

T: Konfiguracja usługi HTTP (Hypertext Transfer Protocol) w systemie Linux.

Zadanie1:

Odszukaj w serwisie internetowym Wikipedii informacje na temat usługi http oraz oprogramowania Apache.

<https://hostovita.pl/blog/konfiguracja-apache-virtualhost-w-ubuntu-16-04/>

HTTP (ang. Hypertext Transfer Protocol) – protokół przesyłania dokumentów hipertekstowych w układzie klient-serwer. Do komunikacji wykorzystywany jest domyślnie port 80 protokołu TCP.

Szyfrowaną wersją protokołu HTTP jest HTTPS wykorzystujący domyślnie port 443 protokołu TCP.

Zadanie2:

Sprawdź dostępną w systemie pomoc na temat serwera http (man apache2).

W celu odzyskania informacji o zainstalowanych programach usługi ftp należy uruchomić YaST i przejść do Zarządzania oprogramowaniem lub w konsoli tekstowej wydać polecenie:

```
rpm -qa | grep apache  
rpm -qa | grep http
```

Do połączenia się ze zdalnym serwerem możemy wykorzystać klienta konsoli tekstowej wydając polecenie:

```
lynx serwer.domena.pl  
links 127.0.0.1
```

W trybie graficznym do połączenia się ze zdalnym serwerem możemy wykorzystać programy z kategorii przeglądarki internetowe wpisując następującą składnię w polu adresu:

```
http://username:password@serwer.domena.pl/path/filename:port
```

Serwer www domyślnie może wykorzystywać dwa porty 80 oraz 443 protokołu TCP. Stan portów w lokalnym komputerze sprawdzimy poleceniem:

```
netstat -ant
```

Stan serwera www możemy sprawdzić i zmienić poleceniami:

```
/etc/init.d/apache2 status  
service apache2 restart  
chkconfig apache2 on
```

Zadanie3:

Sprawdź zainstalowane oprogramowanie serwera http, a w razie potrzeby przeprowadź instalację oprogramowania np. apache2. Odszukaj konfigurację serwera http w centrum sterowania YaST.

Najważniejsze ustawienia serwera apache2 pod systemem SUSE:

- domyślne zasoby serwera (strony www) - /var/www/html (/srv/www/htdocs)
- pliki konfiguracyjne - /etc/apache2/

```
– httpd.conf  
DirectoryIndex index.html index.htm.var  
<Directory />  
    Options None  
    AllowOverride None  
    Order deny,allow  
    Deny from all  
</Directory>  
– default-server.conf  
DocumentRoot /srv/www/host1_www/  
ServerName www.dom.pl  
ServerAdmin root@dom.pl  
<Directory /srv/www/htdocs/>
```

```

    AllowOverride None
    Order allow,deny
    Allow from all
    DirectoryIndex index.htm index.html
</Directory>
<Directory /srv/www/host1_www/>
    AllowOverride None
    Order allow,deny
    Allow from all
    DirectoryIndex default.htm
</Directory>
UserDir public_html
<Directory /home/*/public_html/>
    AllowOverride None
    Order allow,deny
    Allow from all
    DirectoryIndex index.htm index.html strona.htm
</Directory>

```

- **listen.conf**

```

#Listen 12.34.56.78:80
#Listen 443
Listen 80
#NameVirtualHost 12.34.56.78:80
#NameVirtualHost *:80

```

- z serwerem łączymy się pod adresem <http://localhost> lub <http://192.168.10.200>
- w katalogu usera **public_html** tworzymy stronę testową i łączymy się z nią podając adres <http://localhost/~user>

