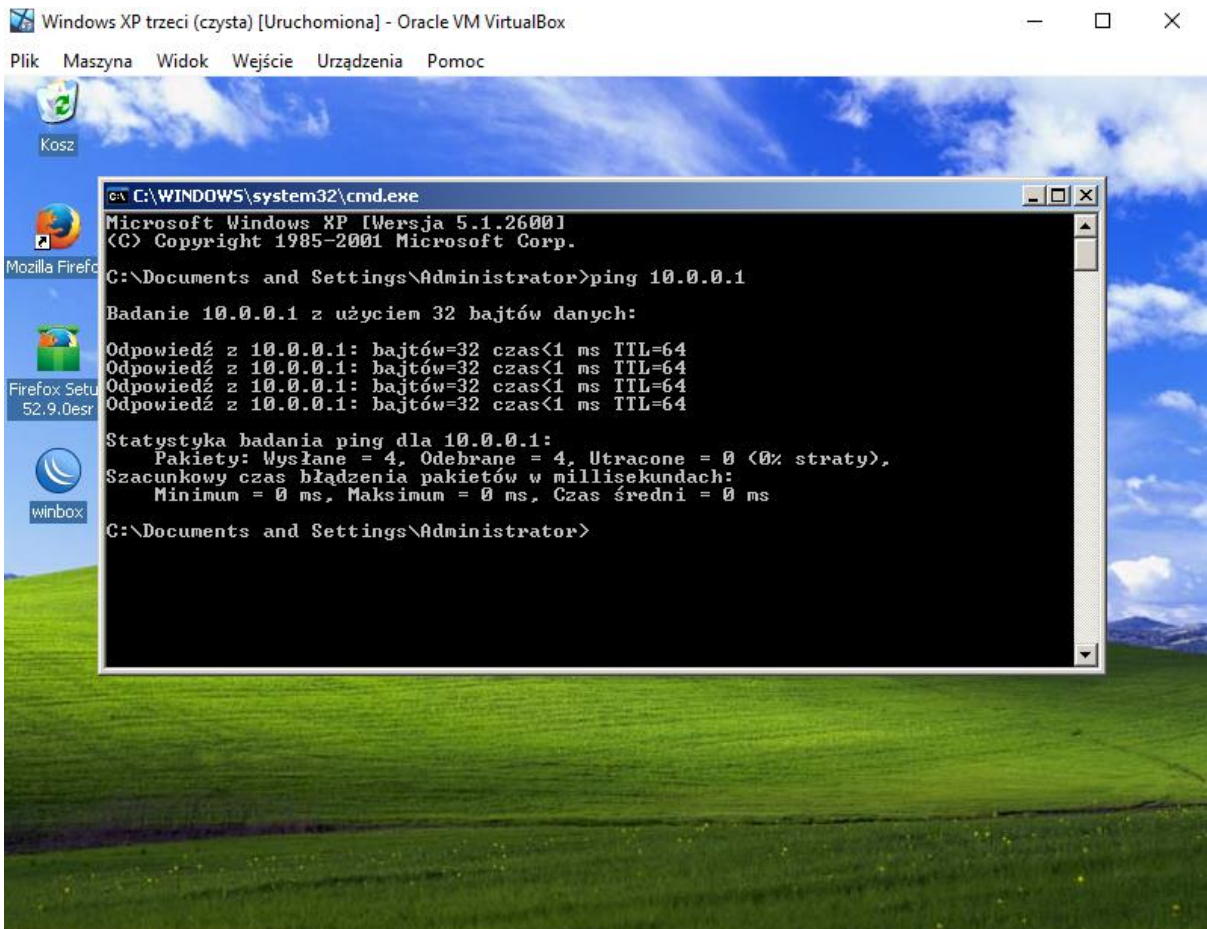
## 6. Możemy sprawdzić czy mamy łączność ze światem.

```
root@jkubuntu:/home/zsme# ping wp.pl
PING wp.pl (212.77.98.9) 56(84) bytes of data.
64 bytes from www.wp.pl (212.77.98.9): icmp_seq=1 ttl=53 time=23.5 ms
64 bytes from www.wp.pl (212.77.98.9): icmp_seq=2 ttl=53 time=23.2 ms
64 bytes from www.wp.pl (212.77.98.9): icmp_seq=3 ttl=53 time=23.0 ms
64 bytes from www.wp.pl (212.77.98.9): icmp_seq=4 ttl=53 time=23.0 ms
^C
--- wp.pl ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3005ms
rtt min/avg/max/mdev = 23.019/23.219/23.554/0.240 ms
root@jkubuntu:/home/zsme#
```

## 7. Uruchamiamy Windowsa XP z sieciówką wewnętrzną intent, ustawiamy mu adres 10.0.0.2 z maską 255.0.0.0 i sprawdzamy czy maszyny się widzą na sieciówkach wewnętrznych:

```
root@jkubuntu:/home/zsme# ping 10.0.0.2
PING 10.0.0.2 (10.0.0.2) 56(84) bytes of data.
64 bytes from 10.0.0.2: icmp_seq=1 ttl=128 time=0.619 ms
64 bytes from 10.0.0.2: icmp_seq=2 ttl=128 time=0.287 ms
64 bytes from 10.0.0.2: icmp_seq=3 ttl=128 time=0.288 ms
64 bytes from 10.0.0.2: icmp_seq=4 ttl=128 time=0.317 ms
^C
--- 10.0.0.2 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3071ms
rtt min/avg/max/mdev = 0.287/0.377/0.619/0.141 ms
root@jkubuntu:/home/zsme# _
```

I z XP:



## 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

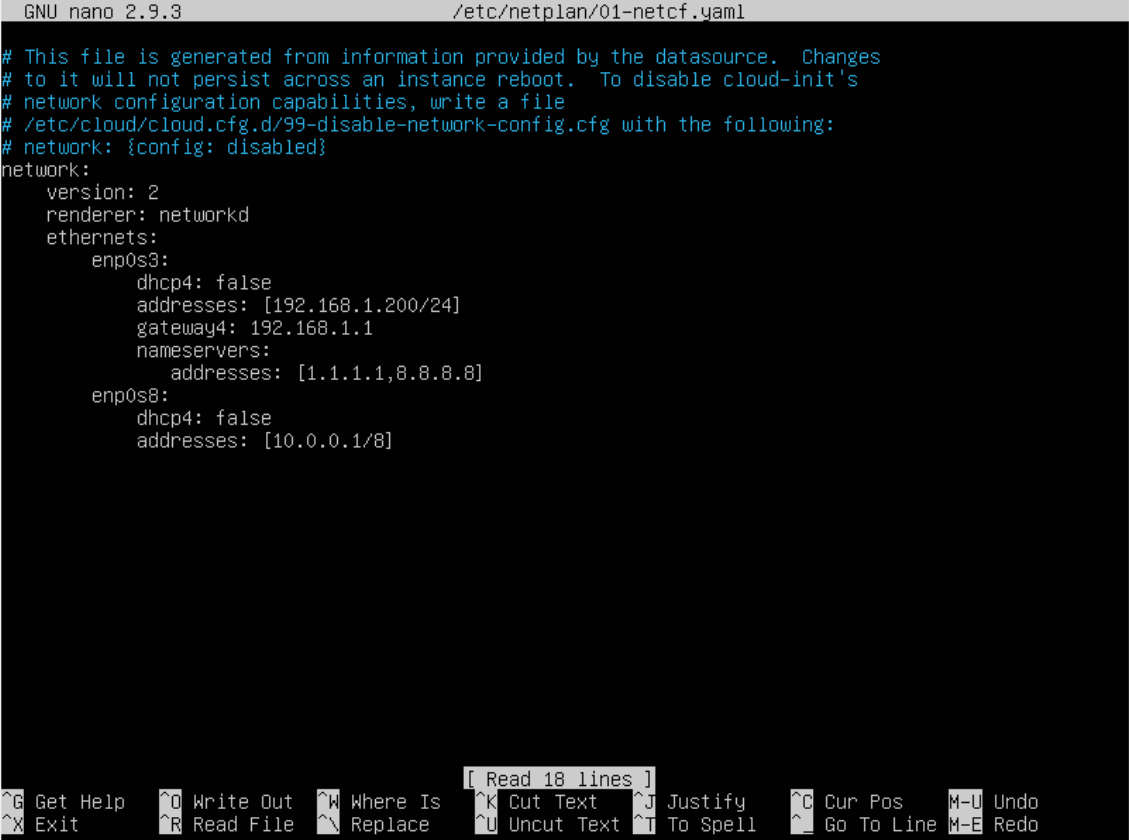
## 2. Konfigurujemy karty sieciowe na maszynie testowej z WinXP/Win7:

- \* 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:



```
GNU nano 2.9.3 /etc/netplan/01-netcf.yaml
# This file is generated from information provided by the datasource. Changes
# to it will not persist across an instance reboot. To disable cloud-init's
# network configuration capabilities, write a file
# /etc/cloud/cloud.cfg.d/99-disable-network-config.cfg with the following:
# network: {config: disabled}
network:
  version: 2
  renderer: networkd
  ethernets:
    enp0s3:
      dhcp4: false
      addresses: [192.168.1.200/24]
      gateway4: 192.168.1.1
      nameservers:
        addresses: [1.1.1.1,8.8.8.8]
    enp0s8:
      dhcp4: false
      addresses: [10.0.0.1/8]
```

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

#### 4. Aktualizujemy zawartość repozytoriów:

```
sudo apt update
```

#### 5. Instalujemy serwer FTP:

```
apt install vsftpd
```

jeżeli nie dało by się zainstalować bo wyskakuje taki błąd:

```
root@jkubuntu:/home/zsme# apt install vsftpd
E: Nie udało się uzyskać blokady /var/lib/dpkg/lock-frontent - open (11: Resource temporarily unavail
lable)
E: Unable to acquire the dpkg frontend lock (/var/lib/dpkg/lock-frontent), is another process using
it?
root@jkubuntu:/home/zsme#
```

Należy zrestartować Ubuntu przy użyciu polecenia:

```
reboot lub shutdown -r now
```

i ponownie spróbować zainstalować

```
root@jkubuntu:/home/zsme# apt install vsftpd
Czytanie list pakietów... Gotowe
Budowanie drzewa zależności
Odczyt informacji o stanie... Gotowe
The following additional packages will be installed:
  libwrap0 ssl-cert
Sugerowane pakiety:
  openssl-blacklist
Zostaną zainstalowane następujące NOWE pakiety:
  libwrap0 ssl-cert vsftpd
0 aktualizowanych, 3 nowo instalowanych, 0 usuwanych i 146 nieaktualizowanych.
Konieczne pobranie 179 kB archiwów.
Po tej operacji zostanie dodatkowo użyte 505 kB miejsca na dysku.
Kontynuować? [T/n] t
Pobieranie:1 http://pl.archive.ubuntu.com/ubuntu bionic/main amd64 libwrap0 amd64 7.6.q-27 [46,3 kB]
Pobieranie:2 http://pl.archive.ubuntu.com/ubuntu bionic/main amd64 ssl-cert all 1.0.39 [17,0 kB]
Pobieranie:3 http://pl.archive.ubuntu.com/ubuntu bionic/main amd64 vsftpd amd64 3.0.3-9build1 [115 k
B]
Pobrano 179 kB w 1s (326 kB/s)
Prekonfiguracja pakietów ...
Wybieranie wcześniej niewybranego pakietu libwrap0:amd64.
(Odczytywanie bazy danych ... 64157 plików i katalogów obecnie zainstalowanych.)
Przygotowywanie do rozpakowania pakietu ../libwrap0_7.6.q-27_amd64.deb ...
Rozpakowywanie pakietu libwrap0:amd64 (7.6.q-27) ...
Wybieranie wcześniej niewybranego pakietu ssl-cert.
Przygotowywanie do rozpakowania pakietu ../ssl-cert_1.0.39_all.deb ...
Rozpakowywanie pakietu ssl-cert (1.0.39) ...
-
Progress: [ 25%] [#####.....]
```

**6. Konfiguracja serwera ftp** znajduje się w pliku:

```
/etc/vsftpd/conf
```

Więc **robimy kopie zapasową**, w razie jakbyśmy coś namieszcili:

```
cp /etc/vsftpd.conf /etc/vsftpd.conf.bac
```

### **7. Konfiguracja dostępu autoryzowanego – wymagane będzie konto żeby się zalogować na serwerze**

Domyślnie serwer powinien od razu działać. Można się z nim spróbować połączyć.

Utwórzmy testowe konto użytkownika stefan z hasłem 123qwe

#### **adduser stefan**

```
root@jkubuntu:/home/zsme# adduser stefan
Adding user `stefan' ...
Adding new group `stefan' (1001) ...
Adding new user `stefan' (1001) with group `stefan' ...
Creating home directory `/home/stefan' ...
Copying files from `/etc/skel' ...
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
Changing the user information for stefan
Enter the new value, or press ENTER for the default
  Full Name []:
  Room Number []:
  Work Phone []:
  Home Phone []:
  Other []:
Is the information correct? [Y/n]
root@jkubuntu:/home/zsme#
```

Po zalogowaniu się na serwer na nasze testowe konto stefan domyślnym katalogiem będzie jego katalog domowy:

```
/home/stefan
```

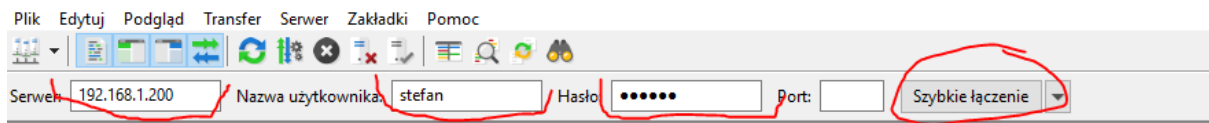
stwórzmy więc jakiś testowy plik, żeby było od razu widać, czy poprawnie się podłączyliśmy:

```
touch /home/stefan/plik1.txt
```

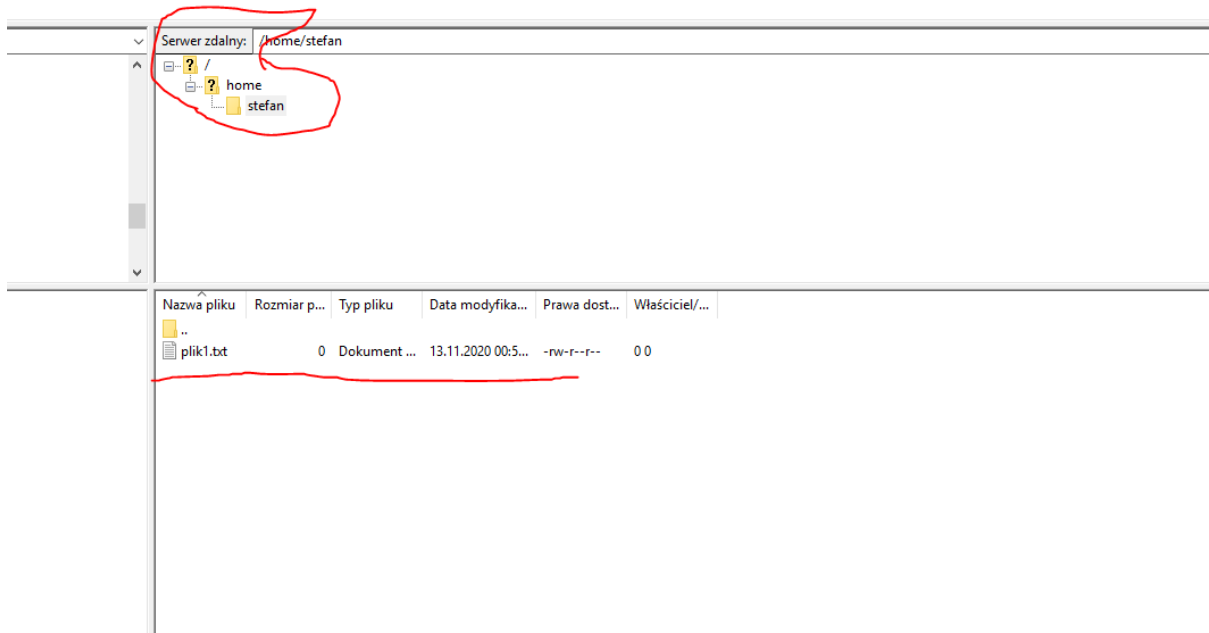
Zainstalujmy teraz klienta FTP - Filezilla:

[https://dl3.cdn.filezilla-project.org/client/FileZilla\\_3.51.0\\_win64-setup.exe?h=qjV41St8NdOGomVvgIO7eQ&x=1605228697](https://dl3.cdn.filezilla-project.org/client/FileZilla_3.51.0_win64-setup.exe?h=qjV41St8NdOGomVvgIO7eQ&x=1605228697)

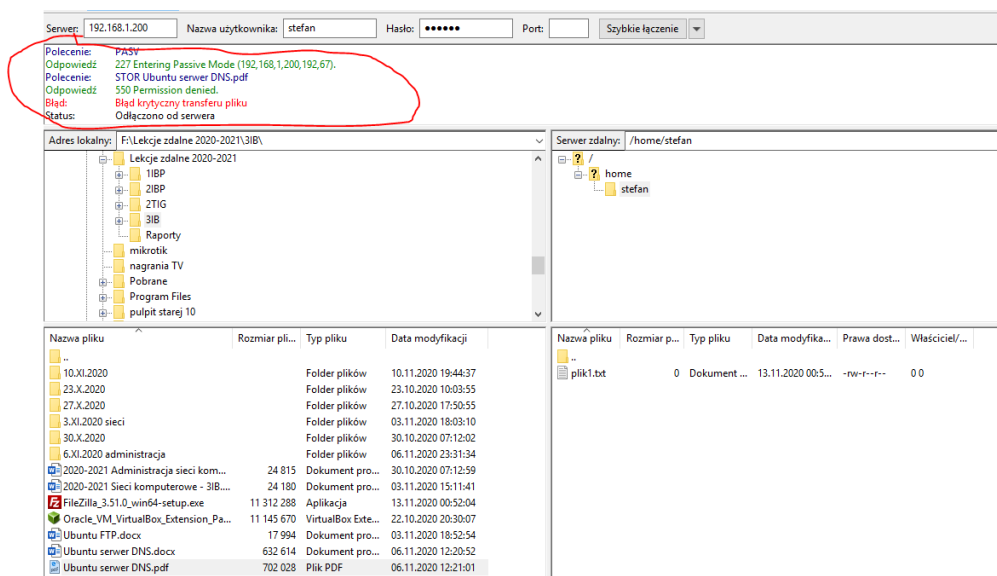
Możemy spróbować się podłączyć wpisując w odpowiednie miejsca adres IP serwera Ubuntu oraz login i hasło:



Jak widać, połączyliśmy się i jesteśmy w katalogu domowym użytkownika ( )



Domyślnie zapis na serwerze jest wyłączony więc przy próbie przesłania czegoś na serwer wyskoczy błąd (klikamy dwa razy w wybrany plik na lewym panelu, żeby go przesłać na serwer):



Żeby włączyć zapis szukamy w configu poniżej zmiennej i zdejmijmy komentarz (usuńmy hash na początku linijki)

```
GNU nano 2.9.3 /etc/vsftpd.conf Modified
# on the IPv6 "any" address (::) will accept connections from both IPv6
# and IPv4 clients. It is not necessary to listen on *both* IPv4 and IPv6
# sockets. If you want that (perhaps because you want to listen on specific
# addresses) then you must run two copies of vsftpd with two configuration
# files.
listen_ipv6=YES
#
# Allow anonymous FTP? (Disabled by default).
anonymous_enable=NO
#
# Uncomment this to allow local users to log in.
local_enable=YES
#
# Uncomment this to enable any form of FTP write command.
write_enable=YES
#
# Default umask for local users is 077. You may wish to change this to 022,
# if your users expect that (022 is used by most other ftpd's)
local_umask=022
#
# Uncomment this to allow the anonymous FTP user to upload files. This only
# has an effect if the above global write enable is activated. Also, you will
# obviously need to create a directory writable by the FTP user.
anon_upload_enable=YES
#
# Uncomment this if you want the anonymous FTP user to be able to create
# new directories.
anon_mkdir_write_enable=YES
#
# Activate directory messages - messages given to remote users when they
# go into a certain directory.
```

Zapiszmy plik, zrestartujmy usługę:

**service vsftpd restart**

Lub

**systemctl restart vsftpd**

I sprawdźmy status serwera:

**service vsftpd status**

```
root@jkubuntu:/home/zsme# service vsftpd restart
root@jkubuntu:/home/zsme# service vsftpd status
● vsftpd.service - vsftpd FTP server
   Loaded: loaded (/lib/systemd/system/vsftpd.service; enabled; vendor preset: enabled)
   Active: active (running) since Fri 2020-11-13 00:12:22 UTC; 9s ago
     Process: 1734 ExecStartPre=/bin/mkdir -p /var/run/vsftpd/empty (code=exited, status=0/SUCCESS)
    Main PID: 1745 (vsftpd)
      Tasks: 1 (limit: 1108)
   CGroup: /system.slice/vsftpd.service
           └─1745 /usr/sbin/vsftpd /etc/vsftpd.conf

lis 13 00:12:22 jkubuntu systemd[1]: Starting vsftpd FTP server...
lis 13 00:12:22 jkubuntu systemd[1]: Started vsftpd FTP server.
root@jkubuntu:/home/zsme# _
```

Jak widać serwer działa poprawnie możemy spróbować znowu przesłać plik:

Nazwa pliku	Rozmiar pliku	Typ pliku	Data modyfikacji	Nazwa pliku	Rozmiar p...	Typ pliku	Data modyfika...	Prawa dost...	Właściciel/...
..				..					
10.XI.2020		Folder plików	10.11.2020 19:44:37	plik1.txt	0	Dokument ...	13.11.2020 00:5...	-rw-r--r--	0 0
23.X.2020		Folder plików	23.10.2020 10:03:55	Ubuntu ...	702 028	Plik PDF	13.11.2020 01:1...	-rw-r-----	1001 1001
27.X.2020		Folder plików	27.10.2020 17:50:55						
3.XI.2020 sieci		Folder plików	03.11.2020 18:03:10						
30.X.2020		Folder plików	30.10.2020 07:12:02						
6.XI.2020 administracja		Folder plików	06.11.2020 23:31:34						
2020-2021 Administracja sieci kom...	24 815	Dokument pro...	30.10.2020 07:12:59						
2020-2021 Sieci komputerowe - 3IB...	24 180	Dokument pro...	03.11.2020 15:11:41						
FileZilla_3.51.0_win64-setup.exe	11 312 288	Aplikacja	13.11.2020 00:52:04						
Oracle_VM_VirtualBox_Extension_Pa...	11 145 670	VirtualBox Ext...	22.10.2020 20:30:07						
Ubuntu FTP.docx	17 994	Dokument pro...	03.11.2020 18:52:54						
Ubuntu serwer DNS.docx	632 614	Dokument pro...	06.11.2020 12:20:52						
Ubuntu serwer DNS.pdf	702 028	Plik PDF	06.11.2020 12:21:01						

Zaznaczono 1 plik. Całkowity rozmiar: 702 028 bajtów

2 pliki. Całkowity rozmiar: 702 028 bajtów

Plik lokalny/serwerowy	Kierunek	Zdalny plik	Rozmiar	Priorytet	Czas
stefan@192.168.1.200					
F:\Lekcje zdalne 2020-2021...	-->	/home/stefan/Ubuntu serwer DNS.pdf	702 028	Norma...	13.11.2020 0...

Pliki w kolejce    Nieudane transfery (1)    **Udane transfery (1)**

Plik udało się wgrać na serwer, możemy też zerknąć z poziomu serwera czy faktycznie taki plik się pojawił:

### ls -l /home/stefan

```
root@jkubuntu:/home/zsme# ls -l /home/stefan/
total 688
-rw-r--r-- 1 root root          0 lis 12 23:54 plik1.txt
-rw----- 1 stefan stefan 702028 lis 13 00:13 'Ubuntu serwer DNS.pdf'
root@jkubuntu:/home/zsme# _
```

## 7.Ustawianie banera powitalnego

ustawiamy poniższą zmienną:

```
# attack (DoS) via the command "SIZE /big/file" in ASCII mode. vsftpd
# predicted this attack and has always been safe, reporting the size of the
# raw file.
# ASCII mangling is a horrible feature of the protocol.
#ascii_upload_enable=YES
#ascii_download_enable=YES
#
# You may fully customise the login banner string:
ftpd_banner=*****Witam serdecznie na serwerze*****
#
# You may specify a file of disallowed anonymous e-mail addresses. Apparently
# useful for combatting certain DoS attacks.
#deny_email_enable=YES
# (default follows)
#banned_email_file=/etc/vsftpd.banned_emails
```

Restartujemy serwer, spróbujmy połączyć się z cmd żeby zobaczyć nasz baner (w cmd wpisujemy **ftp 192.168.1.200**):

```
Wiersz polecenia - ftp 192.168.1.200
Microsoft Windows [Version 10.0.15063]
(c) 2017 Microsoft Corporation. Wszelkie prawa zastrzeżone.

C:\Users\blackdog>ftp 192.168.1.200
Connected to 192.168.1.200.
220 *****Witam serdecznie na serwerze*****
200 Always in UTF8 mode.
User (192.168.1.200:(none)): stefan
331 Please specify the password.
Password:
230 Login successful.
ftp>
```

## 8. Blokada przechodzenia między katalogami:

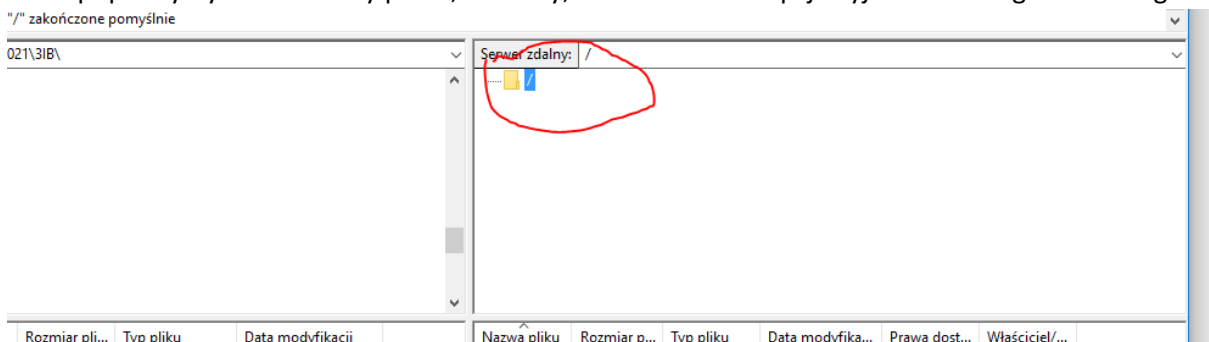
Po zalogowaniu każdy użytkownik może opuścić swój katalog domowy i przeglądać dowolne katalogi



Jeżeli chcielibyśmy to zablokować to powinniśmy ustawić poniższe zmienne (później oczywiście restart serwera, żeby zmiany zaczęły działać):

```
# You may specify an explicit list of local users to chroot() to their home
# directory. If chroot_local_user is YES, then this list becomes a list of
# users to NOT chroot().
# (Warning! chroot'ing can be very dangerous. If using chroot, make sure that
# the user does not have write access to the top level directory within the
# chroot)
chroot_local_user=YES
allow_writeable_chroot=YES_
#chroot_list_enable=YES
# (default follows)
#chroot_list_file=/etc/vsftpd.chroot_list
#
# You may activate the "-R" option to the builtin ls. This is disabled by
# default to avoid remote users being able to cause excessive I/O on large
# sites. However, some broken FTP clients such as "ncftp" and "mirror" assume
# the presence of the "-R" option, so there is a strong case for enabling it.
#ls_recurse_enable=YES
```

Jeżeli popatrzymy na środkowy panel, widzimy, że teraz nie ma opcji wyjścia z katalogu domowego:



Jeżeli chcielibyśmy, żeby jednak wybrane konta miały możliwość przeglądania katalogów innych niż swój domowy użyjemy tej zmiennej:

```
# users to NOT chroot().
# (Warning! chroot'ing can be very dangerous. If using chroot, make sure that
# the user does not have write access to the top level directory within the
# chroot)
chroot_local_user=YES
allow_writeable_chroot=YES
chroot_list_enable=YES
# (default follows)
chroot_list_file=/etc/vsftpd.chroot_list
#
```

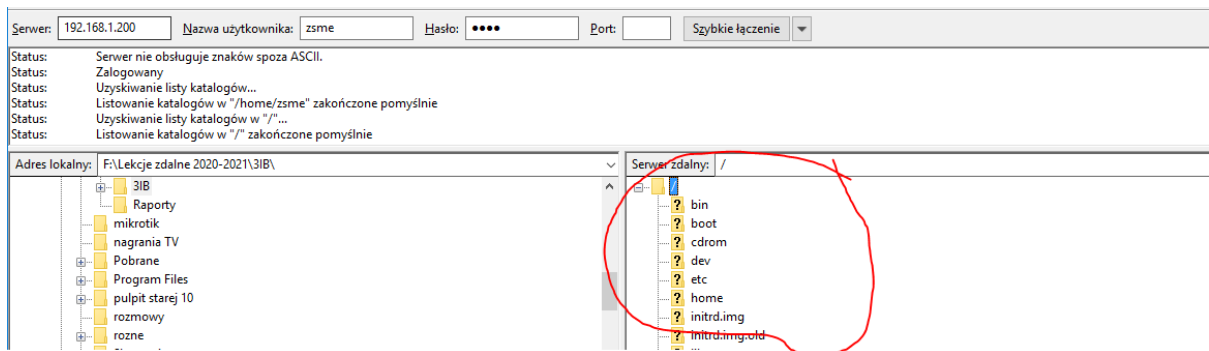
Musimy teraz stworzyć taki plik jak w zmiennej `chroot_list_file` i wpisać do niego konta użytkowników którzy mają mieć możliwość przeglądania katalogów:

**touch /etc/vsftpd.chroot\_list**

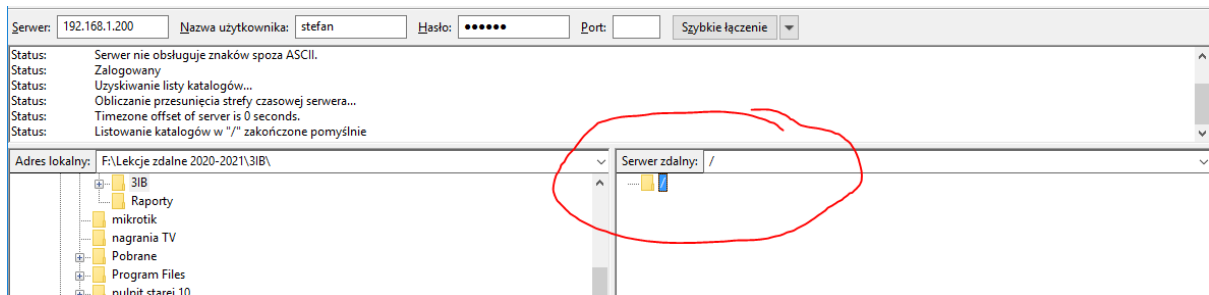
wpiszę tam użytkownika `zsmc`:

**echo zsmc > /etc/vsftpd.chroot\_list**

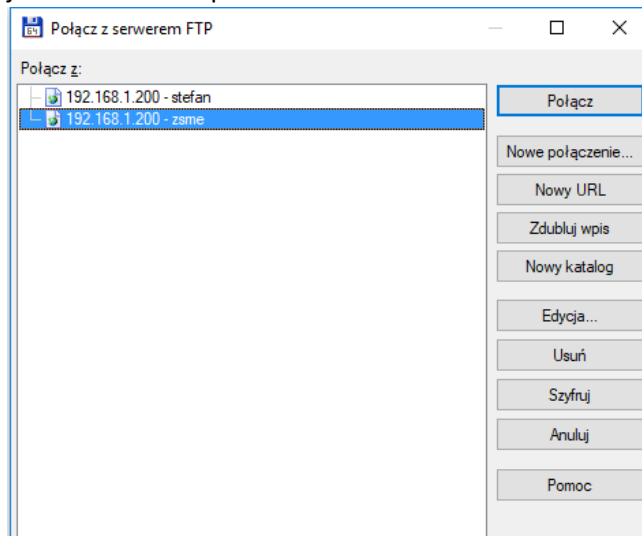
Restartujemy serwer i sprawdzamy czy działa:



a teraz `stefan`:



Wygodniej chyba było by testować to przy użyciu Total Commandera i zdefiniowaniu sobie połączeń już od razu na odpowiednie konta:



## 9. Konfiguracja połączenia anonimowego

Możemy chcieć, żeby była możliwość łączenia się z serwerem bez konieczności logowania się.

Ustawiamy zmienną:

```
# sockets. If you want that (perhaps because you want to listen on specific
# addresses) then you must run two copies of vsftpd with two configuration
# files.
listen_ipv6=YES
#
# Allow anonymous FTP? (Disabled by default).
anonymous_enable=YES_
#
# Uncomment this to allow local users to log in.
local_enable=YES
#
# Uncomment this to enable any form of FTP write command.
write_enable=YES
#
# Default umask for local users is 077. You may wish to change this to 022.
```

Domyślny katalog dla użytkowników anonimowych to

**/srv/ftp**

Utwórzmy tam 3 pliki:

```
root@jkubuntu:/home/zsme# touch /srv/ftp/plik1.txt
root@jkubuntu:/home/zsme# touch /srv/ftp/plik2.txt
root@jkubuntu:/home/zsme# touch /srv/ftp/plik3.txt
root@jkubuntu:/home/zsme#
```

Połączmy się z serwerem anonimowo (jako **login** podajemy **anonymous** a **hasło** to **adres email**, w Filezilli możemy po prostu nie wpisywać ani loginu ani hasła):

Server: 192.168.1.200    Nazwa użytkownika:    Hasło:    Port:    Szybkie łączenie

Status: Połączenie nawiązanie, oczekiwanie na wiadomość powitalną...  
Status: Niezabezpieczony serwer, nie obsługuje FTP przez TLS.  
Status: Serwer nie obsługuje znaków spoza ASCII.  
Status: Zalogowany  
Status: Uzyskiwanie listy katalogów...  
Status: Listowanie katalogów w "/" zakończone pomyślnie

Adres lokalny: F:\Lekcje zdalne 2020-2021\3IB\    Serwer zdalny: /

Nazwa pliku	Rozmiar pliku	Typ pliku	Data modyfikacji
..			
10.XI.2020		Folder plików	10.11.2020 19:44:37
23.XI.2020		Folder plików	23.10.2020 10:03:55
27.XI.2020		Folder plików	27.10.2020 17:50:55
3.XI.2020 sieci		Folder plików	03.11.2020 18:03:10
30.XI.2020		Folder plików	30.10.2020 07:12:02

Nazwa pliku	Rozmiar p...	Typ pliku	Data modyfika...	Prawa dost...	Właściciel/...
..					
plik1.txt	0	Dokument ...	13.11.2020 01:5...	-rw-r--r--	0 0
plik2.txt	0	Dokument ...	13.11.2020 01:5...	-rw-r--r--	0 0
plik3.txt	0	Dokument ...	13.11.2020 01:5...	-rw-r--r--	0 0

Możemy też ustawić, żeby serwer nie pytał użytkowników anonimowych o hasło:

```
GNU nano 2.9.3    /etc/vsftpd.conf    Modified
# sockets. If you want that (perhaps because you want to listen on specific
# addresses) then you must run two copies of vsftpd with two configuration
# files.
listen_ipv6=YES
#
# Allow anonymous FTP? (Disabled by default).
anonymous_enable=YES
no_anon_password=YES
#
```

Przetestujmy to przy użyciu cmd:

```
C:\Users\blackdog>ftp 192.168.1.200
Connected to 192.168.1.200.
220 *****Witam serdecznie na serwerze*****
200 Always in UTF8 mode.
User (192.168.1.200:(none)): anonymous
230 Login successful.
ftp>
```

Raczej logicznym działaniem jest, że domyślnie użytkownicy anonimowi nie mają prawa do zapisu. Jeżeli chcemy aby użytkownicy anonimowi mogli zapisywać pliku, powinniśmy ustawić zmienne:

```
# Uncomment this to allow the anonymous FTP user to upload files. This only
# has an effect if the above global write enable is activated. Also, you will
# obviously need to create a directory writable by the FTP user.
anon_upload_enable=YES
#
# Uncomment this if you want the anonymous FTP user to be able to create
# new directories.
anon_mkdir_write_enable=YES
#
```

W opisach zmiennych widać, co dana zmienna oznacza.

Dopiszmy jeszcze takie zmienne:

```
# Uncomment this if you want the anonymous FTP user to be able to create
# new directories.
anon_mkdir_write_enable=YES
ftp_username=stefan
anon_root=/home/ftp
#
```

**ftp\_username=stefan** – ustawiamy jako kto będą zapisywać anonimowi użytkownicy, jak nie ustawimy tej zmiennej, to będzie to użytkownik o nazwie ftp

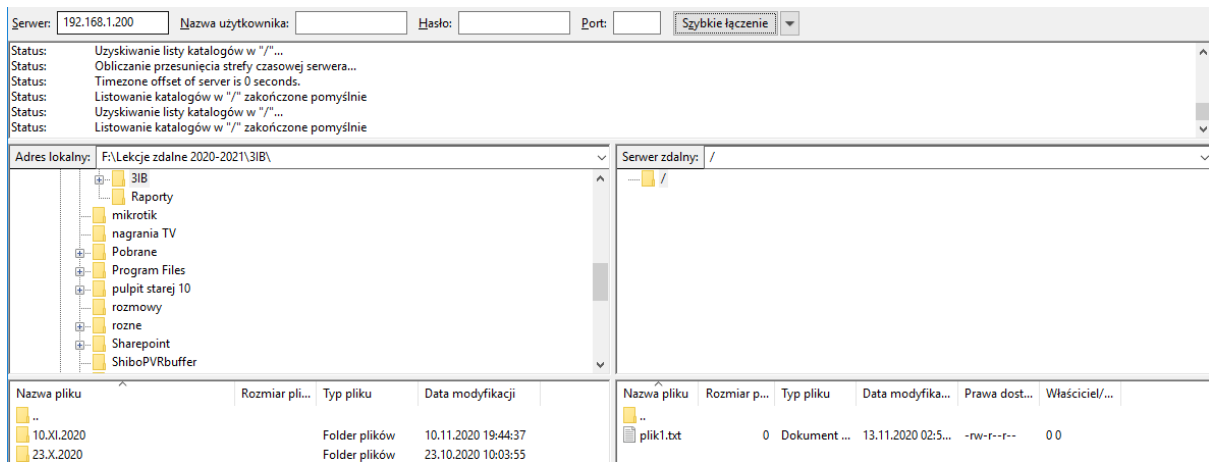
**anon\_root=/home/ftp** – katalog domyślny dla użytkowników anonimowych, jak nie ustawimy tej zmiennej, to będzie to katalog /srv/ftp

W naszym przypadku ustawiłem, że użytkownicy anonimowi będą zapisywać jako konto stefan a katalog domyślny to /home/ftp

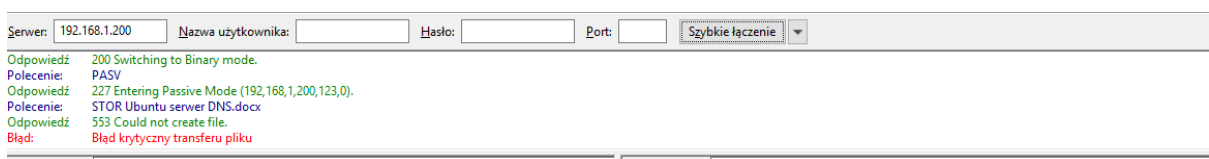
Utwórzmy więc taki katalog i jakiś testowy plik, żeby było widać, czy działa poprawnie:

```
root@jkubuntu:/home/zsme# cd /home
root@jkubuntu:/home# mkdir ftp
root@jkubuntu:/home# ls -l
total 12
drwxr-xr-x 2 root root 4096 lis 13 01:57 ftp
drwxr-xr-x 2 stefan stefan 4096 lis 13 00:13 stefan
drwxr-xr-x 5 zsme zsme 4096 maj 19 13:11 zsme
root@jkubuntu:/home# touch /home/ftp/plik1.txt
root@jkubuntu:/home#
```

Podłączmy się i spróbujmy coś zapisać:



Przy próbie zapisu otrzymamy błąd, bo nie mamy uprawnień do zapisu w katalogu ftp:



Musimy zrobić podkatalog w naszym katalogu /home/ftp

### mkdir /home/ftp/public

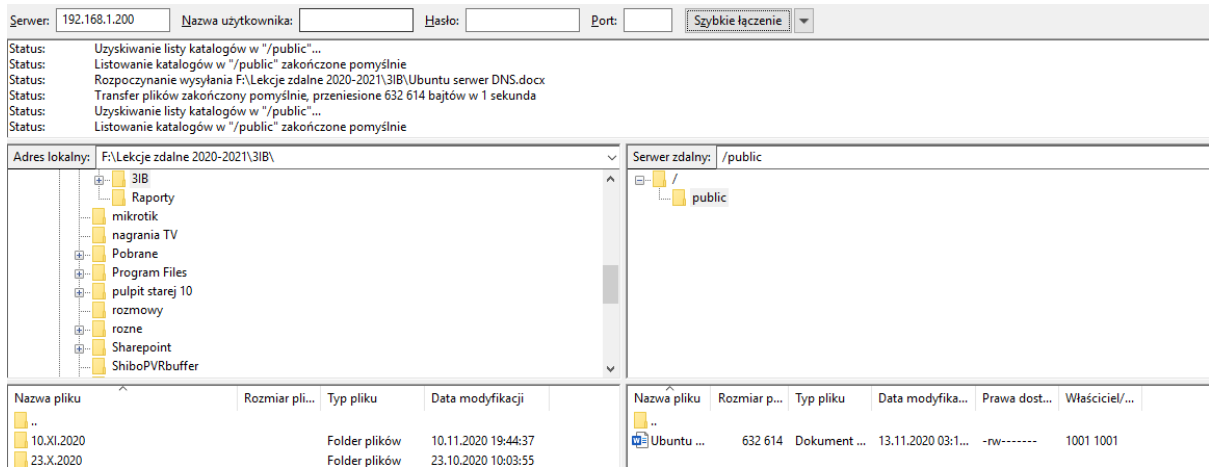
```
root@jkubuntu:/home/ftp# mkdir public
root@jkubuntu:/home/ftp# ls -l
total 4
-rw-r--r-- 1 root root    0 lis 13 01:59 plik1.txt
drwxr-xr-x 2 root root 4096 lis 13 02:07 public
```

i jako że ustawiliśmy że użytkownicy anonimowi zapisują z konta stefan, to np. zmienić mu właściciela na stefan:

### chown stefan /home/ftp/public

```
root@jkubuntu:/home/ftp# chown stefan /home/ftp/public
root@jkubuntu:/home/ftp# ls -l
total 4
-rw-r--r-- 1 root root    0 lis 13 01:59 plik1.txt
drwxr-xr-x 2 stefan root 4096 lis 13 02:07 public
root@jkubuntu:/home/ftp#
```

Jak widać, stefan jako właściciel ma prawa zapisu, więc logując się na użytkownika anonimowego powinniśmy być w stanie zapisywać w utworzonym przez nas podkatalogu public.