Opis najważniejszych ustawień

```

<directory nazwa_katalogu> polecenia </directory>
Options opcja1 opcja2 opcja3          #włącza i wyłącza opcje
FollowSymLinks                        #zezwozenie na wykonywanie łącz symbolicznych
SymLinksIfOwnerMatch                  #zezwozenie na wykonywanie łącz symbolicznych tylko wtedy,
                                        gdy właściciel łączy taki zam jak cel
ExecCGI                               #włączenie wykonywania skryptów CGI
Includes                              #zezwozenie na używanie włączeń od strony serwera
Index                                 #możliwość przesłania listy plików danego katalogu
IncludesNoExec                        #Includes, ale zablokowanie skryptów CGI
AllowOverride co                       #określa dyrektywy, które mogą być zastąpione ustawieniami
                                        z rozproszonej struktury plików
AuthRealm obszar                       #obszar autoryzacji dla katalogu, nazwa użytkownika i hasło
AuthType typ                           #
AuthUserFile ścieżka_bezwzględna       #do pliku haseł httpd, tworzony plik przez htpasswd
AuthGroupFile ścieżka_bezwzględna     #do grup programu httpd, grupa: user1 user2 user3
Limit metody_ograniczenia              #ograniczenia dla katalogu
Order porządek                          #warunek allow i deny, np. Order deny, allow
Allow from host1 host2                  #hosty mające dostęp, IP, .dom.pl
Deny from host3 host4                   #hosty zablokowane, komp.dom.pl
Require user u1 u2                      #dopuszczenie konkretnych użytkowników
Require group g1 g2                     #dopuszczenie konkretnych grup
Require valid-user                      #każdy dopuszczony

```

htpasswd -c /usr/local/etc/httpd/pw/auth beata

#założenie pliku haseł z użytkownikiem

htpasswd /usr/local/etc/httpd/pw/auth leszek

#dodanie użytkownika

Zadanie4:

Przeanalizuj zawartość pliku /etc/apache2/httpd.conf.

W dowolnym katalogu udostępnianym przez serwer Apache można umieścić indywidualny plik o nazwie .htaccess kontrolujący dostęp do zawartości danego katalogu. Przykładowy plik udostępniający dany katalog po poprawnej identyfikacji:

```
AuthType Basic
AuthName nazwa_strony
AuthUserFile /home/user_name/.htpasswd
require user_www
```

Aby utworzyć hasło dla danego użytkownika wydajemy następujące polecenie:

```
htpasswd -c /home/user_name/.htpasswd user_www
```

CGI (Common Gateway Interface) służy do łączenia zewnętrznych programów (np. liczniki, obsługa baz danych) ze stroną www. Pliki cgi powinny być zapisane w określonym katalogu i posiadać atrybut wykonywalności (chmod 755 plik.cgi). Należy pamiętać o zagrożeniach związanych z obsługą CGI. Dostęp do plików można uzyskać pod adresem http://serwer_apache/cgi-bin/plik.cgi.

Konfiguracja serwera Apache pod kątem obsługi CGI:

```
LoadModule cgi_module modules/mod_cgi.so
AddModule mod_cgi.c
AddHandler cgi-script .cgi
AddHandler cgi-script .pl
ScriptAlias /cgi-bin/ "/srv/http/cgi-bin"
<Directory "/srv/http/cgi-bin/">
    AllowOverride None
    Options ExecCGI
    Order allow, deny
    Allow from all
</Directory>
```

PHP (Personal Home Page) to język skryptowy polegający na wykonywaniu poleceń przez serwer i wysłaniu wyników w formie statycznej strony www po stronie klienta. Dla obsługi PHP również należy odpowiednio skonfigurować serwer Apache.

Zadanie5:

Skonfiguruj na swoim stanowisku komputerowym usługę http w systemie Linux tak, aby po połączeniu z nią z dowolnego komputera z sieci lokalnej poprzez protokół http do podkatalogu asso wyświetlił się domyślnie plik o nazwie asso.htm o następującej treści:

```
<html>
<body>
<p>
<center>
<h1>Nie ma jak<br>asso</h1>
</center>
</p>
</body>
</html>
```

Zadanie6:

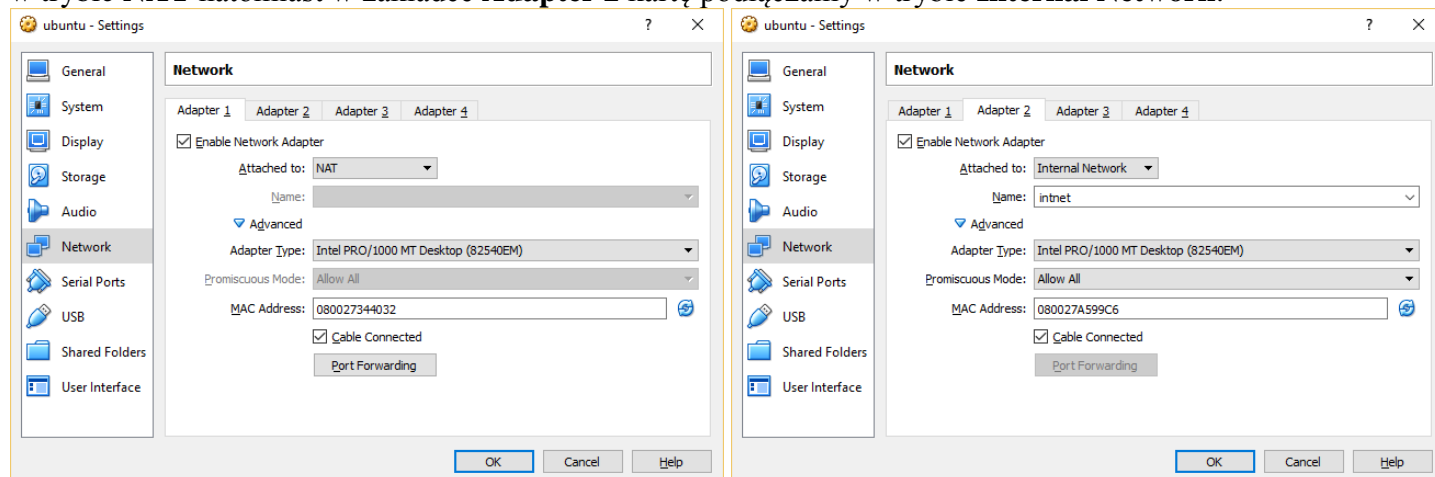
Wykonaj i przeanalizuj poniższe polecenia:

Konfiguracja serwera Apache w systemie Linux Ubuntu:

```
apt-get update
apt-get install apache2
/etc/init.d/apache2 restart
service apache2 restart
/var/www/html/index.html
/etc/apache2/apache2.conf
/etc/apache2/ports.conf
    Listen 0.0.0.0:80
    NameVirtualHost *:80
/etc/apache2/sites-available/000-default.conf
```

```
ServerName localhost
mcedit ./site-available/namehost.conf
ServerName namehost
a2ensite namehost.conf
/etc/hosts
192.168.0.1 namehost
```

Przed przystąpieniem do instalacji usługi HTTP należy sprawdzić, czy w maszynie wirtualnej systemu Linux włączyliśmy kartę sieciową i odpowiednio ją skonfigurowaliśmy. I tak dla maszyny wirtualnej **ubuntu** wybieramy ustawienia (**Settings**), przechodzimy do kategorii **Network** i w zakładce **Adapter 1** kartę podłączamy w trybie **NAT** natomiast w zakładce **Adapter 2** kartę podłączamy w trybie **Internal Network**:



W pierwszej kolejności uruchamiamy konsolę tekstową i logujemy się na konto administratora systemu podając polecenie **sudo su -**, następnie aktualizujemy adresy serwerów źródłowych oprogramowania poleceniem **apt-get update**:

```

root@ubuntu: ~
Ign. cdrom://Ubuntu 14.04.1 LTS _Trusty Tahr_ - Release i386 (20140722.2) trusty/restr
icted Translation-pl
Ign. cdrom://Ubuntu 14.04.1 LTS _Trusty Tahr_ - Release i386 (20140722.2) trusty/restr
icted Translation-en
Ign. http://archive.ubuntu.com trusty InRelease
Pobieranie:1 http://archive.ubuntu.com trusty-updates InRelease [65,9 kB]
Pobieranie:2 http://security.ubuntu.com trusty-security InRelease [65,9 kB]
Stary http://archive.ubuntu.com trusty Release.gpg
Pobieranie:3 http://archive.ubuntu.com trusty-updates/main i386 Packages [924 kB]
Pobieranie:4 http://security.ubuntu.com trusty-security/main i386 Packages [554 kB]
Pobieranie:5 http://archive.ubuntu.com trusty-updates/restricted i386 Packages [16,2 kB
]
Pobieranie:6 http://archive.ubuntu.com trusty-updates/main Translation-en [474 kB]
Pobieranie:7 http://archive.ubuntu.com trusty-updates/restricted Translation-en [3847 B
]
Stary http://archive.ubuntu.com trusty Release
Stary http://archive.ubuntu.com trusty/main i386 Packages
Stary http://archive.ubuntu.com trusty/restricted i386 Packages
Pobieranie:8 http://archive.ubuntu.com trusty/main Translation-pl [315 kB]
Stary http://archive.ubuntu.com trusty/main Translation-en
Pobieranie:9 http://archive.ubuntu.com trusty/restricted Translation-pl [2538 B]
Stary http://archive.ubuntu.com trusty/restricted Translation-en
Pobieranie:10 http://security.ubuntu.com trusty-security/restricted i386 Packages [13,1
kB]
Pobieranie:11 http://security.ubuntu.com trusty-security/main Translation-en [325 kB]
Ign. http://archive.ubuntu.com trusty/main Translation-pl_PL
Ign. http://archive.ubuntu.com trusty/restricted Translation-pl_PL
Pobieranie:12 http://security.ubuntu.com trusty-security/restricted Translation-en [335
7 B]
Pobrano 2762 kB w 2s (1097 kB/s)
Czytanie list pakietów... Gotowe
root@ubuntu:~# apt-get update

```

Możemy potem przystąpić do instalacji oprogramowania wydając polecenie **apt-get install apache2**:

```

root@ubuntu: ~
root@ubuntu:~# apt-get install apache2
Czytanie list pakietów... Gotowe
Budowanie drzewa zależności
Odczyt informacji o stanie... Gotowe
Zostaną zainstalowane następujące dodatkowe pakiety:
 apache2-bin apache2-data libapr1 libaprutil1 libaprutil1-dbd-sqlite3
 libaprutil1-ldap
Sugerowane pakiety:
 apache2-doc apache2-suexec-pristine apache2-suexec-custom apache2-utils
Zostaną zainstalowane następujące NOWE pakiety:
 apache2 apache2-bin apache2-data libapr1 libaprutil1 libaprutil1-dbd-sqlite3
 libaprutil1-ldap
0 aktualizowanych, 7 nowo instalowanych, 0 usuwanych i 708 nieaktualizowanych.
Konieczne pobranie 1255 kB archiwów.
Po tej operacji zostanie dodatkowo użyte 4952 kB miejsca na dysku.
Kontynuować? [T/n] t

```

Możemy upewnić się, czy oprogramowanie zostało zainstalowane w systemie wydając polecenie **dpkg -l | grep apache**. Dodatkową warto sprawdzić domyślny stan usługi w systemie oraz nasłuchiwanie na porcie 80:

```

root@ubuntu:~# dpkg -l | grep apache
ii apache2 2.4.7-1ubuntu4.13
ii apache2-bin 1386 Apache HTTP Server 2.4.7-1ubuntu4.13
ii apache2-data 1386 Apache HTTP Server (binary files and modules) 2.4.7-1ubuntu4.13
root@ubuntu:~# netstat -ant | grep :80
tcp6 0 ::::80 :::* LISTEN
root@ubuntu:~# /etc/init.d/apache2 status
* apache2 is running
root@ubuntu:~# service apache2 status
* apache2 is running
root@ubuntu:~#

```

Na podstawie informacji uzyskanych wcześniej zauważyliśmy, że serwer Apache uruchamiany jest automatycznie. Możemy więc sprawdzić udostępnianą domyślną witrynę www:

```

/etc/apache2/
-- apache2.conf
-- ports.conf
-- mods-enabled
| -- *.load
| -- *.conf
-- conf-enabled
-- *.conf
-- sites-enabled