Zerknijmy czy faktycznie plik został utworzony z właścicielem stefan (tak ustawiliśmy w konfiguracji użytkowników anonimowych w naszym serwerze ftp):

```
root@jkubuntu:/home/ftp# cd public
root@jkubuntu:/home/ftp/public# ls -l
total 620
-rw----- 1 stefan stefan 632614 lis 13 02:12 'Ubuntu serwer DNS.docx'
root@jkubuntu:/home/ftp/public# _
```

Oczywiście możemy też zamiast zmieniać właściciela zmienić prawa dostępu do katalogu /home/ftp/public:

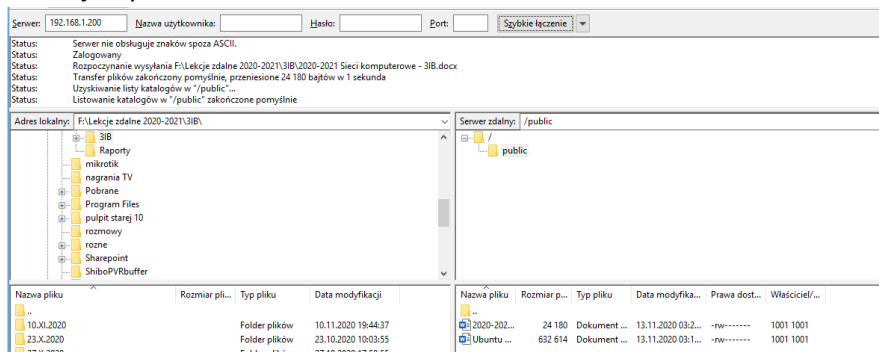
```
root@jkubuntu:/home/ftp# chown root public
root@jkubuntu:/home/ftp# ls -l
total 4
-rw-r--r-- 1 root root 0 lis 13 01:59 plik1.txt
drwxr-xr-x 2 root root 4096 lis 13 02:12 public
root@jkubuntu:/home/ftp#
```

Ustawiłem z powrotem że właścicielem pliku jest root

Teraz zmienię uprawnienia na 777 i też powinniśmy mieć możliwość uploadu:

```
root@jkubuntu:/home/ftp# chmod 777 public
root@jkubuntu:/home/ftp# ls -l
total 4
-rw-r--r-- 1 root root 0 lis 13 01:59 plik1.txt
drwxrwxrwx 2 root root 4096 lis 13 02:12 public
root@jkubuntu:/home/ftp#
```

I testujemy:



Jak widać też udało się przesłać pliki.

Czego nie możemy zrobić ? **Nie możemy** tak po prostu **ustawić praw zapisu dla całego katalogu domyślnego** z którym łączą się użytkownicy anonimowi (w naszym przypadku to **/home/ftp**) Musimy **stworzyć** w nim **podkatalog** i dopiero w nim ustawiać prawa zapisu:

W momencie gdy spróbujemy ustawić np. prawa 777 dla całego katalogu */home/ftp (przed zmianą jest 755)*:

```
root@jkubuntu:/home# ls -l
total 12
drwxr-xr-x 3 root  root  4096 lis 13 02:07 ftp
drwxr-xr-x 2 stefan stefan 4096 lis 13 00:13 stefan
drwxr-xr-x 5 zsme  zsme  4096 maj 19 13:11 zsme
root@jkubuntu:/home# _
```

Zmieniamy prawa:

```
root@jkubuntu:/home# chmod 777 ftp
root@jkubuntu:/home# ls -l
total 12
drwxrwxrwx 3 root  root  4096 lis 13 02:07 ftp
drwxr-xr-x 2 stefan stefan 4096 lis 13 00:13 stefan
drwxr-xr-x 5 zsme  zsme  4096 maj 19 13:11 zsme
root@jkubuntu:/home# _
```

Przy próbie połączenia się jako użytkownik zalogowany (*użyj cmd z windowsa*) otrzymuje komunikat:

```
C:\Users\blackdog>ftp 192.168.1.200
Connected to 192.168.1.200.
220 *****Witam serdecznie na serwerze*****
200 Always in UTF8 mode.
User (192.168.1.200:(none)): anonymous
500 OOPS: vsftpd: refusing to run with writable root inside chroot()
500 OOPS: priv_sock_get_cmd
Connection closed by remote host.

C:\Users\blackdog>
```

Ustawmy więc z powrotem prawa dostępu do */home/ftp* na 755:

```
root@jkubuntu:/home# chmod 755 ftp
root@jkubuntu:/home# ls -l
total 12
drwxr-xr-x 3 root  root  4096 lis 13 02:07 ftp
drwxr-xr-x 2 stefan stefan 4096 lis 13 00:13 stefan
drwxr-xr-x 5 zsme  zsme  4096 maj 19 13:11 zsme
root@jkubuntu:/home#
```

sprawdźmy czy serwer nas wpuści:

```
C:\Users\blackdog>ftp 192.168.1.200
Connected to 192.168.1.200.
220 *****Witam serdecznie na serwerze*****
200 Always in UTF8 mode.
User (192.168.1.200:(none)): anonymous
230 Login successful.
ftp>
```

Zapisujemy zawsze w PODKATALOGU !

Nieważne, czy ustawimy te zmienne w konfigu naszego serwera ftp:

```
ftp_username=stefan
anon_root=/home/ftp
```

(bez nich użytkownik anonimowy uploaduje pliki jako użytkownik o nazwie ftp  
a katalog domyślny z którym się łączy to /srv/ftp)

**TO ZAWSZE DO ZAPISU MUSIMY STWORZYĆ PODKATALOG I DOPIERO W NIM USTAWIĆ PRAWA  
ZAPISU czyli to sławne 777 czy też zmiana właściciela !!!!!!!!!!!**

## 10. Zaawansowane ustawienia serwera

Jak ktoś chce się podoktoryzować, lub po prostu dowiedzieć się więcej na temat konfiguracji serwera vsftpd, to bardzo dobry opis po polsku zmiennych konfiguracyjnych jest tutaj:

<https://morfikov.github.io/post/konfiguracja-vsftpd-w-debianie/>

Jak są jakieś literówki, to zgłaszajcie 😊

## 1. Przygotowanie maszyny wirtualnej z Ubuntu Server 18.04

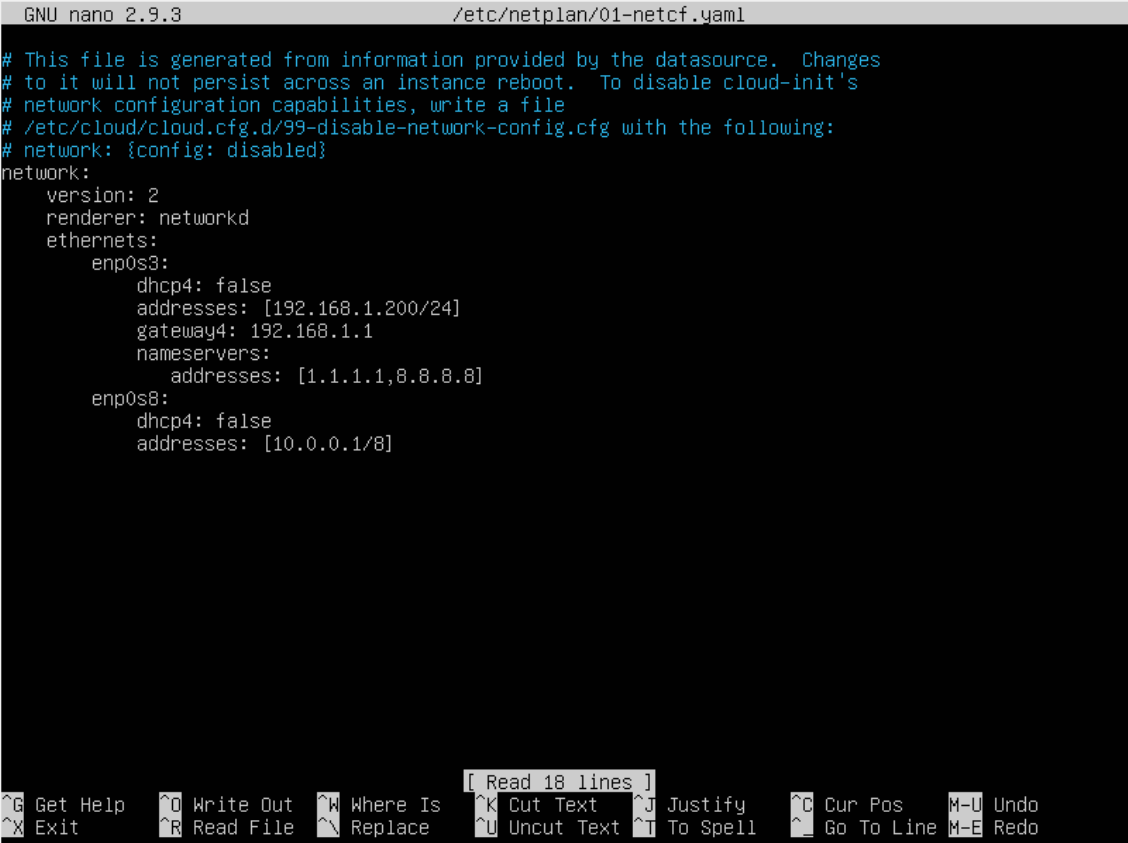
- \* pierwsza karta sieciowa – zmostkowana
- \* druga karta sieciowa – wewnętrzna intnet

jeżeli posiadamy, możemy przywrócić odpowiednią migawkę z kartami już skonfigurowanymi

## 2. Uruchamiamy maszynę i sprawdzamy adres IP na zmostkowanej sieciówce

```
nano /etc/netplan/01-netcf.yaml
```

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



```
GNU nano 2.9.3 /etc/netplan/01-netcf.yaml
# This file is generated from information provided by the datasource. Changes
# to it will not persist across an instance reboot. To disable cloud-init's
# network configuration capabilities, write a file
# /etc/cloud/cloud.cfg.d/99-disable-network-config.cfg with the following:
# network: {config: disabled}
network:
  version: 2
  renderer: networkd
  ethernets:
    enp0s3:
      dhcp4: false
      addresses: [192.168.1.200/24]
      gateway4: 192.168.1.1
      nameservers:
        addresses: [1.1.1.1,8.8.8.8]
    enp0s8:
      dhcp4: false
      addresses: [10.0.0.1/8]

[ Read 18 lines ]
^G Get Help  ^O Write Out  ^W Where Is  ^K Cut Text   ^J Justify    ^C Cur Pos   M-U Undo
^X Exit      ^R Read File  ^_ Replace   ^U Uncut Text ^T To Spell   ^_ Go To Line M-E Redo
```

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

### 3. Aktualizujemy zawartość repozytoriów:

```
sudo apt update
```

### 4. Instalujemy serwer SAMBA:

```
sudo apt install samba
```

Folder gdzie serwer WWW będzie trzymał swoje pliki konfiguracyjne to:

**/etc/samba**

```
root@jkubuntu:/etc/samba# ls -l
total 32
-rw-r--r-- 1 root root 8 paź 16 10:50 gdbcommands
-rw-r--r-- 1 root root 9937 gru 1 16:32 smb.conf
-rw-r--r-- 1 root root 9542 gru 1 16:12 smb.conf.bac
drwxr-xr-x 2 root root 4096 paź 16 10:50 t1s
root@jkubuntu:/etc/samba# _
```

**smb.conf** – główny plik konfiguracyjny

### 5. Konfiguracja ustawień ogólnych w sekcji [global]:

#### Interfejs na którym będzie działał serwer SAMBA

Domyślnie serwer samby pracuje na wszystkich kartach sieciowych. Jeżeli chcielibyśmy by usługa nasłuchiwała na konkretnym interfejsie, w sekcji [global] jest od tego odpowiednia sekcja (zwróćcie uwagę, że zmienne służące do ustawienia tego są za komentowane – wszystko w pliku co się zaczyna od # albo od ; to komentarz):

```
#### Networking ####

# The specific set of interfaces / networks to bind to
# This can be either the interface name or an IP address/netmask;
# interface names are normally preferred
; interfaces = 127.0.0.0/8 eth0

# Only bind to the named interfaces and/or networks; you must use the
# 'interfaces' option above to use this.
# It is recommended that you enable this feature if your Samba machine is
# not protected by a firewall or is a firewall itself. However, this
# option cannot handle dynamic or non-broadcast interfaces correctly.
; bind interfaces only = yes
```

Sprawdźmy nazwę naszego interfejsu sieciowego (zmostkowanego, przez który mamy internet, bo jego będziemy używać do testów)

## ip a

```
root@jkubuntu:/etc/samba# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:1f:bf:6c brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.200/24 brd 192.168.1.255 scope global enp0s3
        valid_lft forever preferred_lft forever
    inet6 fe80::a00:27ff:fe1f:bf6c/64 scope link
        valid_lft forever preferred_lft forever
3: enp0s8: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:45:89:bd brd ff:ff:ff:ff:ff:ff
    inet 10.0.0.1/8 brd 10.255.255.255 scope global enp0s8
        valid_lft forever preferred_lft forever
    inet6 fe80::a00:27ff:fe45:89bd/64 scope link
        valid_lft forever preferred_lft forever
```

U mnie to interfejs o nazwie enp0s3, więc ustawiam odpowiednio zmienne w pliku konfiguracyjnym:

```
#### Networking ####

# The specific set of interfaces / networks to bind to
# This can be either the interface name or an IP address/netmask;
# interface names are normally preferred
interfaces = enp0s3

# Only bind to the named interfaces and/or networks; you must use the
# 'interfaces' option above to use this.
# It is recommended that you enable this feature if your Samba machine is
# not protected by a firewall or is a firewall itself. However, this
# option cannot handle dynamic or non-broadcast interfaces correctly.
bind interfaces only = yes
```

Po zapisaniu pliku, oczywiście musimy zrestartować usługę – w tym przypadku mamy dwa procesy:

```
service smbd restart
service nmbd restart
```

**Ustawmy jeszcze nazwę serwera, grupę roboczą, oraz konto wykorzystywane do dostępu do udziałów bez autoryzacji:**

```
##### Global Settings #####

[global]

## Browsing/Identification ##
ntlm auth = true
# Change this to the workgroup/NT-domain name your Samba server will part of
workgroup = WORKGROUP
netbios name = Inazwisko
# server string is the equivalent of the NT Description field
server string = %h server

#konto wykorzystywane do dostępu dla gości:
guest account = inazwisko
# Windows Internet Name Service Support - Post test
```

**ntlm auth = true** - zmienna dopisana, bo Win 10 ma problemy z podłączaniem się do Samby, to pomaga

**workgroup = WORKGROUP** – nazwa grupy roboczej, dajemy taką jaką mamy w systemie, jeżeli chcemy się podłączać z Windows

**netbios name = inazwisko** – nazwa pod jaką wyświetli się nasz serwer, oczywiście w ćwiczeniu wpisujemy swoje dane

**guest account = inazwisko** – jeżeli zrobimy udział dla gości bez logowania, to możemy zmienić domyślne konto, z którego będą zapisywać pliki na naszym serwerze – jeżeli nie ustawimy tej zmiennej to będzie to konto **nobody** (o tyle jest to istotne, że trzeba później, robiąc przydział, pamiętać o uprawnieniach katalogu dla odpowiedniego konta, z którego będą zapisywać goście)

w naszym ćwiczeniu użyjemy dla gości konta inazwisko (oczywiście wpiszcie swoje) więc trzeba pamiętać, żeby stworzyć sobie takiego użytkownika:

**adduser inazwisko**

## 5. Tworzenie udziałów anonimowych – bez logowania:

Stwórzmy najpierw katalog który udostępnimy bez logowania:

```
mkdir -p /home/samba/uczniowie
```

teraz pamiętajmy o ustawieniu praw zapisu – w naszym przypadku możemy zrobić

```
chown inazwisko /home/samba/uczniowie
```

lub po prostu polecieć po łatwości i ustawić dla tego katalogu prawa 777

```
chmod 777 /home/samba/uczniowie
```

```
root@jkubuntu:/home# ls -l /home/samba/
total 4
drwxr-xr-x 2 inazwisko root 4096 gru  4 01:51 uczniowie
root@jkubuntu:/home#
```

Teraz odtworzymy nasz plik z konfiguracją samby /etc/samba/smb.conf i na końcu pliku dodajmy taki wpis:

```
# Uncomment to allow remote administration of Windows print drivers.
# You may need to replace 'lpadmin' with the name of the group your
# admin users are members of.
# Please note that you also need to set appropriate Unix permissions
# to the drivers directory for these users to have write rights in it
; write list = root, @lpadmin

[uczniowie]
    comment = Udzial sieciowy przeznaczony dla uczniow
    path = /home/samba/uczniowie
    guest ok = yes
    browseable = yes
    writable = yes
```

Nazwa udziału to uczniowie, ścieżka która udostępniamy, to przygotowany przez nas wcześniej katalog, zmienne:

**guest ok = yes** – dostęp bez konieczności autoryzacji hasłem

**browseable = yes** – nasz udział będzie wyświetlał się na liście udziałów po połączeniu z serwerem, jeżeli tego nie ustawimy, to żeby dostać się do naszego udziału, będziemy musieli podać dokładną ścieżkę, czyli no.: //192.168.1.200/uczniowie

**writable = yes** – nie trudno się domyślić, włączamy w naszym udziale możliwość zapisu

**Pamiętajmy żeby po stworzeniu przydziału zrestartować serwer!**

## 6. Tworzenie udziałów wymagających zalogowania:

Stwórzmy najpierw katalog który udostępniemy dla użytkowników:

```
mkdir -p /home/samba/dyrekcja
```

teraz pamiętajmy o ustawieniu praw zapisu – w naszym przypadku będziemy udostępniać katalog użytkownikowi zsme, więc mogą go ustawić jako właściciela

```
chown zsme /home/samba/dyrekcja
```

lub po prostu polecieć po łatwości i ustawić dla tego katalogu prawa 777

```
chmod 777 /home/samba/dyrekcja
```

wtedy nie ważne komu go udostępnimy, powinien mieć prawa zapisu:

```
root@jkubuntu:/home# ls -l /home/samba/
total 8
drwxrwxrwx 2 zsme      root 4096 gru  4 02:10 dyrekcja
drwxrwxrwx 2 inazwisko root 4096 gru  4 01:51 uczniowie
root@jkubuntu:/home#
```

teraz otworzymy plik konfiguracyjny samby i dodajmy na końcu nasz przydział:

```
[dyrekcja]
comment = Udział dla dyrektora, wymagane logowanie
path = /home/samba/dyrekcja/
browseable = yes
guest ok = no
writable = yes
valid users = zsme
```

pojawiła nam się nowa zmienna:

**valid users = zsme** – użytkownikiem uprawnionym dla tego udziału jest użytkownik zsme, jeżeli mamy więcej użytkowników, to oddzielamy ich nazwy przecinkiem

**W ramach ćwiczeń przygotujcie kolejny przydział:  
użytkownik : inazwisko  
wymaga logowania  
ścieżka: /home/samba/inazwisko**

**Pamiętajmy żeby po stworzeniu przydziału zrestartować serwer!**

## 6. Tworzenie udziałów dla grup użytkowników:

Gdy mamy większą ilość osób którym byśmy chcieli przypisać dany udział, wygodniej będzie to zrobić dla grupy.

**Stwórzmy grupę nauczyciele:**

```
sudo groupadd nauczyciele
```

**stwórzmy konta użytkowników stefan i andrzej:**

```
adduser stefan
adduser andrzej
```

**dodajmy do naszej grupy użytkowników : zsme, stefan, andrzej**

```
usermod -a -G nauczyciele zsme
```

(oczywiście powtórzymy to dla wszystkich kont które chcemy dodać do grupy nauczyciele)

Po wszystkim sprawdzimy, czy faktycznie konta które chcemy, są w odpowiedniej grupie

```
nano /etc/group
```

```
stefan:x:1001:  
dyrekcja:x:1002:zsme,stefan  
andrzej:x:1003:  
inazwisko:x:1004:  
nauczyciele:x:1005:zsme,stefan,andrzej
```

**Stwórzmy katalog dla naszego udziału:**

```
mkdir -p /home/samba/nauczyciele
```

**nie zapomnijmy ustawić praw dostępu - ustawmy dla katalogu nauczyciele prawa 777**

```
drwxrwxrwx 2 zsme      root  4096 gru  4 02:10 dyrekcja  
drwxrwxrwx 2 root      root  4096 gru  4 02:40 nauczyciele  
drwxrwxrwx 2 inazwisko root  4096 gru  4 01:51 uczniowie  
root@ikubuntu:/home/samba#
```

teraz otworzymy plik konfiguracyjny samby i dodajmy na końcu nasz przydział dla grupy nauczyciele:

```
[nauczyciele]  
comment = Udział dla grupy nauczyciele, wymagane logowanie  
path = /home/samba/nauczyciele/  
browseable = yes  
guest ok = no  
writable = yes  
valid users = @nauczyciele
```

**Pamiętajmy żeby po stworzeniu przydziału zrestartować serwer!**

## 7. Aktywacja kont użytkowników w SAMBIE:

Z zasobów mogą korzystać zalogowani użytkownicy (w przypadku konfiguracji security = user). Serwer samba wymaga utworzenie użytkowników zasobów (takich samych jak w systemie linux). Konto istniejącego użytkownika systemowego dodaje się do bazy użytkowników serwera samby za pomocą polecenia: **smbpasswd**.

```
CodeHL 1.3 [Command Prompt]  
1  smbpasswd [options] [username]  
2  
3  options:  
4  -a                add user  
5  -d                disable user  
6  -e                enable user  
7  -x                delete user  
8  
9  #np.:  
10  
11 smbpasswd -a jnowak  
12 New SMB password: ****  
13 Retype new SMB password: ****
```

Czyli ustawienie hasła dla użytkownika zsme to:

```
smbpasswd -a zsme
```

Ustawmy hasła dla wszystkich naszych użytkowników.

To samo polecenie służy do zmiany hasła użytkownika:

```
CodeHL 1.3 [Command Prompt]
1 smbpasswd jnowak
2 New SMB password: ****
3 Retype new SMB password: ****
```

## 8. Testy działania:

Utwórzmy w katalogach domowych naszych użytkowników jakieś testowe pliki : w katalogu domowym użytkownika zsme plik zsme.txt, w katalogu domowym użytkownika stefan plik stefan.txt i tak dalej.

**Sprawdźcie wszystkie utworzone przydziały czy działają poprawnie, czy wpuszczają odpowiednich użytkowników i czy pozwalają zapisywać !!!!**

**Dostęp z cli Ubuntu:**

Instalujemy program **smbclient**

```
sudo apt install smbclient
```

podłączamy się do naszych udziałów w ten sposób:

```
sudo smbclient //[IP_address or Host_name]/share_name -U username
```

**[adres\_IP lub nazwa\_hosta]** to adres IP lub nazwa hosta serwera Samba

**[nazwa\_udziału]** to nazwa udziału w Sambie

**[nazwa\_użytkownika]** to nazwa użytkownik uzyskujący dostęp do udziału

Czyli np. podłączenie się do udziału dyrekcja przez użytkownika zsme to:

```
smbclient //192.168.1.200/dyrekcja -U zsme
```

przeglądanie zawartości katalogu to **dir**, wgrywanie coś do katalogu to **put** a ściąganie coś z katalogu to **get**

```
zsme@jkubuntu:~$ smbclient //192.168.1.200/dyrekcja -U zsme
WARNING: The "syslog" option is deprecated
Enter WORKGROUP\zsme's password:
Try "help" to get a list of possible commands.
smb: \> dir
.                D          0   Fri Dec  4 03:07:06 2020
..               D          0   Fri Dec  4 02:40:00 2020

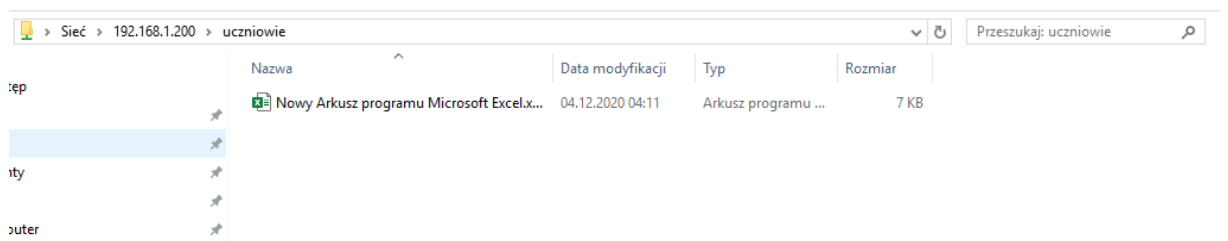
                26068772 blocks of size 1024. 20354320 blocks available
smb: \> put /home/zsme/plik.txt kopianaserwerze.txt
putting file /home/zsme/plik.txt as \kopianaserwerze.txt (0,0 kb/s) (average -nan kb/s)
smb: \> dir
.                D          0   Fri Dec  4 03:08:03 2020
..               D          0   Fri Dec  4 02:40:00 2020
kopianaserwerze.txt  A          0   Fri Dec  4 03:08:03 2020

                26068772 blocks of size 1024. 20354320 blocks available
smb: \> get kopianaserwerze.txt /home/zsme/kopiazserwera.txt
getting file \kopianaserwerze.txt of size 0 as /home/zsme/kopiazserwera.txt (0,0 KiloBytes/sec) (average -nan KiloBytes/sec)
smb: \> ^C
zsme@jkubuntu:~$ ls -l
total 0
-rw-r--r-- 1 zsme zsme 0 gru  4 03:08 kopiazserwera.txt
-rw-r--r-- 1 zsme zsme 0 gru  4 03:05 kopiazsme.txt
-rw-rw-r-- 1 zsme zsme 0 gru  4 03:03 plik.txt
zsme@jkubuntu:~$ _
```

## Dostęp z Windowsa:

Otwieramy dowolne okienko eksploratora i wpisujemy adres naszego przydziału w formacie:

```
\\192.168.1.200\uczniowie
```

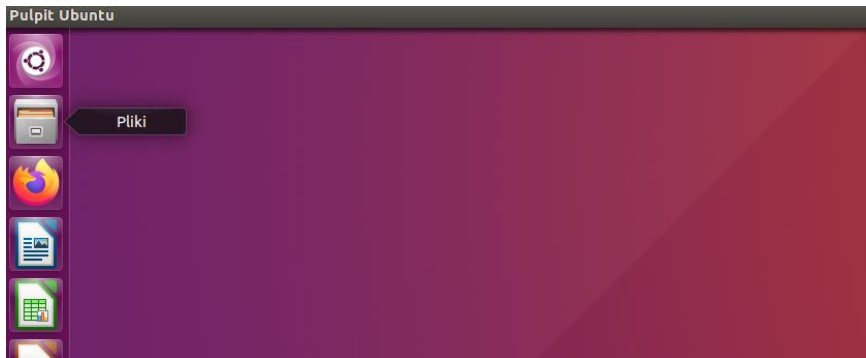


możemy już kopiować pliki jak to w Windowsie.

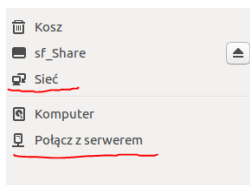
## Dostęp z Ubuntu Desktop

Najwygodniej korzysta się z udziałów w Sambie z Ubuntu desktopowego, ma on wbudowanego natywnego klienta:

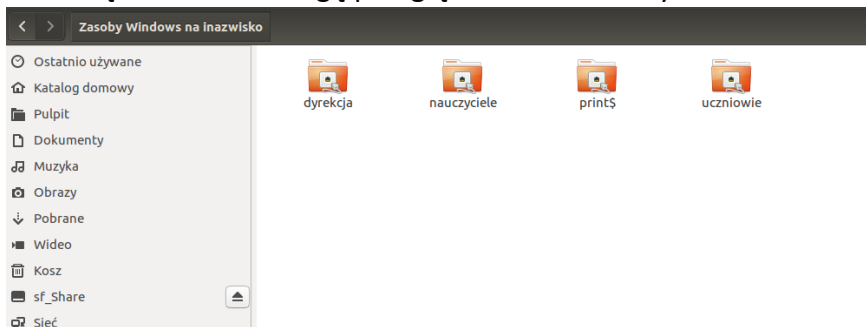
Otwieramy Pliki:



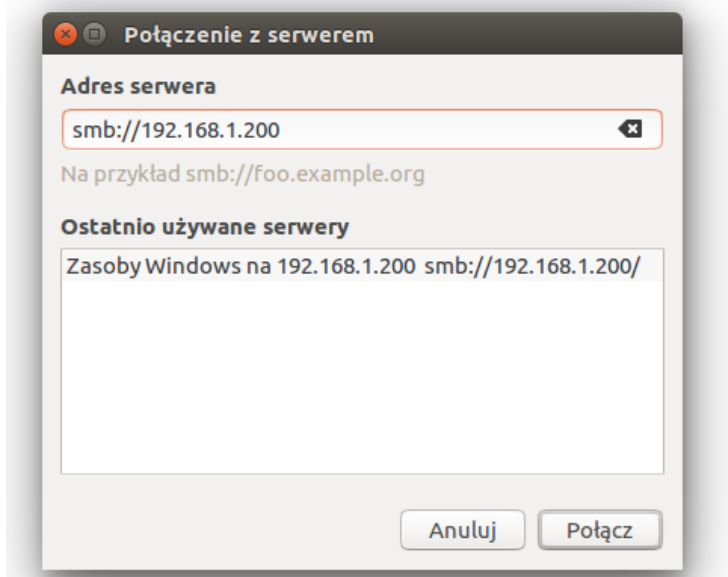
Na dole okna po lewej stronie klikamy albo w sieć, albo w Połącz z serwerem:



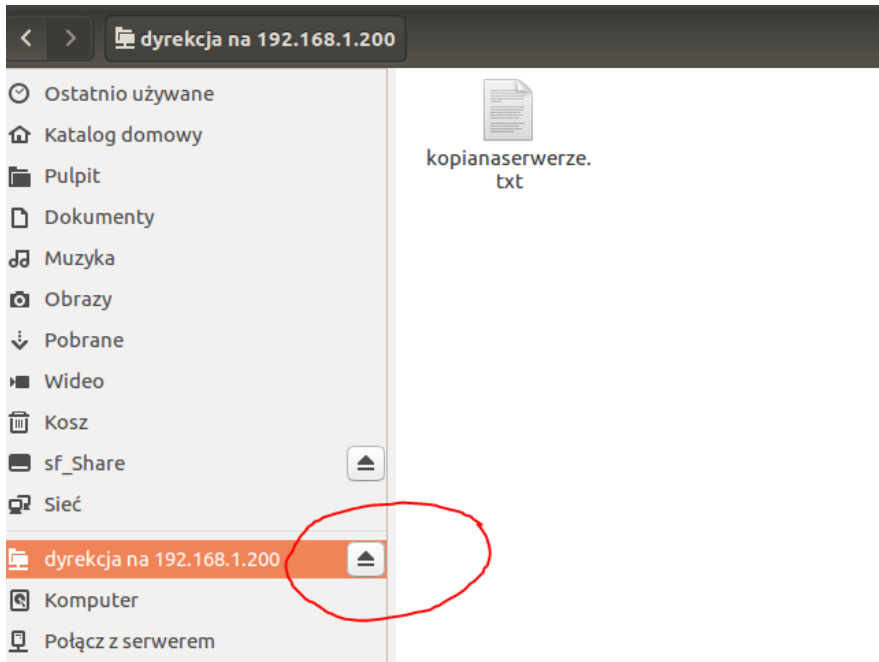
Ja kliknąłem w sieć i mogę przeglądać nasze udziały:



Przy kliknięciu w Połącz z serwerem trzeba podać adres naszego serwera:



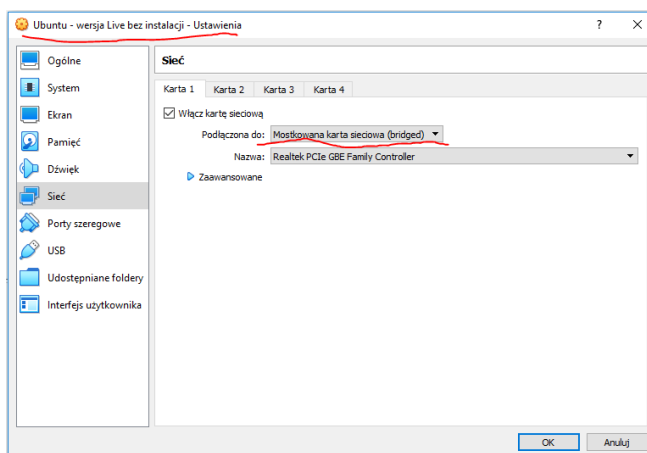
Jak chcemy się „odpiąć” (wylogować) z udziału, to klikamy w tą małą strzałkę na liście udziałów po prawej:



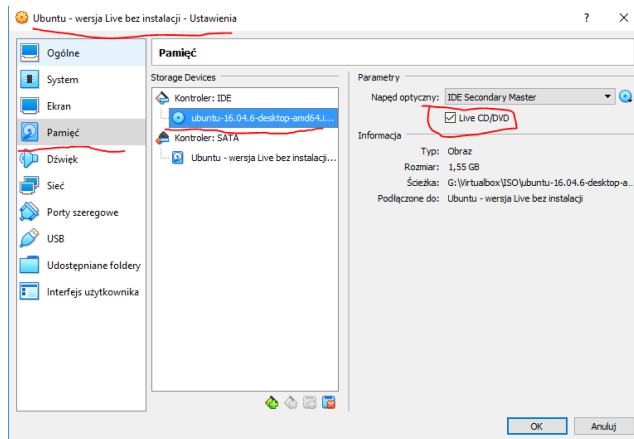
Skąd wziąć Ubuntu Desktop na testy, jak nie mamy maszyny wirtualnej ? Ściągnąć ISO, stworzyć nową maszynę w VB, podpiąć do niej ISO i nie instalować (choć polecam, przed egzaminem przyda się do powtórek) ale odpalić maszynę w trybie Live, bez instalacji:

<https://releases.ubuntu.com/18.04.5/ubuntu-18.04.5-desktop-amd64.iso>

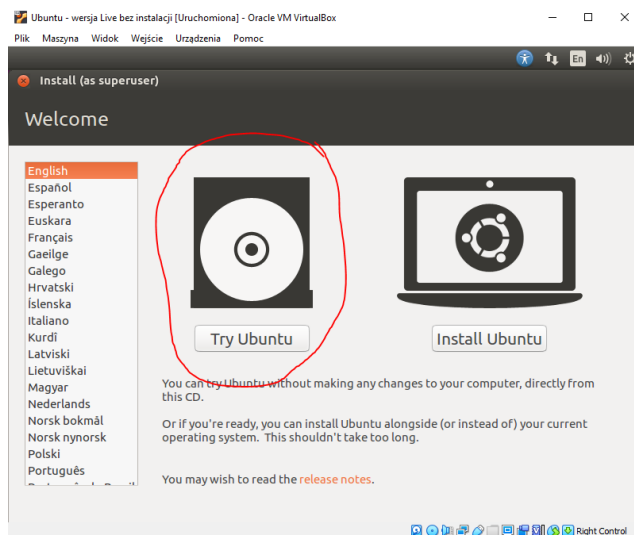
ustawiam sieciówkę na zmostkowaną



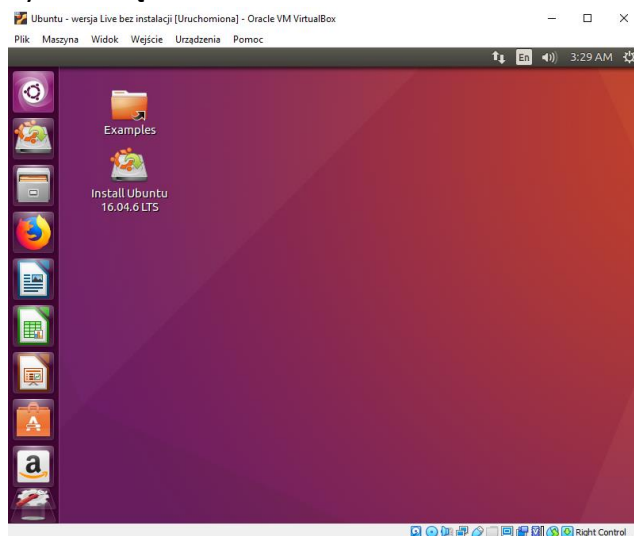
Montuje iso z instalatorem Ubuntu do napędu, zaznaczam ze to LiveCD



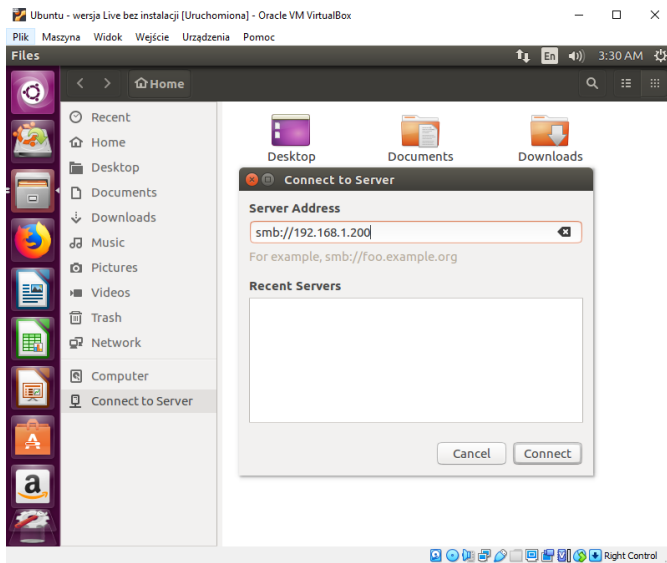
Wybieram, ze nie chce instalować, tylko wypróbować:



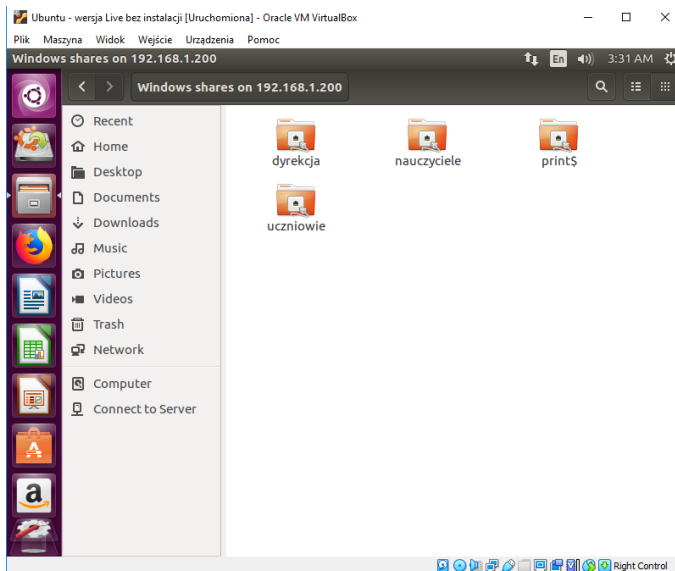
System się uruchomił:



Korzystam z Plików i łączę się z serwerem:

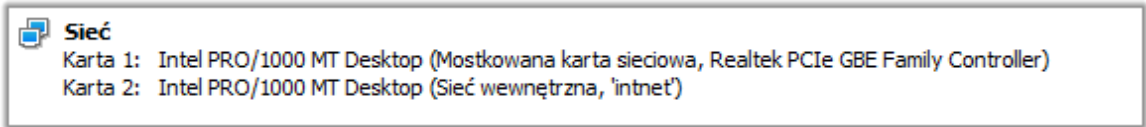


Jak widać mam normalny dostęp do udziałów na naszym serwerze Samby:



## Serwer SSH

1. Na maszynie z US przywracamy czystą migawkę (może być ta z konfiguracji kart sieciowych), **pierwszą sieć** konfigurujemy jako **zmostkowaną**, **drugą** jako **wewnętrzną intnet** (tak jak w poprzednim ćwiczeniu)