```

Domyślna witryna www udostępniana jest w folderze **/var/www**, natomiast katalog z plikami konfiguracyjnymi znajduje się w folderze **/etc/apache2**:

```

root@ubuntu:~# ls /var/www/
html
root@ubuntu:~# ls /var/www/html/
index.html
root@ubuntu:~# ls /etc/apache2/
apache2.conf  conf-enabled  magic  mods-enabled  sites-available  sites-enabled
conf-available  envvars  mods-available  ports.conf
root@ubuntu:~# ls -l /etc/apache2/
razem 48
-rw-r--r-- 1 root root 7115 sty 7 2014 apache2.conf
drwxr-xr-x 2 root root 140 mar 17 11:56 conf-available
drwxr-xr-x 2 root root 140 mar 17 11:56 conf-enabled
-rw-r--r-- 1 root root 1782 sty 3 2014 envvars
-rw-r--r-- 1 root root 31063 sty 3 2014 magic
drwxr-xr-x 2 root root 2600 mar 17 11:56 mods-available
drwxr-xr-x 2 root root 580 mar 17 11:56 mods-enabled
-rw-r--r-- 1 root root 320 sty 7 2014 ports.conf
drwxr-xr-x 2 root root 80 mar 17 11:56 sites-available
drwxr-xr-x 2 root root 60 mar 17 11:56 sites-enabled
root@ubuntu:~#

```

Główny plik konfiguracyjny serwera:

```
root@ubuntu:/etc/apache2 root@ubuntu:/etc/apache2 12:46
root@ubuntu:/etc/apache2# cat ./apache2.conf
# This is the main Apache server configuration file. It contains the
# configuration directives that give the server its instructions.
# See http://httpd.apache.org/docs/2.4/ for detailed information about
# the directives and /usr/share/doc/apache2/README.Debian about Debian specific
# hints.
#
# Summary of how the Apache 2 configuration works in Debian:
# The Apache 2 web server configuration in Debian is quite different to
# upstream's suggested way to configure the web server. This is because Debian's
# default Apache2 installation attempts to make adding and removing modules,
# virtual hosts, and extra configuration directives as flexible as possible, in
# order to make automating the changes and administering the server as easy as
# possible.
#
# It is split into several files forming the configuration hierarchy outlined
# below, all located in the /etc/apache2/ directory:
#
#       /etc/apache2/
#       |-- apache2.conf
#           |-- ports.conf
#       |-- mods-enabled
#           |-- *.load
#           |-- *.conf
#       |-- conf-enabled
#           |-- *.conf
#       |-- sites-enabled
#           |-- *.conf
#
# * apache2.conf is the main configuration file (this file). It puts the pieces
```

Włączenie nasłuchiwanie na wszystkich urządzeniach sieciowych w systemie na protokole IPv4:

```
root@ubuntu:/etc/apache2 root@ubuntu:/etc/apache2 12:50
root@ubuntu:/etc/apache2# cat ./ports.conf
# 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

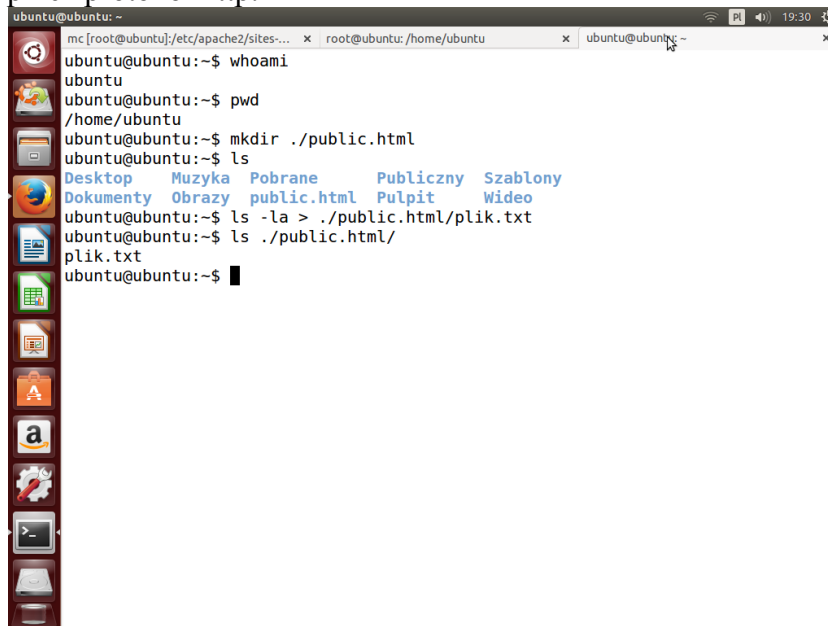
#Listen 80
Listen 0.0.0.0:80

<IfModule ssl_module>
    Listen 443
</IfModule>

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

# vim: syntax=apache ts=4 sw=4 sts=4 sr noet
root@ubuntu:/etc/apache2# service apache2 restart
* Restarting web server apache2
AH00558: apache2: Could not reliably determine the server's fully qualified domain name
, using 127.0.1.1. Set the 'ServerName' directive globally to suppress this message
[ OK ]
root@ubuntu:/etc/apache2# netstat -anp | grep :80
tcp        0      0 0.0.0.0:80          0.0.0.0:*          LISTEN    7407/ap
ache2
root@ubuntu:/etc/apache2#
```