2. Przygotowujemy sobie jakąś drugą maszynę na testy (7,10, XP) sieć **wewnętrzna intnet**
3. Uruchamiamy Ubuntu, konfigurujemy kartę sieciową (tak aby adresacja pasowała do naszej sieci domowej) i **końcówka adresu IP** to był **nasz numer z dziennika**. Druga sieć to 192.168.X.1/24. U mnie niech to będzie:

pierwsza sieć

IP:10.39.100.X

M:255.0.0.0

B:10.0.0.1

DNS:8.8.8.8

druga sieć:

IP:192.168.X.1

M:255.255.255.0

**cd /etc/netplan/**

**nano 01-netcf.yaml**

**modyfikujemy ustawienia**

**netplan apply**

```
GNU nano 2.9.3 /etc/netplan/01-netcf.yaml
# This file is generated from information provided by the datasource. Changes
# to it will not persist across an instance reboot. To disable cloud-init's
# network configuration capabilities, write a file
# /etc/cloud/cloud.cfg.d/99-disable-network-config.cfg with the following:
# network: {config: disabled}
network:
  version: 2
  renderer: networkd
  ethernets:
    enp0s3:
      dhcp4: false
      addresses: [192.168.1.200/24]
      gateway4: 192.168.1.1
      nameservers:
        addresses: [1.1.1.1,8.8.8.8]
    enp0s8:
      dhcp4: false
      addresses: [10.0.0.1/8]

[ Read 18 lines ]
^G Get Help      ^O Write Out    ^W Where Is    ^K Cut Text     ^J Justify     ^C Cur Pos     M-U Undo
^X Exit          ^R Read File    ^_ Replace     ^U Uncut Text  ^T To Spell   ^_ Go To Line  M-E Redo
```

4. Uaktualniamy listę zawartości repozytoriów przed instalacją serwera SSH (od Ubuntu 16.04 w górę możemy używać `apt update` zamiast `apt-get update`, tak samo przy instalacji wystarczy `apt install` zamiast `apt-get install`)

#### **apt update**

```
root@jkubuntu:/etc/netplan# apt update
Stary:1 http://pl.archive.ubuntu.com/ubuntu bionic InRelease
Stary:2 http://pl.archive.ubuntu.com/ubuntu bionic-updates InRelease
Stary:3 http://pl.archive.ubuntu.com/ubuntu bionic-backports InRelease
Stary:4 http://pl.archive.ubuntu.com/ubuntu bionic-security InRelease
Czytanie list pakietów... Gotowe
Budowanie drzewa zależności
Odczyt informacji o stanie... Gotowe
38 packages can be upgraded. Run 'apt list --upgradable' to see them.
root@jkubuntu:/etc/netplan#
```

5. Instalujemy serwer ssh - użyjemy pakietu **openssh-server**

**apt install openssh-server**

```

ntu1.18.04 [248 kB]
Pobieranie:3 http://p1.archive.ubuntu.com/ubuntu bionic-updates/main amd64 openssh-sftp-server amd64
1:7.6p1-4ubuntu0.3 [45,6 kB]
Pobieranie:4 http://p1.archive.ubuntu.com/ubuntu bionic-updates/main amd64 openssh-server amd64 1:7.
6p1-4ubuntu0.3 [333 kB]
Pobieranie:5 http://p1.archive.ubuntu.com/ubuntu bionic-updates/main amd64 ssh-import-id all 5.7-0ub
untu1.1 [10,9 kB]
Pobrano 683 kB w 0s (14578 kB/s)
Prekonfiguracja pakietów ...
Wybieranie wcześniej niewybranego pakietu libwrap0:amd64.
(Odczytywanie bazy danych ... 99682 pliki i katalogi obecnie zainstalowane.)
Przygotowywanie do rozpakowania pakietu ../libwrap0_7.6.q-27_amd64.deb ...
Rozpakowywanie pakietu libwrap0:amd64 (7.6.q-27) ...
Wybieranie wcześniej niewybranego pakietu ncurses-term.
Przygotowywanie do rozpakowania pakietu ../ncurses-term_6.1-1ubuntu1.18.04_all.deb ...
Rozpakowywanie pakietu ncurses-term (6.1-1ubuntu1.18.04) ...
Wybieranie wcześniej niewybranego pakietu openssh-sftp-server.
Przygotowywanie do rozpakowania pakietu ../openssh-sftp-server_1%3a7.6p1-4ubuntu0.3_amd64.deb ...
Rozpakowywanie pakietu openssh-sftp-server (1:7.6p1-4ubuntu0.3) ...
Wybieranie wcześniej niewybranego pakietu openssh-server.
Przygotowywanie do rozpakowania pakietu ../openssh-server_1%3a7.6p1-4ubuntu0.3_amd64.deb ...
Rozpakowywanie pakietu openssh-server (1:7.6p1-4ubuntu0.3) ...
Wybieranie wcześniej niewybranego pakietu ssh-import-id.
Przygotowywanie do rozpakowania pakietu ../ssh-import-id_5.7-0ubuntu1.1_all.deb ...
Rozpakowywanie pakietu ssh-import-id (5.7-0ubuntu1.1) ...
Konfigurowanie pakietu ncurses-term (6.1-1ubuntu1.18.04) ...
Konfigurowanie pakietu openssh-sftp-server (1:7.6p1-4ubuntu0.3) ...
Konfigurowanie pakietu ssh-import-id (5.7-0ubuntu1.1) ...
Konfigurowanie pakietu libwrap0:amd64 (7.6.q-27) ...
Konfigurowanie pakietu openssh-server (1:7.6p1-4ubuntu0.3) ...

Creating config file /etc/ssh/sshd_config with new version
Created symlink /etc/systemd/system/ssh.service → /lib/systemd/system/ssh.service.
Created symlink /etc/systemd/system/multi-user.target.wants/ssh.service → /lib/systemd/system/ssh.se
rvice.

Progress: [ 92%] [#####.....]

```

(jak widzimy, polecenie apt install pokazuje nam pasek postępu instalacji)

6. Nasz serwer zaraz po instalacji powinien działać i pozwalać na podłączenie się na domyślnym 22 porcie. Możemy to sprawdzić przy pomocy polecenia:

**service ssh status**

lub

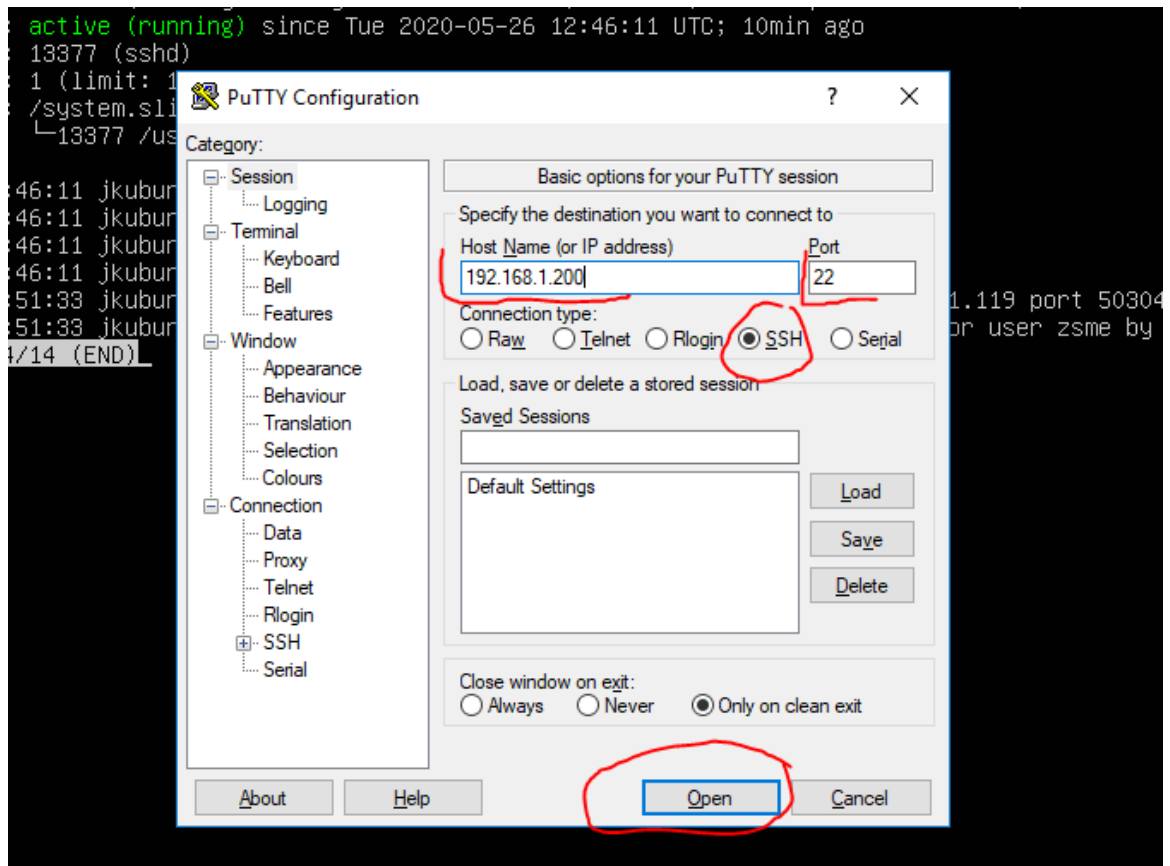
**systemctl status ssh**

(żeby wyjść z polecenia używamy Ctrl + C)

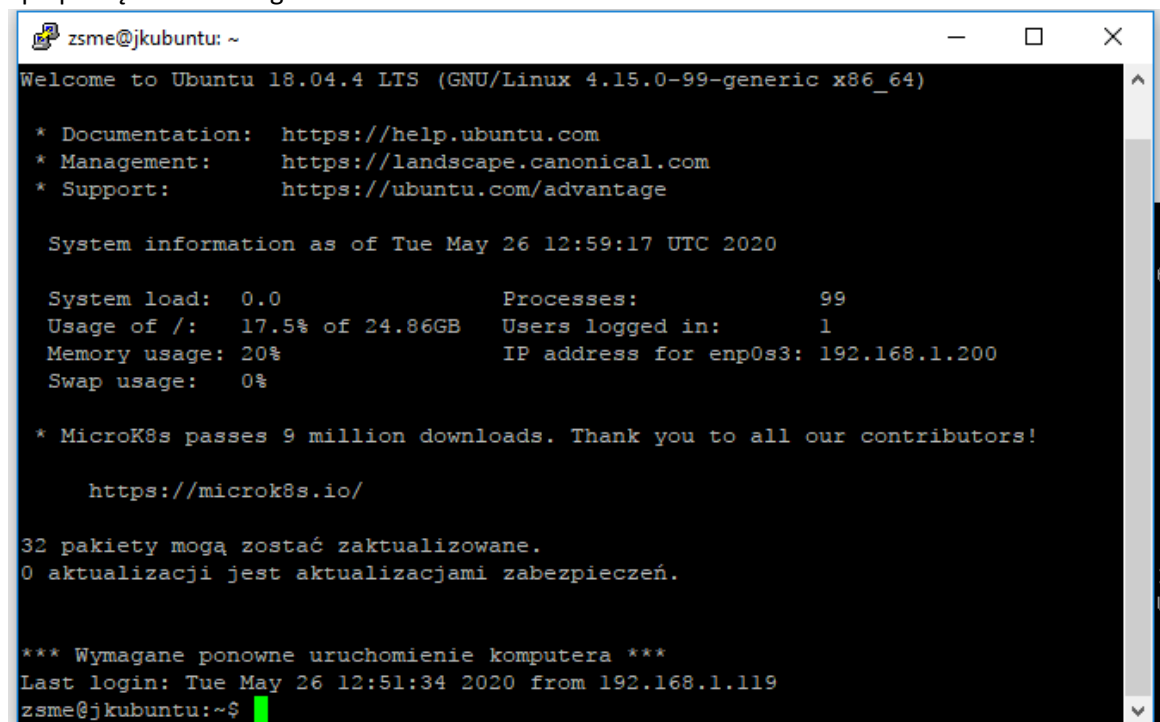
```
root@jkubuntu:/etc/ssh# service ssh status
● ssh.service - OpenBSD Secure Shell server
   Loaded: loaded (/lib/systemd/system/ssh.service; enabled; vendor preset: enabled)
   Active: active (running) since Tue 2020-05-26 12:46:11 UTC; 10min ago
 Main PID: 13377 (sshd)
    Tasks: 1 (limit: 1108)
   CGroup: /system.slice/ssh.service
           └─13377 /usr/sbin/sshd -D

maj 26 12:46:11 jkubuntu systemd[1]: Starting OpenBSD Secure Shell server...
maj 26 12:46:11 jkubuntu sshd[13377]: Server listening on 0.0.0.0 port 22.
maj 26 12:46:11 jkubuntu sshd[13377]: Server listening on :: port 22.
maj 26 12:46:11 jkubuntu systemd[1]: Started OpenBSD Secure Shell server.
maj 26 12:51:33 jkubuntu sshd[13592]: Accepted password for zsme from 192.168.1.119 port 50304 ssh2
maj 26 12:51:33 jkubuntu sshd[13592]: pam_unix(sshd:session): session opened for user zsme by (uid=0)
lines 1-14/14 (END)
```

7. Powinniśmy być się w stanie podłączyć na adres naszego serwera na domyślnym 22 porcie. Uruchamiamy Putty i zestawiamy połączenie:



i po podłączeniu i zalogowaniu:



8. Konfiguracja naszego serwera SSH znajduje się w katalogu `/etc/ssh` w pliku `sshd_config`

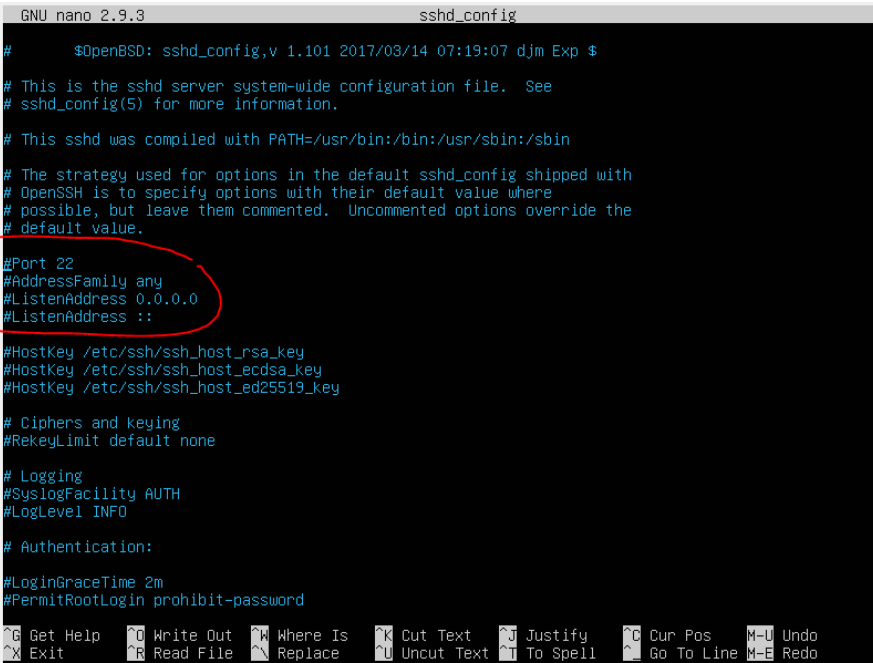
Przed zmianą konfiguracji zrobmy sobie oczywiście **kopie pliku konfiguracyjnego**, żeby w

razie wtopy można go było łatwo przywrócić:

```
cp /etc/ssh/sshd_config /etc/ssh/sshd_config.bac
```

otwieramy w edytorze plik konfiguracyjny

```
nano /etc/ssh/sshd_config
```



```
GNU nano 2.9.3 sshd_config
# $OpenBSD: sshd_config,v 1.101 2017/03/14 07:19:07 djm 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.
#Port 22
#AddressFamily any
#ListenAddress 0.0.0.0
#ListenAddress ::

#HostKey /etc/ssh/ssh_host_rsa_key
#HostKey /etc/ssh/ssh_host_ecdsa_key
#HostKey /etc/ssh/ssh_host_ed25519_key

# Ciphers and keying
#RekeyLimit default none

# Logging
#SyslogFacility AUTH
#LogLevel INFO

# Authentication:

#LoginGraceTime 2m
#PermitRootLogin prohibit-password

G Get Help  O Write Out  W Where Is  K Cut Text  J Justify  C Cur Pos  M-U Undo
X Exit      R Read File  R Replace  U Uncut Text  T To Spell  G Go To Line M-B Redo
```

najczęściej będzie nasz interesowała opcja wybrania adresu na którym ma działać serwer oraz portu

Możemy dopisać dodatkowe porty na których ma nasłuchiwać nasz serwer, zmienić domyślny port - całość polega na odkomentowaniu odpowiedniej linijki i ewentualnie dopisania żądanych ustawień - założymy, że chcemy, żeby oprócz domyślnego portu serwer działał też na porcie 2222:

```
# 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.

Port 22
Port 2222
#AddressFamily any
#ListenAddress 0.0.0.0
#ListenAddress ::

#HostKey /etc/ssh/ssh_host_rsa_key
#HostKey /etc/ssh/ssh_host_ecdsa_key
#HostKey /etc/ssh/ssh_host_ed25519_key

# Ciphers and keying
#RekeyLimit default none

# Logging
#SyslogFacility AUTH
#LogLevel INFO

# Authentication:

#LoginGraceTime 2m

[ Wrote 123 lines ]
^G Get Help  ^O Write Out  ^W Where Is   ^K Cut Text   ^J Justify   ^C Cur Pos   M-U Undo
Use "fg" to return to nano.  ^R Replace    ^U Uncut Text ^T To Spell   ^_ Go To Line  M-E Redo
```

oczywiście, żeby nasze zmiany miały skutek musimy zrestartować usługę poleceniem:

**service ssh restart**

lub

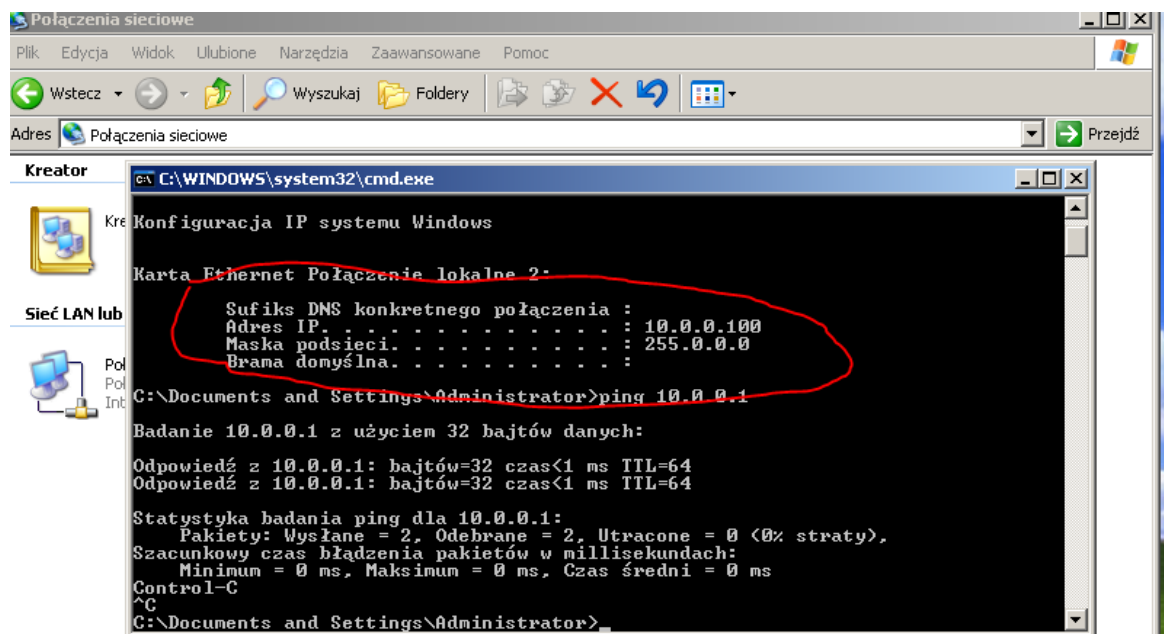
**systemctl restart ssh**

Serwer powinien pozwalać teraz na połączenia na porcie 22 i 2222

9. Przygotujmy sobie teraz maszynę testową - ja użyję Windowsa XP – najpierw uruchomię go z kartą sieciową zmostkowaną, żeby mieć internet, ściągnę Putty (32bity, bo to XP)



Po ściągnięciu Putty, zmieniamy typ sieciówki na wewnętrzną i ustawiamy adresację na pasującą do drugiej karty sieciowej na naszym serwerze, czyli na przykład 10.0.0.100/8