W celu udostępnienia przez każdego użytkownika własnej strony w katalogu **public_html** należy wykonać następujące czynności: każdy z użytkowników tworzy w katalogu domowym katalog **public_html** i umieszcza w nim pliki do publikacji przez protokół **http**:



```
ubuntu@ubuntu: ~  
mc [root@ubuntu:]/etc/apache2/sites-... x root@ubuntu: /home/ubuntu x ubuntu@ubuntu: ~  
ubuntu@ubuntu:~$ whoami  
ubuntu  
ubuntu@ubuntu:~$ pwd  
/home/ubuntu  
ubuntu@ubuntu:~$ mkdir ./public.html  
ubuntu@ubuntu:~$ ls  
Desktop Muzyka Pobrane Publiczny Szablony  
Dokumenty Obrazy public.html Pulpit Wideo  
ubuntu@ubuntu:~$ ls -la > ./public.html/plik.txt  
ubuntu@ubuntu:~$ ls ./public.html/  
plik.txt  
ubuntu@ubuntu:~$ █
```

Następnie należy włączyć moduł **userdir.conf** serwera Apache odpowiedzialny za możliwość udostępniania stron domowych użytkowników oraz dla zastosowania wprowadzonych zmian należy zrestartować serwer:



```
root@ubuntu: /etc/apache2  
mc [root@ubuntu:]/etc/apache2/mods-... x root@ubuntu: /etc/apache2 x ubuntu@ubuntu: ~  
root@ubuntu:/etc/apache2# ln -s /etc/apache2/mods-available/userdir.conf /etc/a  
pache2/mods-enabled/userdir.conf  
root@ubuntu:/etc/apache2# ln -s /etc/apache2/mods-available/userdir.load /etc/a  
pache2/mods-enabled/userdir.load  
root@ubuntu:/etc/apache2# service apache2 restart  
* Restarting web server apache2  
AH00558: apache2: Could not reliably determine the server's fully qualified dom  
ain name, using 127.0.1.1. Set the 'ServerName' directive globally to suppress  
this message  
[ OK ]  
root@ubuntu: /etc/apache2# █
```

Strona domowa wywoływana jest adresem hosta z dopisaną nazwą użytkownika i przedstawia się następująco:

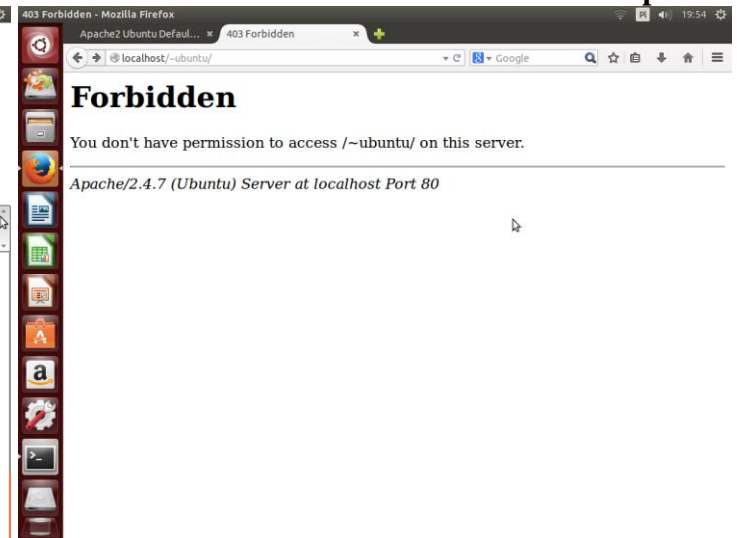


```
Index of /~ubuntu - Mozilla Firefox  
Apache2 Ubuntu Defaul... x Index of /~ubuntu x +  
localhost/~ubuntu/  
Name Last modified Size Description  
Parent Directory -  
plik.txt 2017-03-21 19:19 1.4K  
Apache/2.4.7 (Ubuntu) Server at localhost Port 80
```


W celu zablokowania przeglądania zawartości katalogu stron domowych użytkowników należy zmodyfikować plik opisujący moduł **userdir.conf** usuwając ustawienie **Indexes** w ustawieniach **AllowOverride** oraz **Options**:

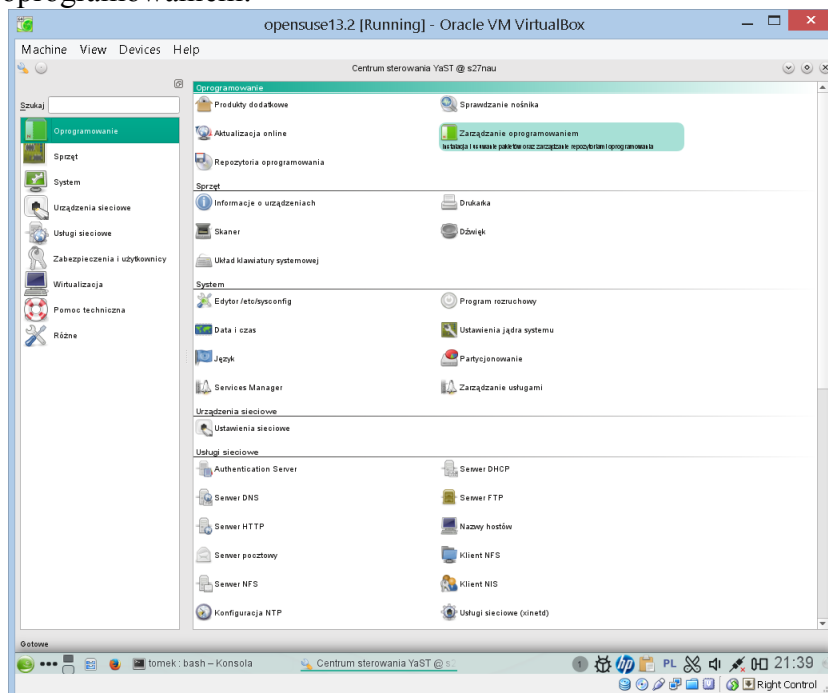
```
root@ubuntu: /etc/apache2/mods-enabled# pwd
/etc/apache2/mods-enabled
root@ubuntu: /etc/apache2/mods-enabled# ln -s ../mods-available/userdir.conf
root@ubuntu: /etc/apache2/mods-enabled# ln -s ../mods-available/userdir.load
root@ubuntu: /etc/apache2/mods-enabled# cat ./userdir.conf
LoadModule userdir_module mod_userdir.c