- Możemy ustawić na jakim adresie/adresach IP ma nasłuchiwać nasz serwer – domyślnie nasłuchuje na wszystkich (powinien Was wpuścić i z komputera gospodarza i z komputera testowego na obu ustawionych portach) – odkomentowanie odpowiedniej linii i wpisanie adresu spowoduje, że serwer będzie nasłuchiwał tylko na jednym adresie:

```
# 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.

Port 22
Port 2222
#AddressFamily any
ListenAddress 192.168.1.200
#ListenAddress ..

#HostKey /etc/ssh/ssh_host_rsa_key
#HostKey /etc/ssh/ssh_host_ecdsa_key
#HostKey /etc/ssh/ssh_host_ed25519_key

# Ciphers and keying
#RekeyLimit default none

# Logging
#SyslogFacility AUTH
#LogLevel INFO

# Authentication:

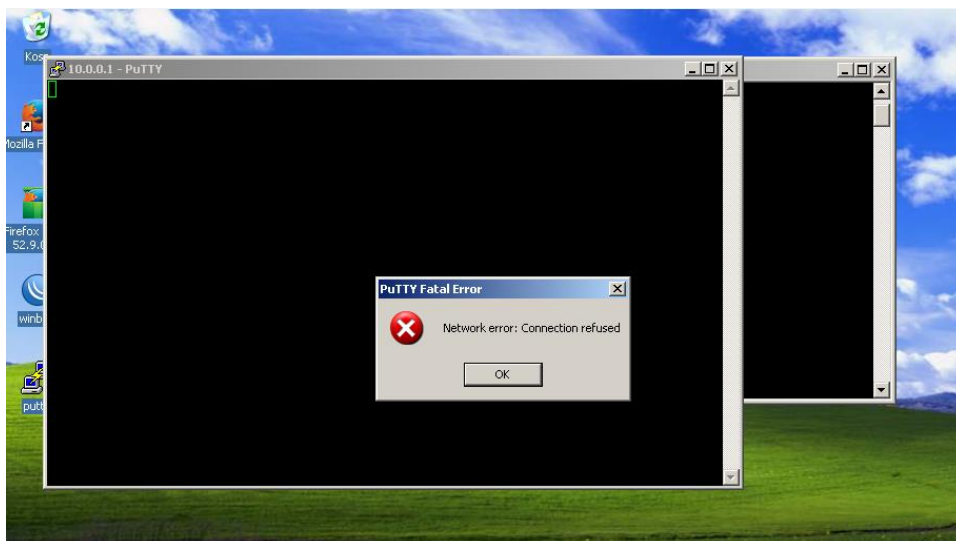
#LoginGraceTime 2m

[ Wrote 123 lines ]
Get Help Write Out Where Is Cut Text Justify Cur Pos M-U Undo
Use "fg" to return to nano. Replace U Uncut Text T To Spell Go To Line M-E Redo

[3]+ Stopped nano sshd_config
root@jkubuntu:/etc/ssh# systemctl restart ssh
root@jkubuntu:/etc/ssh# _
```

Oczywiście po zapisaniu restartujemy za każdym razem usługę ssh (service ssh restart)

Teraz próbując zalogować się z komputera testowego na adres 10.0.0.1 serwer powinien odrzucić połączenie:



11. Możemy ręcznie dopisać drugi adres w konfiguracji – tutaj dodatkowo podałem port 3333 – przy takiej notacji serwer będzie zezwalał na łączenie się na tym adresie tylko na porcie 3333 pomijając te ustawione wyżej:

```
# 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.

Port 22
Port 2222
#AddressFamily any
ListenAddress 192.168.1.200
ListenAddress 10.0.0.1:3333
#ListenAddress :

#HostKey /etc/ssh/ssh_host_rsa_key
#HostKey /etc/ssh/ssh_host_ecdsa_key
#HostKey /etc/ssh/ssh_host_ed25519_key

# Ciphers and keying
#RekeyLimit default none

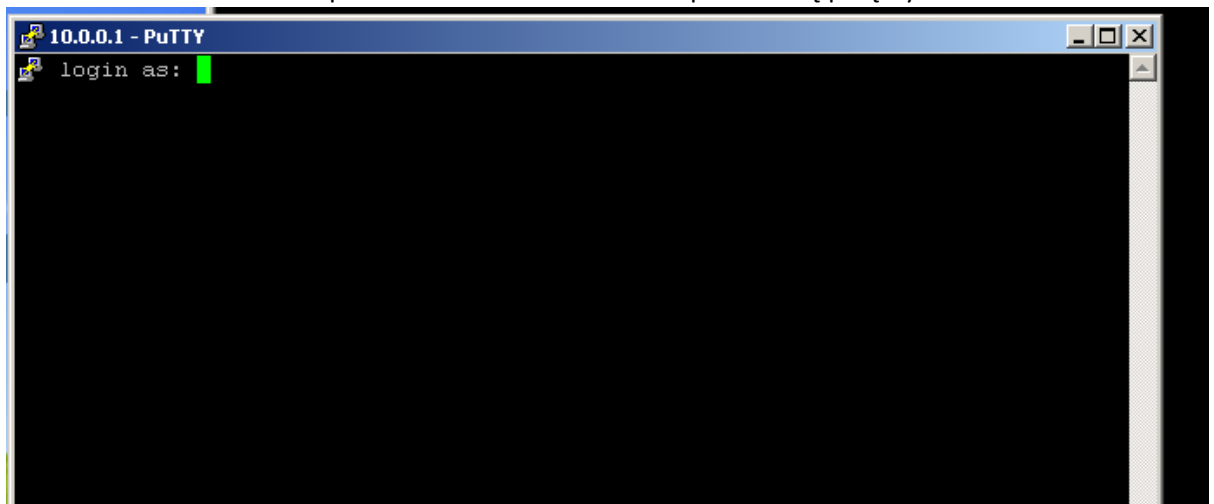
# Logging
#SyslogFacility AUTH
#LogLevel INFO

# Authentication:

[ Wrote 124 lines ]
^G Get Help ^O Write Out ^W Where Is ^K Cut Text ^J Justify ^G Cur Pos M-U Undo
Use "fg" to return to nano. ^R Replace ^U Uncut Text ^T To Spell ^_ Go To Line M-E Redo

[4]+ Stopped nano sshd_config
root@jkubuntu:/etc/ssh#
```

Oczywiście po restarcie usługi ssh i próbie połączenia się z komputera testowego na 10.0.0.1:22 serwer odrzuci połączenie a na 10.0.0.1:3333 pozwoli się połączyć.



## 12. Blokada lub zezwalanie na dostęp użytkowników/grup

Możemy też chcieć zablokować lub zezwalać na dostęp określonych użytkowników/grup

Musimy wtedy w naszym configu użyć odpowiednich zmiennych (działają zgodnie z nazwami):

```
DenyUsers user1 user2 user3
```

```
DenyGroups group1 group2
```

```
AllowUsers user1 user2
AllowGroups group1 group2
```

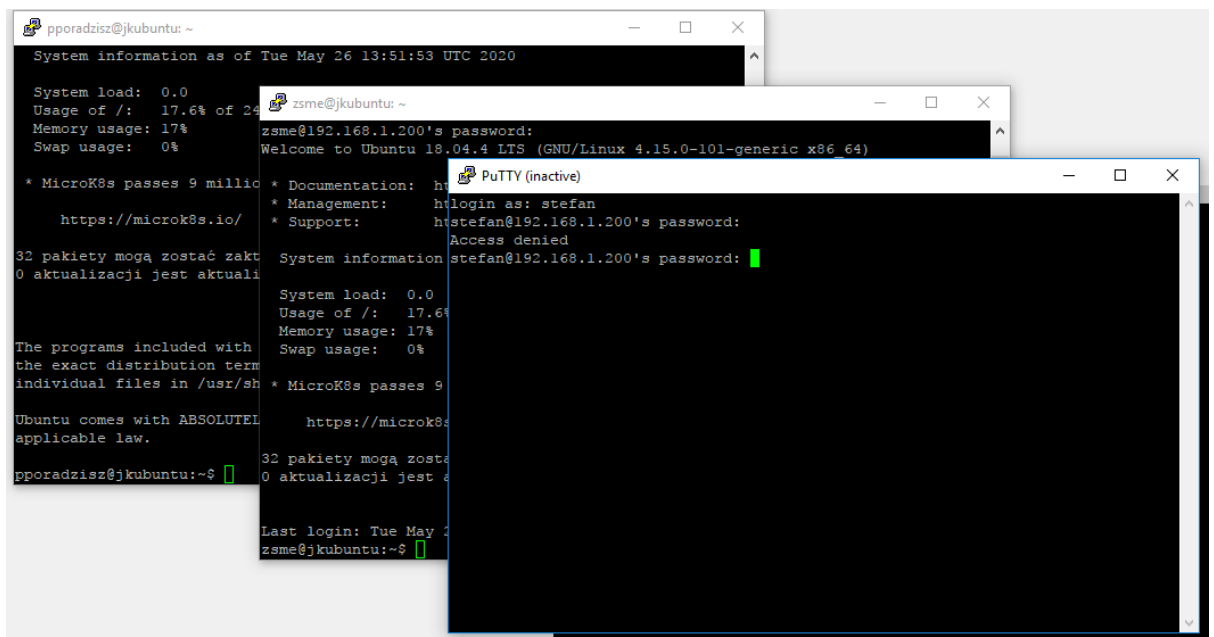
Założmy na serwerze testowe konto **inazwisko** z hasłem **123qwe** oraz **stefan** z hasłem **123qwe**

```
adduser pporadzisz
adduser stefan
```

I teraz spróbujemy ustawić tak, że konta inazwisko i zsmie są wpuszczane na serwer konto stefan nie:

```
GNU nano 2.9.3 sshd_config
# $OpenBSD: sshd_config,v 1.101 2017/03/14 07:19:07 djm 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.
Port 22
Port 2222
#AddressFamily any
ListenAddress 192.168.1.200
ListenAddress 10.0.0.1:3333
#ListenAddress :
DenyUsers stefan
#HostKey /etc/ssh/ssh_host_rsa_key
#HostKey /etc/ssh/ssh_host_ecdsa_key
#HostKey /etc/ssh/ssh_host_ed25519_key
# Ciphers and keying
```

restartujemy serwer SSH i sprawdzamy:



Podobnie oczywiście będzie działało to z grupami.

## 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

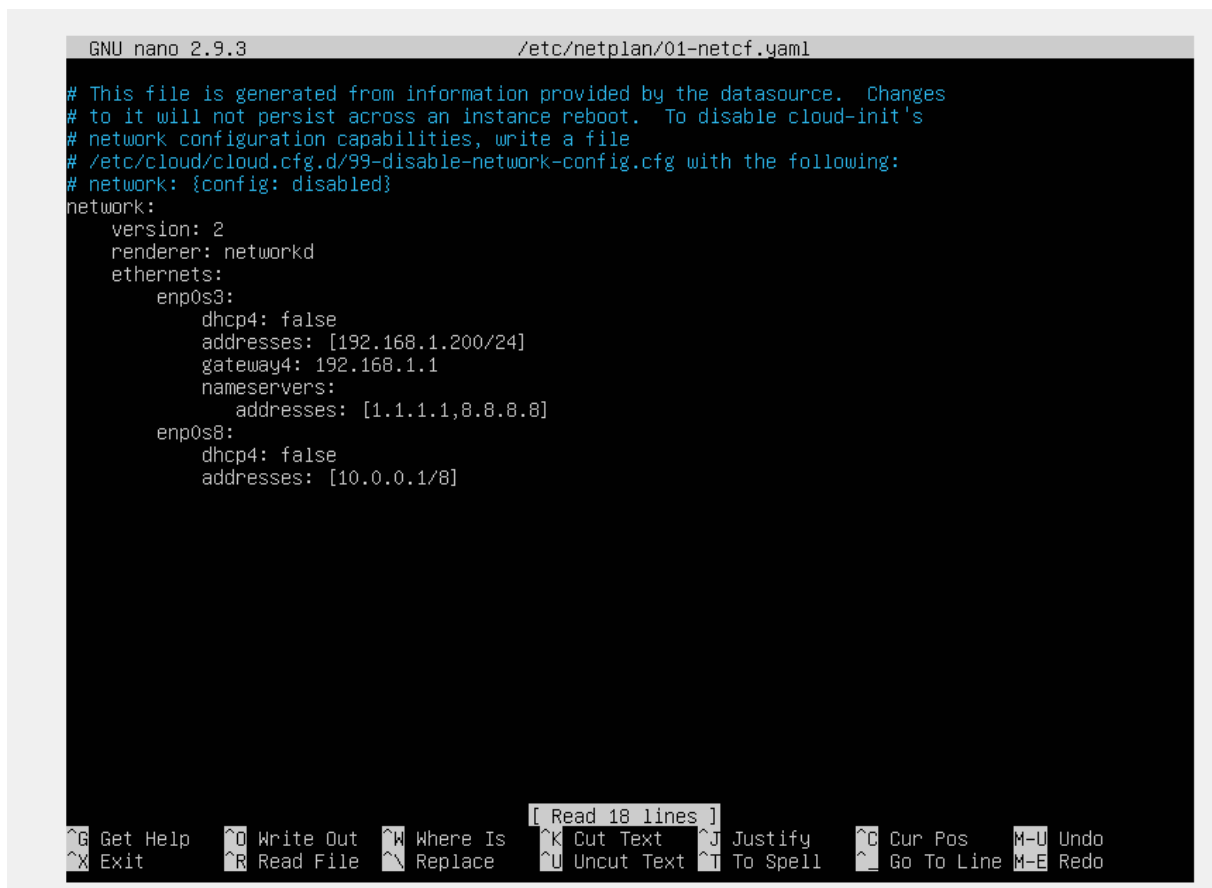
## 2. Konfigurujemy karty sieciowe na maszynie testowej z WinXP:

- \* 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:



```
GNU nano 2.9.3 /etc/netplan/01-netcf.yaml
# This file is generated from information provided by the datasource. Changes
# to it will not persist across an instance reboot. To disable cloud-init's
# network configuration capabilities, write a file
# /etc/cloud/cloud.cfg.d/99-disable-network-config.cfg with the following:
# network: {config: disabled}
network:
  version: 2
  renderer: networkd
  ethernets:
    enp0s3:
      dhcp4: false
      addresses: [192.168.1.200/24]
      gateway4: 192.168.1.1
      nameservers:
        addresses: [1.1.1.1,8.8.8.8]
    enp0s8:
      dhcp4: false
      addresses: [10.0.0.1/8]
```

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

#### 4. Aktualizujemy zawartość repozytoriów:

```
sudo apt update
```

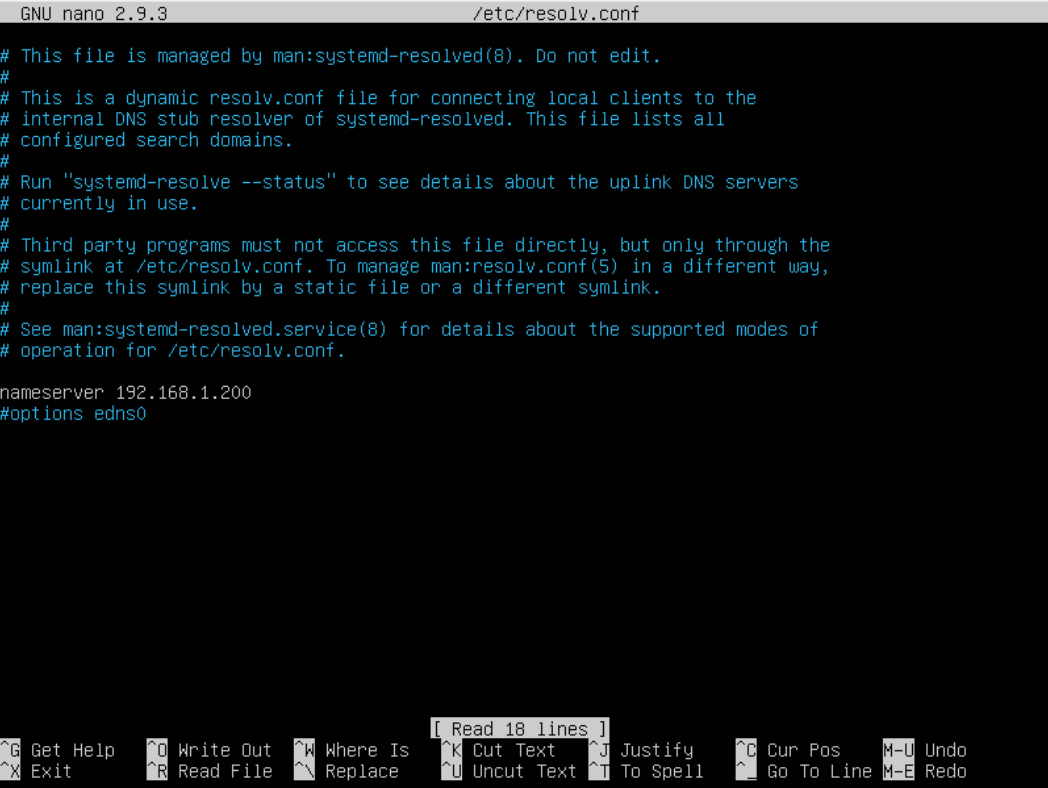
#### 5. Instalujemy serwer DNS:

```
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:



```
GNU nano 2.9.3 /etc/resolv.conf
# This file is managed by man:systemd-resolved(8). Do not edit.
#
# This is a dynamic resolv.conf file for connecting local clients to the
# internal DNS stub resolver of systemd-resolved. This file lists all
# configured search domains.
#
# Run "systemd-resolve --status" to see details about the uplink DNS servers
# currently in use.
#
# Third party programs must not access this file directly, but only through the
# symlink at /etc/resolv.conf. To manage man:resolv.conf(5) in a different way,
# replace this symlink by a static file or a different symlink.
#
# See man:systemd-resolved.service(8) for details about the supported modes of
# operation for /etc/resolv.conf.
nameserver 192.168.1.200
#options edns0
```

#### 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 działać na adresie 192.168.1.200 i ma się odpytywać serwer DNS google, jeżeli czego nie będzie wiedział.

```
GNU nano 2.9.3 /etc/bind/named.conf.options Modified
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 {
        0.0.0.0;
    };

    //=====
    // 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 no;

    auth-nxdomain no; # conform to RFC1035
    //listen-on-v6 { any; };

    listen-on {192.168.1.200;
    };

    forward only;
};

^G Get Help ^O Write Out ^W Where Is ^K Cut Text ^J Justify ^C Cur Pos M-U Undo
^X Exit ^R Read File ^\ Replace ^U Uncut Text ^T To Spell ^_ Go To Line M-E Redo
```

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 CodeHL 1.7 [Command Prompt]
1 #utworzenie grupy ZaufaniKlienci mogącej odpytywać ten serwer DNS
2 acl ZaufaniKlienci {
3     192.168.0.0/24;
4     localhost;
5     localnets; #lokalne sieci serwera (wszystkie interfejsy)
6 };
7
8 options {
9 #lokalizacja pamięci podręcznej serwera DNS:
10     directory "/var/cache/bind";
11
12 #zezwolenie na odpytywanie tego serwera DNS tylko przez określonych klientów (np. z utworzonej grupy):
13     allow-query {
14         ZaufaniKlienci;
15         10.0.0.0/8;
16     };
17
18 #Adresy IP serwerów DNS, do których będą wysyłane dalsze zapytania.
19     forwarders {
20         8.8.8.8;
21         8.8.4.4;
22     };
23
24 #wybranie adresów IP, na których serwer DNS będzie nasłuchiwał:
25     listen-on {
26         192.168.0.1;
27     };
28
29 #forward (only | first)
30 #forward only - zapytania przesyłane są wyłącznie do adresów określonych w sekcji forwarders
31 #forward first - (domyślnie) jeżeli serwery z sekcji forwarders nie są w stanie udzielić odpowiedzi,
32 #                serwer podejmie próbę rozwiązania nazwy np. za pomocą serwerów najwyższego poziomu (root)
33     forward only;
34 };
```

## 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:)

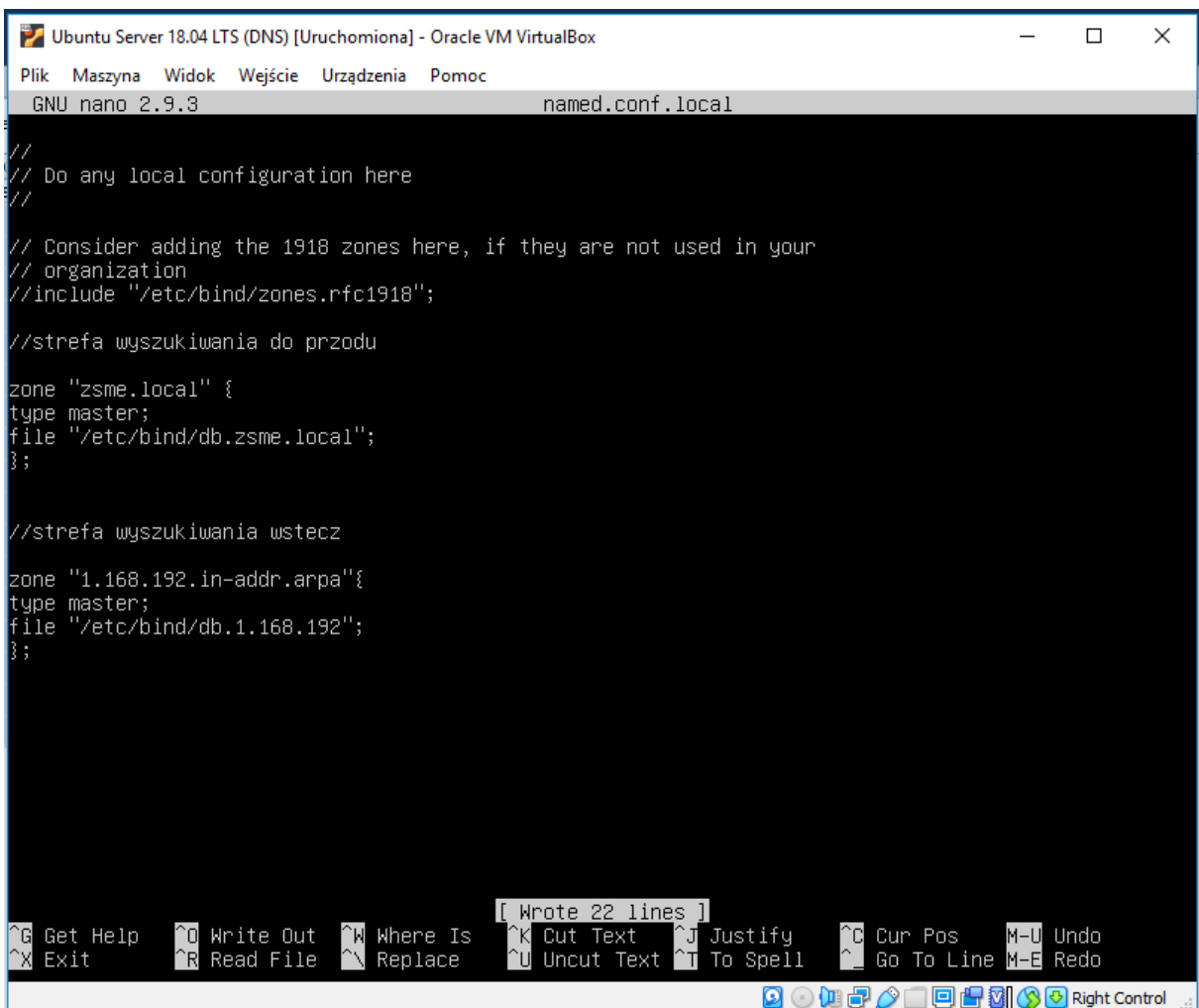
### Definicja strefy wyszukiwania w przód i wstecz.

```
Strefa wyszukiwania w przód: named.conf.local CodeHL 1.7 [Command Prompt]
1 #definicja strefy wyszukiwania w przód np.: zsmc.local - 192.168.0.1
2
3 zone "zsmc.local" {
4     #typ strefy: master | slave
5     type master;
6
7     #lokalizacja pliku z definicją danej strefy
8     file "/etc/bind/db.zsmc.local";
9
10    #zezwozenie na wysyłanie danych do dodatkowego serwera DNS
11    #allow-transfer { 192.168.0.2; };
12
13    #serwer dodatkowy - slave - powinien mieć zdefiniowany adres serwera głównego
14    #masters { 192.168.0.1; };
15 };
16
17 #definicja strefy wyszukiwania wstecz np.: 192.168.0.1 - zsmc.local
18 zone "0.168.192.in-addr.arpa" {
19     #typ strefy: master | slave
20     type master;
21
22     #lokalizacja pliku z definicją danej strefy
23     file "/etc/bind/db.0.168.192";
24
25     #zezwozenie na wysyłanie danych do dodatkowego serwera DNS
26     #allow-transfer { 192.168.0.2; };
27 };
```

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:



```
Ubuntu Server 18.04 LTS (DNS) [Uruchomiona] - Oracle VM VirtualBox
Plik Maszyna Widok Wejście Urządzenia Pomoc
GNU nano 2.9.3 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";
//strefa wyszukiwania do przodu
zone "zsmc.local" {
type master;
file "/etc/bind/db.zsmc.local";
};
//strefa wyszukiwania wstecz
zone "1.168.192.in-addr.arpa"{
type master;
file "/etc/bind/db.1.168.192";
};
[ Wrote 22 lines ]
^G Get Help ^O Write Out ^W Where Is ^K Cut Text ^J Justify ^C Cur Pos M-U Undo
^X Exit ^R Read File ^\ Replace ^U Uncut Text ^T To Spell ^_ Go To Line M-E Redo
Right Control
```

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.

```
GNU nano 2.9.3 db.zsme.local
;
; BIND data file for local loopback interface
;
$TTL 604800
@ IN SOA ns1.zsme.local. admin.zsme.local. (
    2 ; Serial
    604800 ; Refresh
    86400 ; Retry
    2419200 ; Expire
    604800 ) ; Negative Cache TTL
;
@ IN NS ns1.zsme.local.
@ IN A 192.168.1.200
ns1 IN A 192.168.1.200
xp IN A 192.168.1.150
komp1 IN CNAME xp
router IN A 192.168.1.1

[ Wrote 17 lines ]
^G Get Help ^O Write Out ^W Where Is ^K Cut Text ^J Justify ^C Cur Pos M-U Undo
^X Exit ^R Read File ^_ Replace ^U Uncut Text ^T To Spell ^_ Go To Line M-E Redo
```

## 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, że 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 skleci to w całość**.

```
GNU nano 2.9.3                               db.1.168.192
;
; BIND reverse data file for local loopback interface
;
$TTL      604800
@         IN      SOA      ns1.zsme.local. admin.zsme.local. (
; Serial
          1          ; Refresh
          604800     ; Retry
          86400     ; Expire
          2419200   ; Negative Cache TTL
          604800 )
;
@         IN      NS       ns1.zsme.local.
200      IN      PTR      ns1.zsme.local.
150      IN      PTR      xp.zsme.local.
1_       IN      PTR      router.zsme.local.

[ Wrote 15 lines ]
^G Get Help      ^O Write Out    ^W Where Is    ^K Cut Text     ^J Justify     ^C Cur Pos     M-U Undo
^X Exit          ^R Read File   ^_ Replace     ^U Uncut Text  ^T To Spell   ^_ Go To Line  M-E Redo
```

tak jak wcześniej pisałem – jeżeli zadeklarowalibyśmy strefę wyszukiwania wstecz dla sieci 10.0.0/8:

```
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]
1	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]
1	@	20H	IN	SOA	ns1.zsme.local. root.zsme.local (
2				1234	; serial
3				10800s	; refresh = 3 godziny
4				15M	; retry = 15 minut
5				2w12h	; expiry = 2 tygodnie i 12 godzin
6				2h30M	; neg. cache = 2 i pół godziny
7				)	

## 10. Restart usługi i testy

Po skonfigurowaniu stref restartujemy nasz serwer DNS:

**service bind9 restart**

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

**service bind9 status**

```
root@jkubuntu:/etc/bind# service bind9 status
• bind9.service - BIND Domain Name Server
  Loaded: loaded (/lib/systemd/system/bind9.service; enabled; vendor preset: enabled)
  Active: active (running) since Tue 2020-11-03 16:17:45 UTC; 1min 42s ago
  Docs: man:named(8)
  Process: 2085 ExecStop=/usr/sbin/rndc stop (code=exited, status=0/SUCCESS)
  Main PID: 2089 (named)
  Tasks: 4 (limit: 1108)
  CGroup: /system.slice/bind9.service
          └─2089 /usr/sbin/named -f -u bind

lis 03 16:17:45 jkubuntu named[2089]: managed-keys-zone: loaded serial 9
lis 03 16:17:45 jkubuntu named[2089]: zone 0.in-addr.arpa/IN: loaded serial 1
lis 03 16:17:45 jkubuntu named[2089]: zone 127.in-addr.arpa/IN: loaded serial 1
lis 03 16:17:45 jkubuntu named[2089]: zone 1.168.192.in-addr.arpa/IN: loaded serial 1
lis 03 16:17:45 jkubuntu named[2089]: zone 255.in-addr.arpa/IN: loaded serial 1
lis 03 16:17:45 jkubuntu named[2089]: zone zsme.local/IN: loaded serial 2
lis 03 16:17:45 jkubuntu named[2089]: zone localhost/IN: loaded serial 2
lis 03 16:17:45 jkubuntu named[2089]: all zones loaded
lis 03 16:17:45 jkubuntu named[2089]: running
lis 03 16:17:45 jkubuntu named[2089]: managed-keys-zone: Key 20326 for zone . acceptance timer compl
lines 1-20/20 (END)
```

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**

```
GNU nano 2.9.3 /etc/resolv.conf
# This file is managed by man:systemd-resolved(8). Do not edit.
#
# This is a dynamic resolv.conf file for connecting local clients to the
# internal DNS stub resolver of systemd-resolved. This file lists all
# configured search domains.
#
# Run "systemd-resolve --status" to see details about the uplink DNS servers
# currently in use.
#
# Third party programs must not access this file directly, but only through the
# symlink at /etc/resolv.conf. To manage man:resolv.conf(5) in a different way,
# replace this symlink by a static file or a different symlink.
#
# See man:systemd-resolved.service(8) for details about the supported modes of
# operation for /etc/resolv.conf.

nameserver 192.168.1.200
options edns0

[ Read 18 lines ]
^G Get Help  ^O Write Out  ^W Where Is   ^K Cut Text   ^J Justify    ^C Cur Pos    M-U Undo
^X Exit      ^R Read File  ^\ Replace   ^U Uncut Text ^T To Spell   ^_ Go To Line  M-E Redo
```

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 polecenie nslookup a po uruchomieniu programu wpisać nazwę domenową, o którą chcemy odpytać serwer DNS:

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

Name:   zsme.local
Address: 192.168.1.200
>
```

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 też 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.

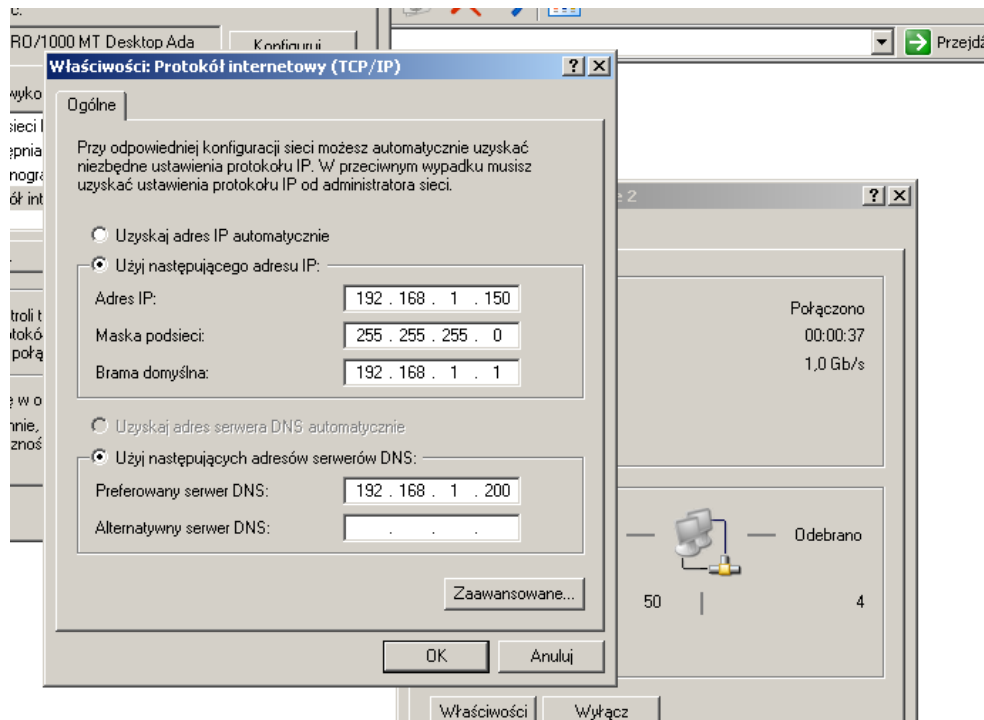
Odpalmy teraz maszynę testową z XP, skonfigurujemy ją do pracy w tej samej sieci z pasującą adresacją:

karta sieciowa: zmostkowana

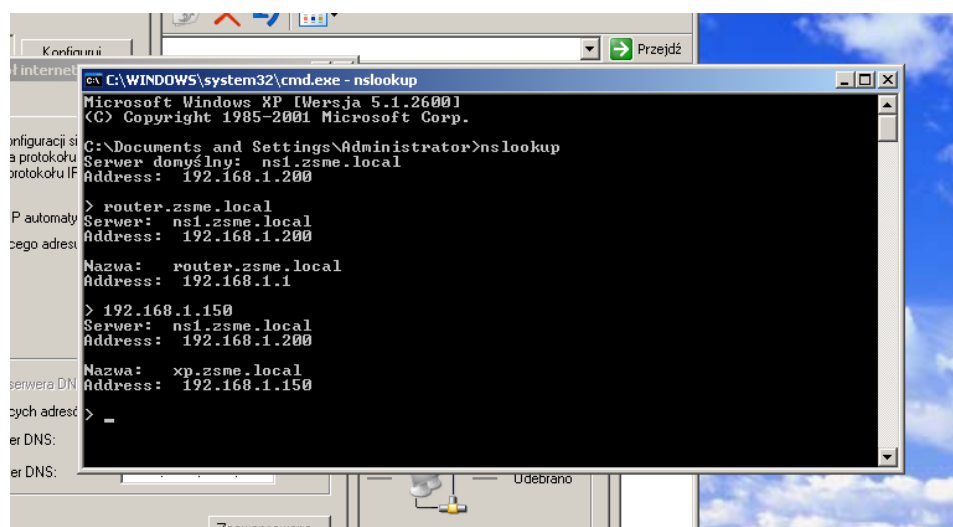
IP:192.168.1.150/24

B:192.168.1.1

DNS:192.168.1.200 (adres naszego Ubuntu Server z uruchomionym serwerem DNS)



Użyjmy też narzędzia nslookup i odpytajmy serwer o parę rekordów:



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

```
C:\WINDOWS\system32\cmd.exe
C:\Documents and Settings\Administrator>ping ns1.zsme.local
Badanie ns1.zsme.local [192.168.1.200] z użyciem 32 bajtów danych:
Odpowiedź z 192.168.1.200: bajtów=32 czas<1 ms TTL=64
Odpowiedź z 192.168.1.200: bajtów=32 czas<1 ms TTL=64
Odpowiedź z 192.168.1.200: bajtów=32 czas<1 ms TTL=64
Odpowiedź z 192.168.1.200: bajtów=32 czas<1 ms TTL=64
Statystyka badania ping dla 192.168.1.200:
Pakiety: Wysłane = 4, Odebrane = 4, Utracone = 0 (0% straty),
Szacunkowy czas błędzenia pakietów w milisekundach:
Minimum = 0 ms, Maksimum = 0 ms, Czas średni = 0 ms
C:\Documents and Settings\Administrator>_
```

## 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)

```
C:\WINDOWS\system32\cmd.exe - nslookup
C:\Documents and Settings\Administrator>nslookup
Serwer domyślny: ns1.zsme.local
Address: 192.168.1.200

> server 8.8.8.8
Serwer domyślny: dns.google
Address: 8.8.8.8

> onet.pl
Server: dns.google
Address: 8.8.8.8

Nieautorytatywna odpowiedź:
Nazwa: onet.pl
Address: 213.180.141.140

> _
```

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

**nslookup – 8.8.8.8**

```
zsm@jkubuntu:~$ nslookup - 8.8.8.8
> onet.pl
Server:          8.8.8.8
Address:         8.8.8.8#53

Non-authoritative answer:
Name:   onet.pl
Address: 213.180.141.140
>
```

Czyszczenie pamięci podręcznej serwera DNS:

**rndc flush**

Czyszczenie pamięci podręcznej DNS w Windows:

**ipconfig /flushdns**

## 1. Przygotowanie maszyny wirtualnej z Ubuntu Server 18.04

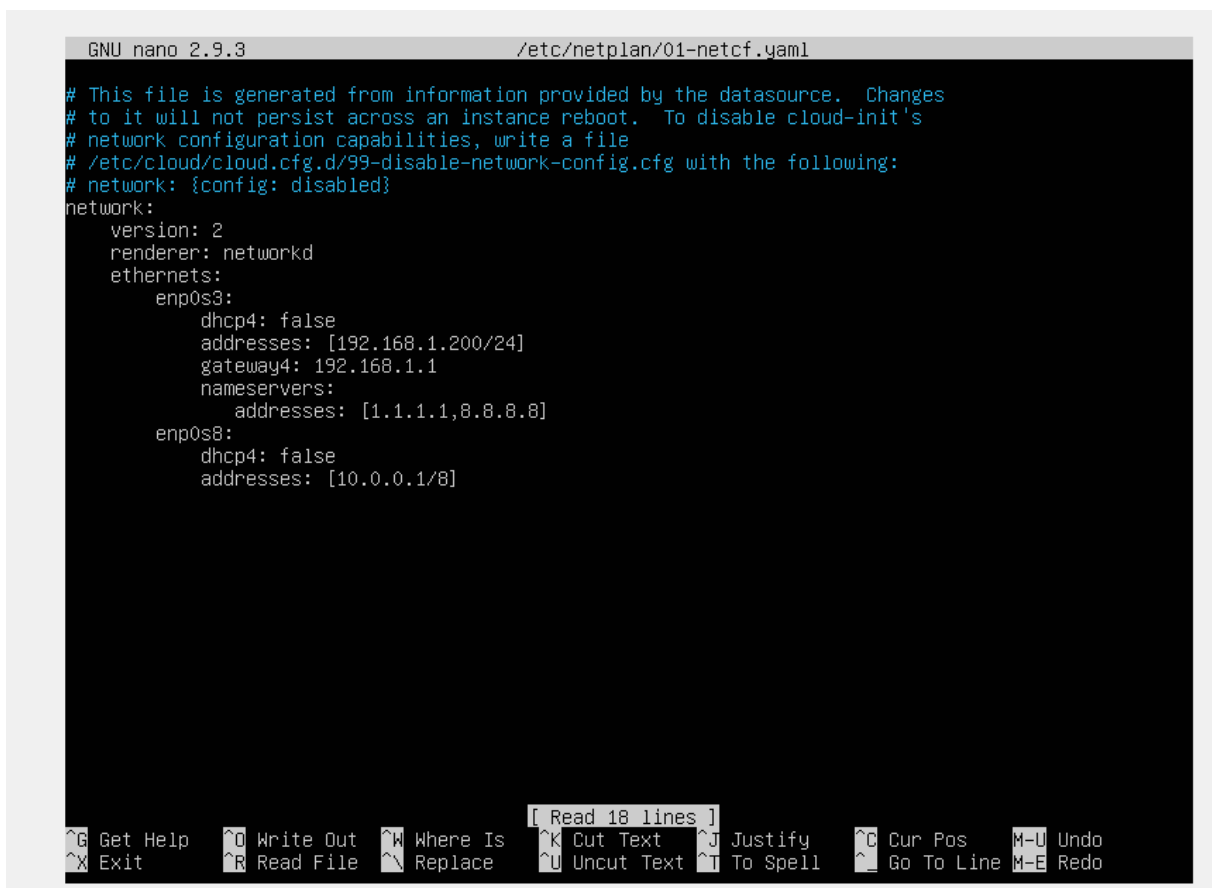
- \* pierwsza karta sieciowa – zmostkowana
- \* druga karta sieciowa – wewnętrzna intnet

jeżeli posiadamy, możemy przywrócić odpowiednią migawkę z kartami już skonfigurowanymi

## 2. Uruchamiamy maszynę i sprawdzamy adres IP na zmostkowanej sieciówce

**nano /etc/netplan/01-netcf.yaml**

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



```
GNU nano 2.9.3 /etc/netplan/01-netcf.yaml
# This file is generated from information provided by the datasource. Changes
# to it will not persist across an instance reboot. To disable cloud-init's
# network configuration capabilities, write a file
# /etc/cloud/cloud.cfg.d/99-disable-network-config.cfg with the following:
# network: {config: disabled}
network:
  version: 2
  renderer: networkd
  ethernets:
    enp0s3:
      dhcp4: false
      addresses: [192.168.1.200/24]
      gateway4: 192.168.1.1
      nameservers:
        addresses: [1.1.1.1,8.8.8.8]
    enp0s8:
      dhcp4: false
      addresses: [10.0.0.1/8]
[ Read 18 lines ]
^G Get Help  ^O Write Out  ^W Where Is  ^K Cut Text   ^J Justify    ^C Cur Pos    M-U Undo
^X Exit      ^R Read File  ^\ Replace   ^U Uncut Text ^T To Spell   ^_ Go To Line  M-E Redo
```

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

## 3. Aktualizujemy zawartość repozytoriów:

sudo apt update

## 4. Instalujemy serwer WWW Apache 2:

`sudo apt install apache2`

Folder gdzie serwer WWW będzie trzymał swoje pliki konfiguracyjne to:

`/etc/apache2`

```
root@jkubuntu:/etc# cd apache2/
root@jkubuntu:/etc/apache2# ls -l
total 80
-rw-r--r-- 1 root root 7224 sie 12 21:33 apache2.conf
drwxr-xr-x 2 root root 4096 lis 19 23:24 conf-available
drwxr-xr-x 2 root root 4096 lis 19 23:24 conf-enabled
-rw-r--r-- 1 root root 1782 lip 16 2019 envvars
-rw-r--r-- 1 root root 31063 lip 16 2019 magic
drwxr-xr-x 2 root root 12288 lis 19 23:24 mods-available
drwxr-xr-x 2 root root 4096 lis 19 23:24 mods-enabled
-rw-r--r-- 1 root root 320 lip 16 2019 ports.conf
drwxr-xr-x 2 root root 4096 lis 19 23:24 sites-available
drwxr-xr-x 2 root root 4096 lis 19 23:24 sites-enabled
root@jkubuntu:/etc/apache2#
```

Dla nas najistotniejsze będą w tym ćwiczeniu:

**apache2.conf** – główny plik konfiguracyjny

**ports.conf** – konfiguracja portów na których działa apache

**sites-available** i **sites-enabled** – konfiguracja hostów wirtualnych i katalog w którym możemy sprawdzić które z nich są włączone

**Apache** jest najczęściej wykorzystywanym serwerem HTTP w sieci Internet. Na jego koncie jest ponad połowa wszystkich aktywnych stron internetowych, a jego atutami są moc i elastyczność.

Apache rozdziela swoje funkcje i komponenty na osobne moduły, które można dostosowywać i konfigurować indywidualnie. **Podstawowym modułem opisującym poszczególną stronę lub domenę jest virtual host.**

Wspomniane moduły pozwalają administratorowi na korzystanie z jednego serwera do obsługi wielu domen lub stron z jednego interfejsu lub IP wykorzystując odpowiedni mechanizm. Jest to bardzo istotne w sytuacji, jeżeli mamy zamiar obsługiwać wiele stron poprzez jeden serwer VPS.

Każda skonfigurowana domena przekierowuje użytkowników do specjalnego katalogu mieszczącego informacje o stronie, nie informując o tym, że ten sam serwer odpowiada też za inne strony.

Domyślnie po zainstalowaniu serwera możemy w przeglądarce odpalić stronę domyślną wpisując adres IP serwera:



Główny katalog dla stron WWW to:

**`/var/www/`**

Katalog w którym przechowywane są pliki tej strony to:

**`/var/www/html`**

Spróbujmy zmodyfikować naszą domyślną stronę:

**`nano /var/www/html/index.html`**

Ja zmieniłem napis u góry:



Wy oczywiście wpiszą swoje Imię i Nazwisko.

## 5. Hosty wirtualne.

Żeby serwer mógł obsłużyć kilka osobnych witryn, musimy stworzyć i skonfigurować odpowiednie **hosty wirtualne**.

**Żeby móc obsłużyć różne hosty wirtualne (czyli więcej niż jedną stronę) muszą się one różnić jedną z poniższych cech:**

- muszą działać na różnych portach
- muszą działać na innym adresie IP
- muszą mieć inną nazwę domenową

Konfiguracja hostów wirtualnych jest przechowywana w:

**/etc/apache2/sites-available**

Domyślnie mamy jednego hosta wirtualnego – stronę domyślną, plik z jej konfiguracją to

**/etc/apache2/sites-available/000-default.conf**

```
GNU nano 2.9.3 000-default.conf
<VirtualHost *:80>
# The ServerName directive sets the request scheme, hostname and port that
# the server uses to identify itself. This is used when creating
# redirection URLs. In the context of virtual hosts, the ServerName
# specifies what hostname must appear in the request's Host: header to
# match this virtual host. For the default virtual host (this file) this
# value is not decisive as it is used as a last resort host regardless.
# However, you must set it for any further virtual host explicitly.
#ServerName www.example.com

2 ServerAdmin webmaster@localhost
  DocumentRoot /var/www/html

# Available loglevels: trace8, ..., trace1, debug, info, notice, warn,
# error, crit, alert, emerg.
# It is also possible to configure the loglevel for particular
# modules, e.g.
#LogLevel info ssl:warn

ErrorLog ${APACHE_LOG_DIR}/error.log
CustomLog ${APACHE_LOG_DIR}/access.log combined

# For most configuration files from conf-available/, which are
# enabled or disabled at a global level, it is possible to
# include a line for only one particular virtual host. For example the
# following line enables the CGI configuration for this host only
# after it has been globally disabled with "a2disconf".
#Include conf-available/serve-cgi-bin.conf
</VirtualHost>

# vim: syntax=apache ts=4 sw=4 sts=4 sr noet
```

Jak widzimy w configu – strona **działa na wszystkich adresach** ustawionych na naszym serwerze Ubuntu, **na porcie 80** (punkt 1), mail do osoby zarządzającej stroną to **webmaster@localhost** a **ścieżka** z plikami strony to **/var/www/html** (punkt 2)

**Cały proces tworzenia kolejnych hostów wirtualnych będzie wyglądał tak:**

1. Tworzymy plik z konfiguracją hosta w katalogu /etc/apache2/sites-available
2. Ustawiamy w configu adres IP, domene, port na którym ma działać nasz host oraz oczywiście ścieżkę do katalogu w którym będą trzymane pliki strony
3. Tworzymy katalog w którym będzie przechowywana nasza strona – taki sam, jak ustawiliśmy w configu
4. Włączamy naszego hosta poleceniem **a2ensite plikzconfigkiem.conf**

### 3.1 Host wirtualny na innym porcie niż 80

a) kopiujemy config strony domyślnej, zapiszmy go pod nazwą port\_test.conf

```
cp /etc/apache2/sites-available/000-default.conf /etc/apache2/sites-available/port_test.conf
```

```
root@jkubuntu:/etc/apache2/sites-available# ls -l
total 16
-rw-r--r-- 1 root root 1332 lip 16 2019 000-default.conf
-rw-r--r-- 1 root root 6338 lip 16 2019 default-ssl.conf
-rw-r--r-- 1 root root 1332 lis 20 00:32 port_test.conf
root@jkubuntu:/etc/apache2/sites-available# _
```

b) modyfikujemy plik konfiguracyjny – zmieniamy port i ścieżkę do katalogu strony

**port – 8080**

**sciezka – /var/www/port\_test/html**

```
GNU nano 2.9.3 port_test.conf Modified
<VirtualHost *:8080>
# The ServerName directive sets the request scheme, hostname and port that
# the server uses to identify itself. This is used when creating
# redirection URLs. In the context of virtual hosts, the ServerName
# specifies what hostname must appear in the request's Host: header to
# match this virtual host. For the default virtual host (this file) this
# value is not decisive as it is used as a last resort host regardless.
# However, you must set it for any further virtual host explicitly.
#ServerName www.example.com

ServerAdmin webmaster@localhost
DocumentRoot /var/www/port_test/html
```

c) tworzymy katalog dla naszej strony tak jak w pliku konfiguracyjnym, potem tworzymy w nim plik index.html:

```
mkdir -p /var/www/port_test/html
```

```
nano /var/www/port_test/html/index.html
```

```
GNU nano 2.9.3 /var/www/port_test/html/index.html Modified
<html>
<head></head>
<body>
<h1>Host wirtualny dzialajacy na porcie 8080</h1>
</body>
</html>
```

d) musimy w ustawieniach Apache dodać port na którym ma działać nasz host wirtualny – 8080

**nano /etc/apache2/ports.conf**

```
GNU nano 2.9.3 /etc/apache2/ports.conf Modified
# If you just change the port or add more ports here, you will likely also
# have to change the VirtualHost statement in
# /etc/apache2/sites-enabled/000-default.conf

Listen 80
Listen 8080

<IfModule ssl_module>
    Listen 443
</IfModule>

<IfModule mod_gnutls.c>
    Listen 443
</IfModule>

# vim: syntax=apache ts=4 sw=4 sts=4 sr noet
```

e) włączamy naszego hosta poleceniem `a2ensite` i podajemy nazwę pliku w którym jest konfiguracja naszego hosta:

**a2ensite port\_test.conf**

```
root@jkubuntu:/etc/apache2/sites-available# a2ensite port_test.conf
Enabling site port_test.
To activate the new configuration, you need to run:
    systemctl reload apache2
```

f) restartujemy usługę Apache i testujemy działanie naszej strony w przeglądarce:

**service apache2 restart**



## Host wirtualny działający na porcie 8080

strona działa poprawnie – w ten oto sposób stworzyliśmy samodzielnie naszego pierwszego hosta wirtualnego na Apache

### 3.2 Host wirtualny na innym adresie IP (wtedy port może zostać 80)

a) kopiujemy configa z naszego poprzedniego hosta i zapisujemy go pod nazwą `ip_test.conf`

```
cp /etc/apache2/sites-available/port_test.conf /etc/apache2/sites-available/ip_test.conf
```

b) modyfikujemy go:

**IP – 192.168.1.222** (oczywiście ustawcie jakies z Waszej puli adresów)

**port – 80**

**katalog /var/www/ip\_test/html**

```
GNU nano 2.9.3 ip_test.conf Modifi
<VirtualHost 192.168.1.222:80>
# The serverName directive sets the request scheme, hostname and port that
# the server uses to identify itself. This is used when creating
# redirection URLs. In the context of virtual hosts, the ServerName
# specifies what hostname must appear in the request's Host: header to
# match this virtual host. For the default virtual host (this file) this
# value is not decisive as it is used as a last resort host regardless.
# However, you must set it for any further virtual host explicitly.
#ServerName www.example.com

ServerAdmin webmaster@localhost
DocumentRoot /var/www/ip_test/html
```

c) tworzymy katalog dla strony, kopiujemy i przerabiamy `index.html` z poprzedniego hosta:

```
mkdir -p /var/www/ip_test/html
```

```
cp /var/www/port_test/html/index.html /var/www/ip_test/html/index.html
```

```
nano /var/www/ip_test/html/index.html
```

```
GNU nano 2.9.3 /var/www/ip_test/html/index.html
<html>
<head></head>
<body>
<h1>Host wirtualny dzialajacy na adresie 192.168.1.222</h1>
</body>
</html>
```

d) jako że witryna ma działać na innym adresie IP musimy ustawić drugi adres IP dla naszej skarty sieciowej:

**nano /etc/netplan/01-netcf.yaml**

```
# Network configuration capabilities, write a file
# /etc/cloud/cloud.cfg.d/99-disable-network-config.cfg with the following:
# network: {config: disabled}
network:
  version: 2
  renderer: networkd
  ethernets:
    enp0s3:
      dhcp4: false
      addresses: [192.168.1.200/24, 192.168.1.222/24]
      gateway4: 192.168.1.1
      nameservers:
        addresses: [1.1.1.1, 8.8.8.8]
    enp0s8:
      dhcp4: false
      addresses: [10.0.0.1/8]
```

Zatwierdzamy oczywiście przy użyciu **netplan apply**

e) aktywujemy naszą stronę poleceniem **a2ensite ip\_test.conf** i restartujemy Apache

```
root@jkubuntu:/etc/apache2/sites-available# a2ensite ip_test.conf
Enabling site ip_test.
To activate the new configuration, you need to run:
  systemctl reload apache2
root@jkubuntu:/etc/apache2/sites-available# service apache2 restart
```

f) sprawdzamy czy nasz host wirtualny działa poprawnie:



### 3.3 Host wirtualny działający po nazwie domenowej

Dodamy teraz dwa hosty działające po nazwie domenowej: **zsme.local** oraz **inazwisko.local** (oczywiście Wy wpiszcie pierwsza literę imienia i swoje nazwisko)

a) kopiujemy config strony domyślnej, zapiszmy go pod nazwą zsme.local.conf oraz inazawisko.local.conf

```
cp /etc/apache2/sites-available/000-default.conf /etc/apache2/sites-available/zsme.local.conf
```

```
cp /etc/apache2/sites-available/000-default.conf /etc/apache2/sites-available/inazwisko.local.conf
```

```
root@jkubuntu:/etc/apache2/sites-available# ls -l
total 28
-rw-r--r-- 1 root root 1332 lip 16 2019 000-default.conf
-rw-r--r-- 1 root root 6338 lip 16 2019 default-ssl.conf
-rw-r--r-- 1 root root 1332 lis 20 01:16 inazwisko.local.conf
-rw-r--r-- 1 root root 1352 lis 20 01:00 ip_test.conf
-rw-r--r-- 1 root root 1344 lis 20 00:40 port_test.conf
-rw-r--r-- 1 root root 1332 lis 20 01:15 zsme.local.conf
root@jkubuntu:/etc/apache2/sites-available# _
```

b) zmieniamy configi:

#### zsme.local.conf:

```
IP dowolne, port 80
ServerAdmin admin@zsme.local
ServerName zsme.local
ServerAlias www.zsme.local
DocumentRoot /var/www/zsme.local/html
```

```
GNU nano 2.9.3 zsme.local.conf
<VirtualHost *:80>
# The ServerName directive sets the request scheme, hostname and port that
# the server uses to identify itself. This is used when creating
# redirection URLs. In the context of virtual hosts, the ServerName
# specifies what hostname must appear in the request's Host: header to
# match this virtual host. For the default virtual host (this file) this
# value is not decisive as it is used as a last resort host regardless.
# However, you must set it for any further virtual host explicitly.
#ServerName www.example.com

ServerAdmin admin@zsme.local
ServerName zsme.local
ServerAlias www.zsme.local
DocumentRoot /var/www/zsme.local/html

# Available options:
#
```

## inazwisko.local.conf

IP dowolne, port 80  
ServerAdmin admin@inazwisko.local  
ServerName inazwisko.local  
ServerAlias www.inazwisko.local  
DocumentRoot /var/www/inazwisko.local/html

c) tworzymy katalogi dla naszych hostów, kopiujemy z poprzednich hostów index.html

```
mkdir -p /var/www/zsme.local/html
```

```
mkdir -p /var/www/inazwisko.local/html
```

```
cp /var/www/port_test/html/index.html /var/www/zsme.local/html/index.html
```

```
cp /var/www/port_test/html/index.html /var/www/inazwisko.local/html/index.html
```

Modyfikujemy ich zawartość:

```
GNU nano 2.9.3 /var/www/zsme.local/html/index.html Modified
<html>
<head></head>
<body>
<h1>Host wirtualny zsme.local</h1>
</body>
</html>
```

oraz:

```
GNU nano 2.9.3 /var/www/inazwisko.local/html/index.html Modified
<html>
<head></head>
<body>
<h1>Host wirtualny inazwisko.local</h1>
</body>
</html>
```

c) włączamy nasze strony i restartujemy Apache:

```
a2ensite zsme.local.conf
```

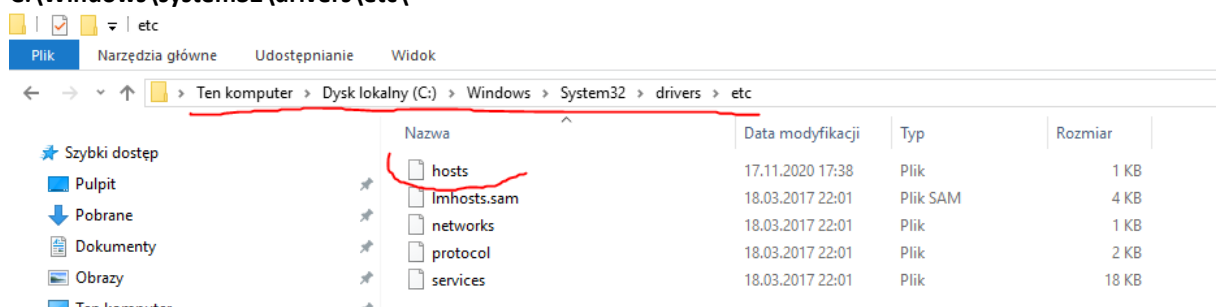
```
a2ensite inazwisko.local.conf
```

```
service apache2 restart
```

d) żeby nasz komputer – gospodarz - z którego testujemy działanie stron umiał się połączyć z naszymi stronami zsme.local i inazwisko.local, musi potrafić zamienić te nazwy domenowe na adres IP serwera obsługującego nasze hosty wirtualne. Potrzebujemy albo serwera DNS ze skonfigurowanymi strefami z tymi domenami – co jest dosyć pracochłonne. Żeby nie męczyć się z tym, **użyjemy pliku hosts**

Plik host w Windows znajduje się w katalogu:

**C:\Windows\system32\drivers\etc\**

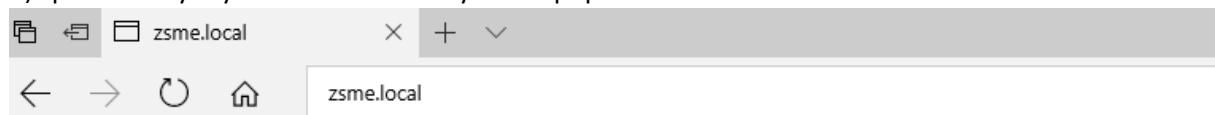


Musimy go edytować z uprawnieniami administratora o inaczej nie pozwoli się zapisać po modyfikacji (najlepiej użyć np. Notepad++)

Dodamy do niego dwa rekordy łączące nazwy domenowe naszych stron z adresem IP naszego serwera na którym pracuje Apache obsługujący te strony.

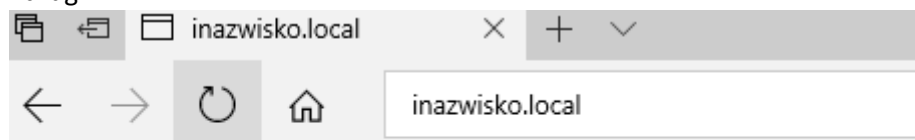
```
11 #.Additionally, .comments .(such .as .these) .may .be .inserted .on .individual
12 #.lines .or .following .the .machine .name .denoted .by .a .'#' .symbol.
13 #
14 #.For .example:
15 #
16 #.....102.54.94.97.....rhino.acme.com.....#.source .server
17 #.....38.25.63.10.....x.acme.com.....#.x .client .host
18
19 #.localhost .name .resolution .is .handled .within .DNS .itself.
20 #→127.0.0.1.....localhost
21 #→:1.....localhost
22 #→192.168.1.200→zsme.local
23 #→192.168.1.200→inazwisko.local
24
```

e) sprawdzamy czy nasz host wirtualny działa poprawnie:



## Host wirtualny zsme.local

i drugi:



## Host wirtualny inazwisko.local

Uwagi:

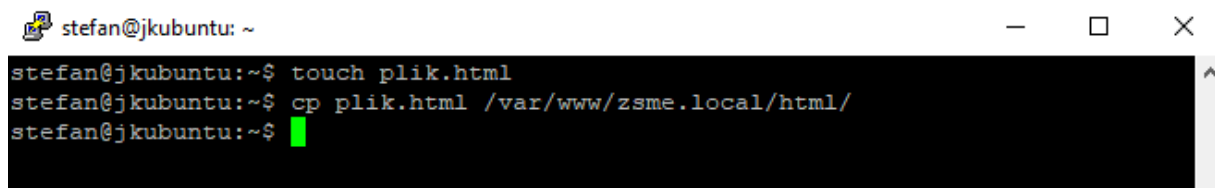
Jeżeli chcielibyśmy, żeby właścicielem strony był ktoś inny niż root (czyli żeby mógł wrzucać pliki do katalogu strony), musimy oczywiście zmienić właściciela katalogu z daną stroną oraz ustawić mu prawa dostępu do niego, np. ustawię, że właścicielem witryny zsme.local będzie użytkownik stefan (należy sobie utworzyć takie konto):

a) zmieniam właściciela katalogu z witryną na stefana oraz ustawiam 775 dla całego katalogu /var/www, żeby mieć pewność, że będzie możliwość odczytu

```
sudo chown -R stefan:stefan /var/www/zsme.local/html
sudo chmod -R 755 /var/www
```

```
root@jkubuntu:/var/www/zsme.local# ls -l
total 4
drwxr-xr-x 2 stefan stefan 4096 lis 20 01:34 html
root@jkubuntu:/var/www/zsme.local#
```

b) przeloguję się na konto stefana i sprawdzam czy może coś wrzucić do katalogu ze stroną

A terminal window titled 'stefan@jkubuntu: ~' with standard window controls (minimize, maximize, close) in the top right corner. The terminal content shows three lines of text: 'stefan@jkubuntu:~\$ touch plik.html', 'stefan@jkubuntu:~\$ cp plik.html /var/www/zsme.local/html/', and 'stefan@jkubuntu:~\$' followed by a green cursor. The terminal background is black, and the text is white.

```
stefan@jkubuntu:~$ touch plik.html
stefan@jkubuntu:~$ cp plik.html /var/www/zsme.local/html/
stefan@jkubuntu:~$
```

Jak widać, stefan może zapisywać w katalogu domowym strony zsme.local