```

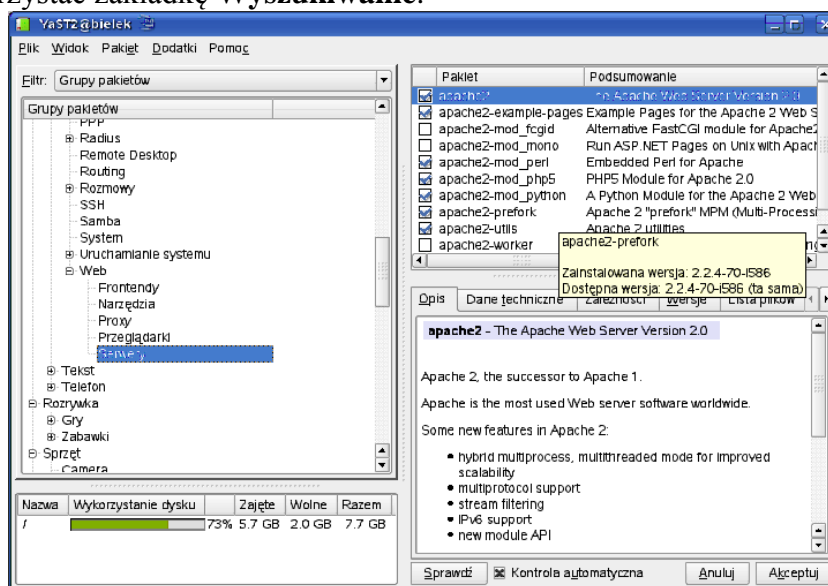


Instalacja i konfiguracja usługi http w systemie Linux OpenSUSE

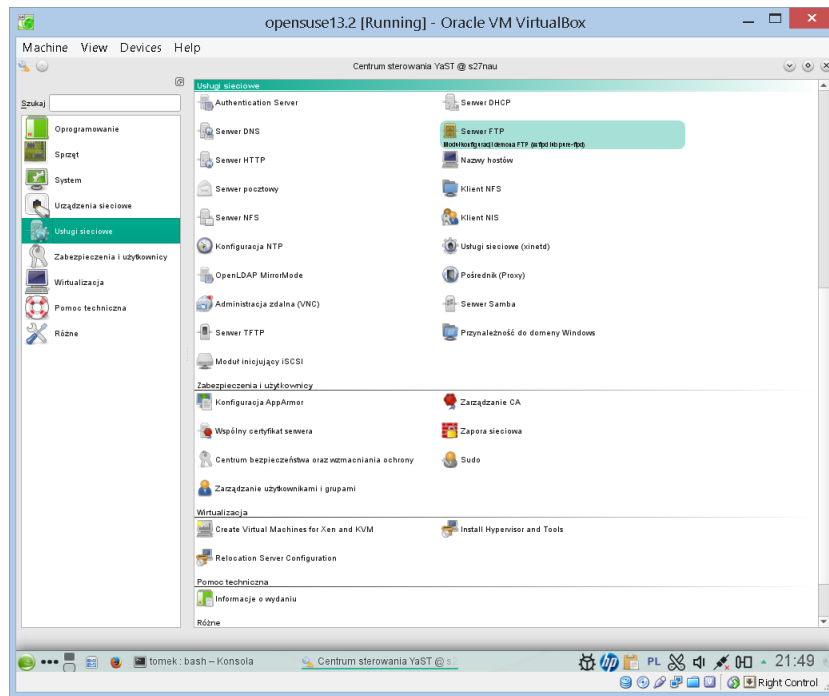
W systemie operacyjnym Linux OpenSUSE uruchamiamy konsolę Centrum zarządzania YaST rozwijając menu Start KDE => System => YaST. Do zarządzania systemem uprawnienia posiada administrator systemu, dlatego należy w oknie podać hasło użytkownika root. W oknie programu YaST wybieramy w kategorii Oprogramowanie narzędzie Zarządzanie oprogramowaniem:



Sprawdzenie, czy w systemie jest zainstalowane oprogramowanie serwera http o nazwie **apache**. Do odszukania pakietów można wykorzystać zakładkę **Wyszukiwanie**:



Po zainstalowaniu serwera http w Centrum sterowania YaST znajdziemy w kategorii Usługi sieciowe narzędzie do zarządzania usługą w trybie graficznym o nazwie **Serwer HTTP**:



W celu sprawdzenia bieżącego stanu usługi http należy w konsoli tekstowej wydać poniższe polecenia: