

**P O L I T E C H N I K A   W R O C Ł A W S K A**

**W Y D Z I A Ł   E L E K T R O N I K I**

**Katedra Systemów i Sieci  
Komputerowych**

**T E C H N O L O G I E   S I E C I O W E   2**

**Zastosowanie platformy Banana Pi jako serwer usług**

## Ćwiczenie 6: Platforma Banana Pi jako serwer usług

### Cel ćwiczenia

Celem laboratorium jest stwierdzenie, czy platforma serwerowa zestawiona w oparciu o minikomputer Banana Pi może sprawdzić się jako serwer w różnych zastosowaniach (np. biznesowych). Przeprowadzenie testów w kierunku sprawdzenia wydajności platformy jako serwera WWW.

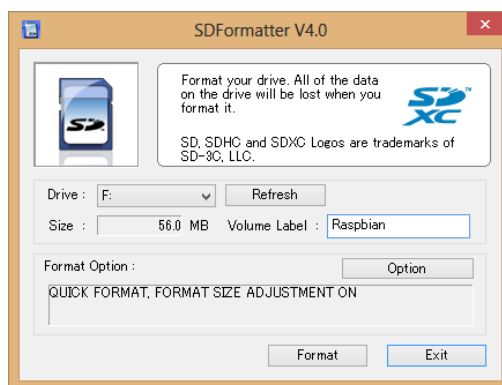
### Wprowadzenie

Elektronikę uważa się za jedną z najbardziej dynamicznie rozwijających się dziedzin nauki. W trend tego niesamowitego postępu i miniaturyzacji wpisują się minikontrolery oraz minikomputery. Minikomputer Banana Pi jest rozwinięciem idei zaproponowanej przez platformę Raspberry Pi - posiadają one wiele cech wspólnych, jednakże Banana Pi posiada zdecydowanie wydajniejszą konfigurację pod względem procesora, pamięci czy karty Ethernet.

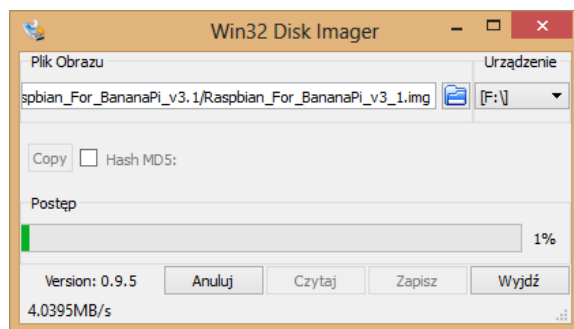
### Przygotowanie platformy

#### 1. Instalacja systemu (opcjonalne)

Do funkcjonowania platformy niezbędne jest wyposażenie jej w system operacyjny. Za pomocą oprogramowania SD Formatter należy sformatować kartę pamięci tak, aby można było bezproblemowo wgrać obraz systemu.

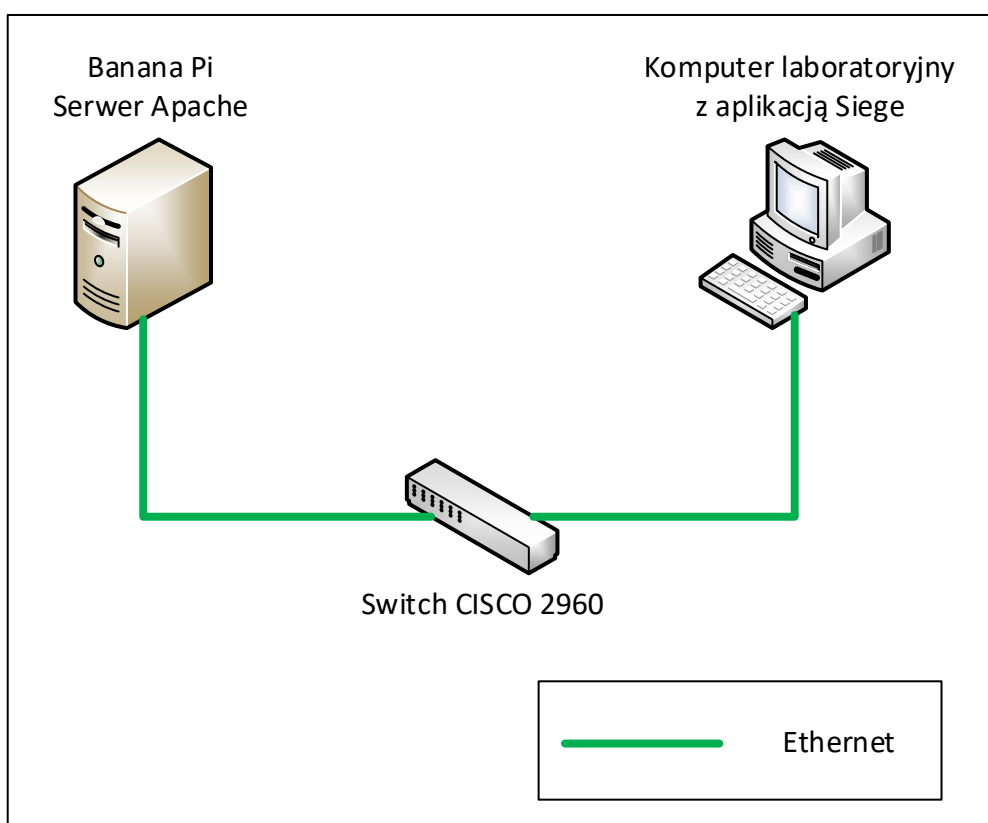


Następnie, wykorzystując program Win32DiskImager, wgrywamy wybrany system na kartę pamięci (opcja „zapis”). Jako system operacyjny wybrano Raspbian w najnowszej stabilnej wersji (do pobrania ze strony producenta – <http://www.lemaker.org/product-bananapi-resource.html> )



## 2. Topologia sieciowa

Poniżej przedstawiono topologię sieciową, w której należy zestawić urządzenia aby móc wykonać zadanie laboratoryjne.



## 3. Aktualizacja pakietów systemu Debian (Raspbian) – polecenie **apt-get update**

```
192.168.1.9 - PuTTY
login as: root
root@192.168.1.9's password:
Linux lemaker 3.4.103 #1 SMP PREEMPT Thu Dec 18 13:07:12 CST 2014 armv7l

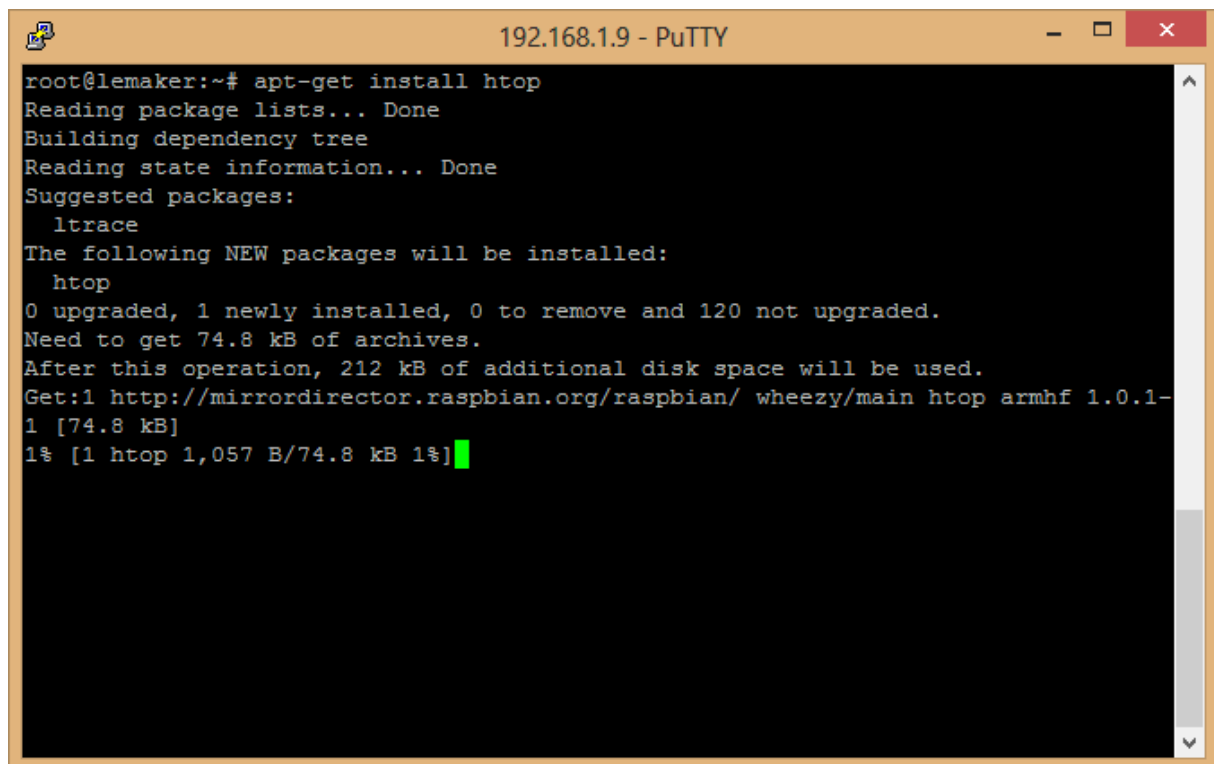
The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
root@lemaker:~# htop
-bash: htop: command not found
root@lemaker:~# apt-get update
Get:1 http://archive.raspberrypi.org wheezy Release.gpg [473 B]
Get:2 http://raspberrypi.collabora.com wheezy Release.gpg [836 B]
Get:3 http://mirrordirector.raspbian.org wheezy Release.gpg [490 B]
Get:4 http://raspberrypi.collabora.com wheezy Release [7,493 B]
Get:5 http://archive.raspberrypi.org wheezy Release [17.6 kB]
Get:6 http://mirrordirector.raspbian.org wheezy Release [14.4 kB]
Get:7 http://raspberrypi.collabora.com wheezy/rpi armhf Packages [2,214 B]
Get:8 http://archive.raspberrypi.org wheezy/main armhf Packages [131 kB]
Get:9 http://mirrordirector.raspbian.org wheezy/main armhf Packages [6,904 kB]
Hit http://repository.wolfram.com stable Release.gpg
Hit http://repository.wolfram.com stable Release
```

#### 4. Instalacja serwera WWW Apache - polecenie **apt-get install apache2**

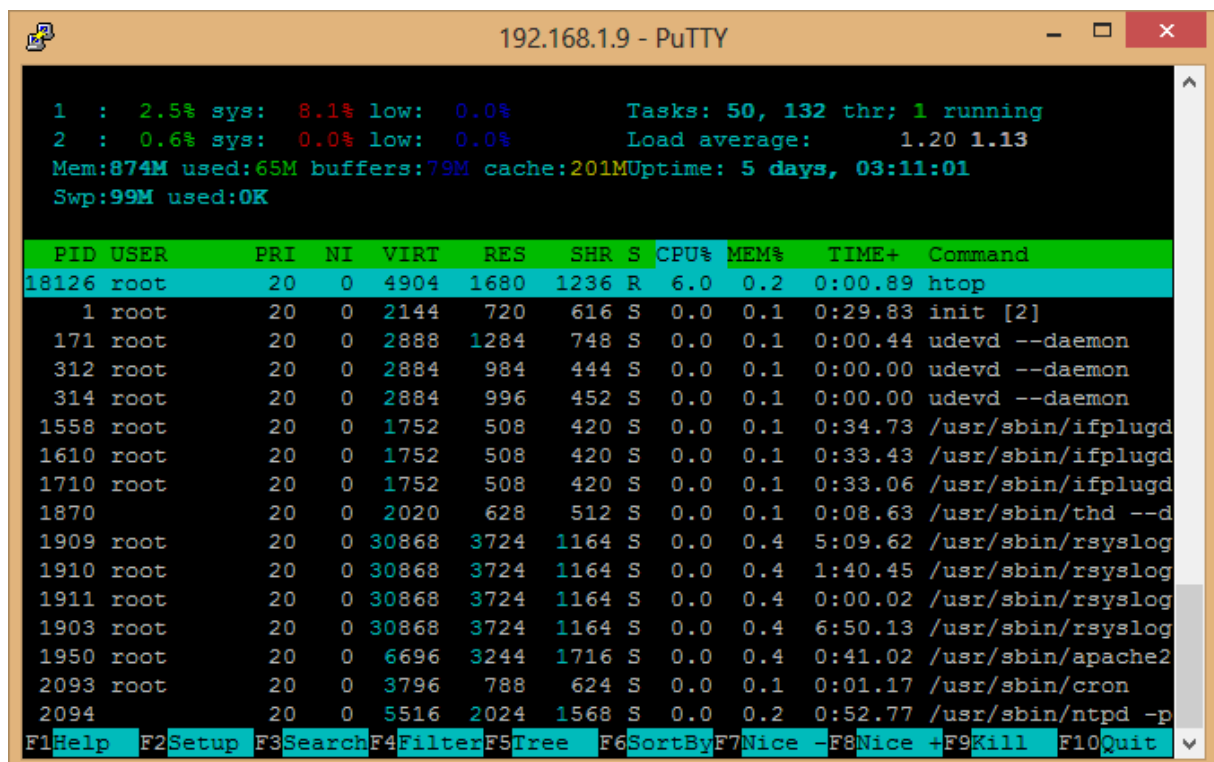
```
192.168.1.9 - PuTTY
root@lemaker:~# apt-get install apache2
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following extra packages will be installed:
  apache2-mpm-worker apache2-utils apache2.2-bin apache2.2-common libapr1
  libaprutil1 libaprutil1-dbd-sqlite3 libaprutil1-ldap ssl-cert
Suggested packages:
  apache2-doc apache2-suexec apache2-suexec-custom openssl-blacklist
The following NEW packages will be installed:
  apache2 apache2-mpm-worker apache2-utils apache2.2-bin apache2.2-common
  libapr1 libaprutil1 libaprutil1-dbd-sqlite3 libaprutil1-ldap ssl-cert
0 upgraded, 10 newly installed, 0 to remove and 113 not upgraded.
Need to get 1,360 kB of archives.
After this operation, 4,936 kB of additional disk space will be used.
Do you want to continue [Y/n]? y
```

5. Instalacja aplikacji **htop** (aplikacja służy do monitorowania zasobów maszyny) – polecenie **apt-get install htop**



```
192.168.1.9 - PuTTY
root@lemaker:~# apt-get install htop
Reading package lists... Done
Building dependency tree
Reading state information... Done
Suggested packages:
  ltrace
The following NEW packages will be installed:
  htop
0 upgraded, 1 newly installed, 0 to remove and 120 not upgraded.
Need to get 74.8 kB of archives.
After this operation, 212 kB of additional disk space will be used.
Get:1 http://mirrordirector.raspbian.org/raspbian/ wheezy/main htop armhf 1.0.1-1 [74.8 kB]
1% [1 htop 1,057 B/74.8 kB 1%]
```

Wygląd programu **htop** po uruchomieniu :



```
192.168.1.9 - PuTTY

 1 :  2.5% sys:  8.1% low:  0.0%      Tasks: 50, 132 thr; 1 running
 2 :  0.6% sys:  0.0% low:  0.0%      Load average:      1.20 1.13
Mem:874M used:65M buffers:79M cache:201MUptime: 5 days, 03:11:01
Swp:99M used:0K

 PID USER   PRI  NI  VIRT   RES   SHR  S  CPU% MEM%   TIME+  Command
18126 root    20   0  4904  1680  1236 R   6.0  0.2   0:00.89 htop
   1 root    20   0  2144   720   616 S   0.0  0.1   0:29.83 init [2]
  171 root    20   0  2888  1284   748 S   0.0  0.1   0:00.44 udevd --daemon
  312 root    20   0  2884   984   444 S   0.0  0.1   0:00.00 udevd --daemon
  314 root    20   0  2884   996   452 S   0.0  0.1   0:00.00 udevd --daemon
1558 root    20   0  1752   508   420 S   0.0  0.1   0:34.73 /usr/sbin/ifplugd
1610 root    20   0  1752   508   420 S   0.0  0.1   0:33.43 /usr/sbin/ifplugd
1710 root    20   0  1752   508   420 S   0.0  0.1   0:33.06 /usr/sbin/ifplugd
1870      20   0  2020   628   512 S   0.0  0.1   0:08.63 /usr/sbin/thd --d
1909 root    20   0 30868  3724  1164 S   0.0  0.4   5:09.62 /usr/sbin/rsyslog
1910 root    20   0 30868  3724  1164 S   0.0  0.4   1:40.45 /usr/sbin/rsyslog
1911 root    20   0 30868  3724  1164 S   0.0  0.4   0:00.02 /usr/sbin/rsyslog
1903 root    20   0 30868  3724  1164 S   0.0  0.4   6:50.13 /usr/sbin/rsyslog
1950 root    20   0  6696  3244  1716 S   0.0  0.4   0:41.02 /usr/sbin/apache2
2093 root    20   0  3796   788   624 S   0.0  0.1   0:01.17 /usr/sbin/cron
2094      20   0  5516  2024  1568 S   0.0  0.2   0:52.77 /usr/sbin/ntpd -p
F1Help  F2Setup F3Search F4Filter F5Tree  F6SortBy F7Nice -F8Nice +F9Kill F10Quit
```

Aby korzystać z aplikacji wygodniej, można zmienić sposób wyświetlania danych na wartości procentowe. Dokonać tego można poprzez wciśnięcie klawisza F2, a następnie przejście do kolumny drugiej oznaczającej sposób wyświetlania danych (edycja następuje poprzez wciśnięcie klawisza Enter).

The screenshot shows a PuTTY terminal window titled "192.168.1.9 - PuTTY". The terminal displays system statistics:

```

1 : 6.9% sys: 8.1% low: 0.0% Tasks: 50, 132 thr; 1 running
2 : 3.2% sys: 0.6% low: 0.0% Load average: 1.19 1.14
Mem:874M used:65M buffers:79M cache:201MUptime: 5 days, 03:13:33
Swp[ ]

```

Below the statistics is a configuration menu with four columns:

Setup	Left column	Right column	Available meters
Meters	CPUs (1/1) [Text]	Task counter [Text]	Clock
Display options	Memory [Text]	Load average [Text]	Load average
Colors	Swap [Bar]	Uptime [Text]	Load
Columns			Memory
			Swap
			Task counter
			Uptime
			Battery
			Hostname
			CPUs (1/1)
			CPUs (1&2/2)
			CPUs (1/2)
			CPUs (2/2)
			CPUs (1&2/4)
			CPUs (3&4/4)
			CPU average

At the bottom of the terminal, there is a row of function key shortcuts: F1, F2, F3, F4Type, F5, F6, F7MoveUp, F8MoveDn, F9Remove, F10Done.

## 6. Instalacja XRDP (Usługa RDP dla Linuxa)

Usługa RDP dla Linuxa może zostać zrealizowana poprzez pakiet XRDP.

The screenshot shows a PuTTY terminal window titled "192.168.1.25 - PuTTY". The terminal displays the command to install XRDP:

```

root@Bpi:~# apt-get install xrdp
Reading package lists... Done
Building dependency tree
Reading state information... Done
xrdp is already the newest version.
0 upgraded, 0 newly installed, 0 to remove and 35 not upgraded.
root@Bpi:~#

```

## 7. Instalacja aplikacji testowe Siege

Aplikację Siege pobieramy ze strony <https://www.joedog.org/siege-home/> (APLIKACJA PRZEZNACZONA NA KOMPUTERY Z SYSTEMAMI Z RODZINY UNIX/LINUX)

Wersja testowa na system WINDOWS - <https://code.google.com/archive/p/siege-windows/>

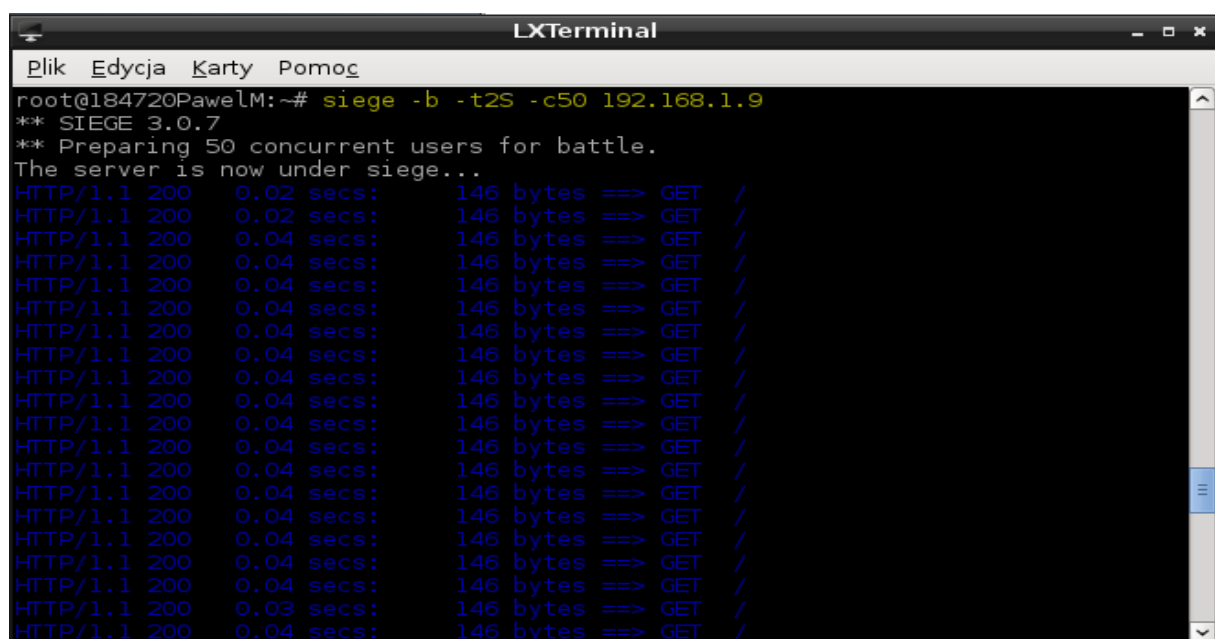
Pobraną aplikację umieszczamy na maszynie, za pomocą której będziemy testować serwer Apache. Po rozpakowaniu archiwum \*.tar.gz (polecenie `tar -zxvf <ścieżka_do_archiwum>`), otwieramy folder z rozpakowaną aplikacją i wykonujemy następujące polecenia:

- `./configure` – konfiguracja instalatora pod dany system
- `Make` – kompilacja aplikacji wg. wcześniejszej konfiguracji
- `Make install` – instalacja wcześniej skompilowanego programu

Po wykonaniu powyższego zestawu instrukcji możemy zacząć korzystać z aplikacji. Można tego dokonać za pomocą polecenia:

- `Siege <parametry> <adres_IP_serwera_www>`
- Można też sprecyzować port:
  - `Siege <parametry> <adres_IP_serwera_www:port>`

Przykładowe wywołanie aplikacji Siege

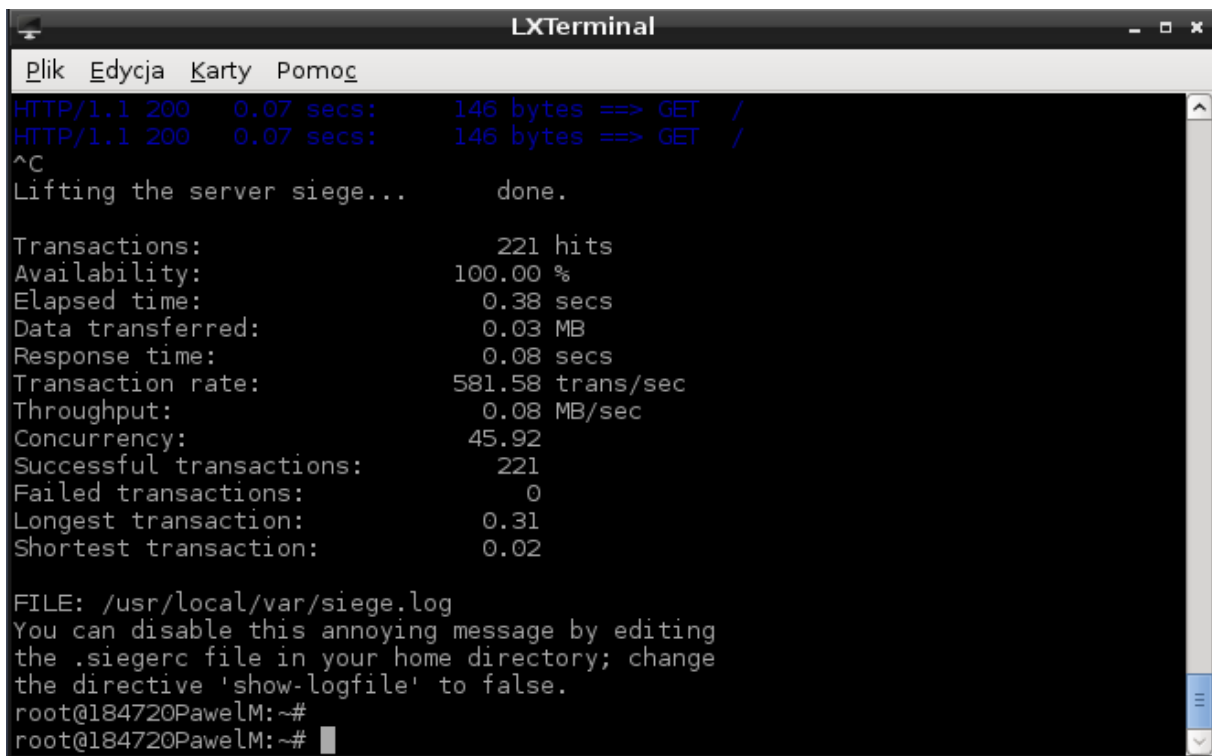


```
LXTerminal
Plik Edycja Karty Pomoc
root@184720PawelM:~# siege -b -t2S -c50 192.168.1.9
** SIEGE 3.0.7
** Preparing 50 concurrent users for battle.
The server is now under siege...
HTTP/1.1 200 0.02 secs: 146 bytes ==> GET /
HTTP/1.1 200 0.02 secs: 146 bytes ==> GET /
HTTP/1.1 200 0.04 secs: 146 bytes ==> GET /
HTTP/1.1 200 0.04 secs: 146 bytes ==> GET /
HTTP/1.1 200 0.04 secs: 146 bytes ==> GET /
HTTP/1.1 200 0.04 secs: 146 bytes ==> GET /
HTTP/1.1 200 0.04 secs: 146 bytes ==> GET /
HTTP/1.1 200 0.04 secs: 146 bytes ==> GET /
HTTP/1.1 200 0.04 secs: 146 bytes ==> GET /
HTTP/1.1 200 0.04 secs: 146 bytes ==> GET /
HTTP/1.1 200 0.04 secs: 146 bytes ==> GET /
HTTP/1.1 200 0.04 secs: 146 bytes ==> GET /
HTTP/1.1 200 0.04 secs: 146 bytes ==> GET /
HTTP/1.1 200 0.04 secs: 146 bytes ==> GET /
HTTP/1.1 200 0.04 secs: 146 bytes ==> GET /
HTTP/1.1 200 0.04 secs: 146 bytes ==> GET /
HTTP/1.1 200 0.04 secs: 146 bytes ==> GET /
HTTP/1.1 200 0.04 secs: 146 bytes ==> GET /
HTTP/1.1 200 0.03 secs: 146 bytes ==> GET /
HTTP/1.1 200 0.04 secs: 146 bytes ==> GET /
```

- **b** – oznacza tryb, w którym nie dodaje się do testu dodatkowych opóźnień

- **t2M** – określenie czasu trwania testu, w tym przypadku przedziałem czasowym są dwie minuty, dla np. 10 sekund będzie to **t10S**
- **c** – oznacza ilość symulowanych użytkowników biorących udział w teście, tutaj wartością tą było **50** symulowanych użytkowników

Po zakończeniu testu otrzymujemy wynik:



```

LXTerminal
Plik Edycja Karty Pomoc
HTTP/1.1 200 0.07 secs: 146 bytes ==> GET /
HTTP/1.1 200 0.07 secs: 146 bytes ==> GET /
^C
Lifting the server siege... done.

Transactions:      221 hits
Availability:      100.00 %
Elapsed time:      0.38 secs
Data transferred:  0.03 MB
Response time:     0.08 secs
Transaction rate:  581.58 trans/sec
Throughput:        0.08 MB/sec
Concurrency:       45.92
Successful transactions: 221
Failed transactions: 0
Longest transaction: 0.31
Shortest transaction: 0.02

FILE: /usr/local/var/siege.log
You can disable this annoying message by editing
the .siegerc file in your home directory; change
the directive 'show-logfile' to false.
root@184720PawelM:~#
root@184720PawelM:~#

```

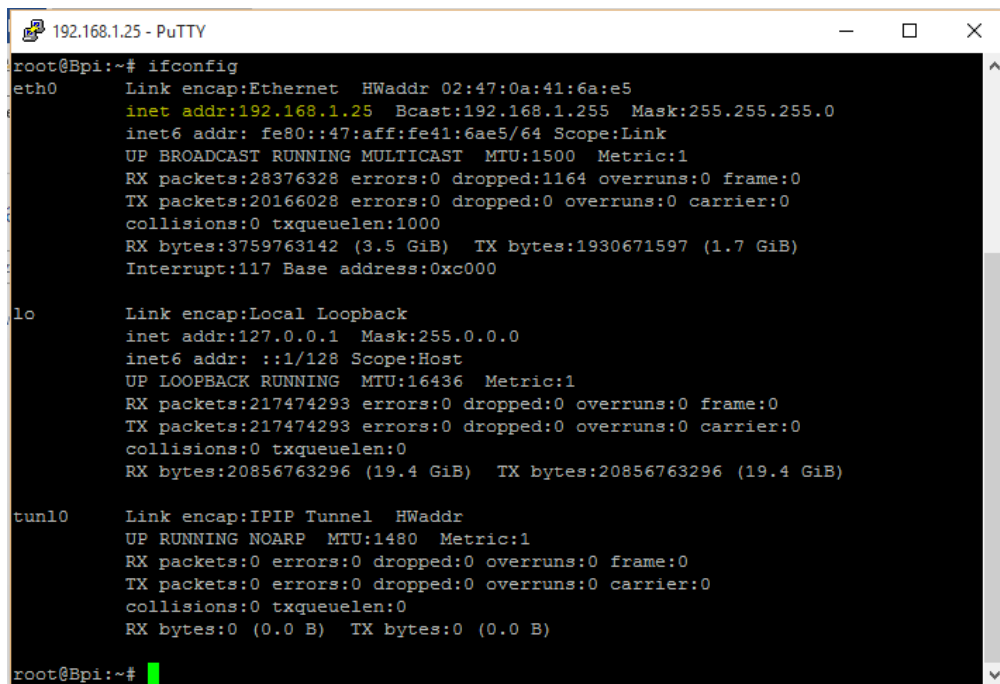
Dodatkowo wynik zostaje zapisany w lokalizacji **/usr/local/var/siege.log** (jeśli włączymy tą opcję w pliku konfiguracyjnym).

## Zadania do wykonania

### 1. Ustalenie adresu IP urządzenia

W celu poznania przypisanego adresu IP musimy wywołać polecenie **ifconfig**. Aby uzyskać pełną listę wszystkich interfejsów sieciowych (nawet tych nieaktywnych – fizycznie są w urządzeniu jednak nie mają podłączonego medium kablowego lub bezprzewodowego) należy wywołać polecenie **ifconfig -a**





```
root@Bpi:~# ifconfig
eth0      Link encap:Ethernet  HWaddr 02:47:0a:41:6a:e5
          inet addr:192.168.1.25  Bcast:192.168.1.255  Mask:255.255.255.0
          inet6 addr: fe80::47:aff:fe41:6ae5/64  Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:28376328 errors:0 dropped:1164 overruns:0 frame:0
          TX packets:20166028 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:3759763142 (3.5 GiB)  TX bytes:1930671597 (1.7 GiB)
          Interrupt:117 Base address:0xc000

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128  Scope:Host
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:217474293 errors:0 dropped:0 overruns:0 frame:0
          TX packets:217474293 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:20856763296 (19.4 GiB)  TX bytes:20856763296 (19.4 GiB)

tunl0     Link encap:IPIP Tunnel  HWaddr
          UP RUNNING NOARP  MTU:1480  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)

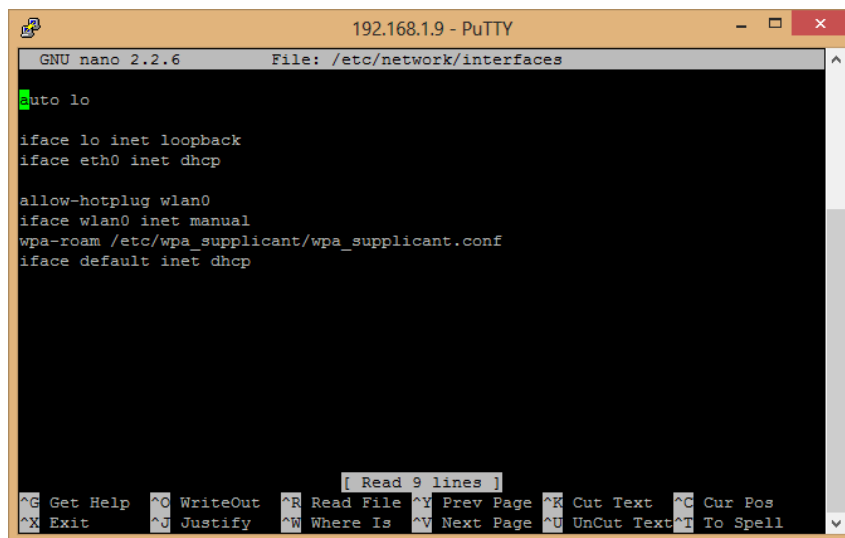
root@Bpi:~#
```

## 2. Ustalenie adresu IP urządzenia

Istnieją dwa sposoby na ustawienie konfiguracji sieciowej. Jeden z nich jest to ustawienie karty sieciowej w tryb DHCP co wiąże się z automatycznym uzyskaniem adresu IP, maski podsieci, adresu bramy czy adresu DNS. Ten sposób jest preferowany i wykorzystany podczas zajęć laboratoryjnych. Drugi sposób przewiduje ręczną konfigurację ustawień sieciowych co wiąże się z ustawieniem odpowiedniego adresu IP dla wszystkich urządzeń które mają się ze sobą komunikować (adresy IP powinny znajdować się w tej samej podsieci. Dodatkowo jeśli mają komunikować się ze „światem” zewnętrznym niezbędne będzie ustawienie adresu bramy oraz adresu serwera DNS). Aby umożliwić komunikację w topologii bez działającej usługi DHCP, należy odpowiednio ustawić interfejsy sieciowe tak komputera laboratoryjnego, jak i Banana Pi. Interfejsy sieciowe w takiej topologii muszą być skonfigurowane statycznie i należeć do tej samej podsieci. Interfejs sieciowy Banana Pi jest już ustawiony na działanie w podsieci 192.168.1.0/24 (Adres ustawiony na wartość 192.168.1.10). Kolejno należy ustawić adres IP komputera testowego na wolny adres przynależący do wykorzystanej już podsieci.

Dla systemów z rodziny Linux (w procesie przygotowania instrukcji wykorzystano system Debian) plik konfiguracyjny interfejsów sieciowych znajduje się w lokalizacji **/etc/network/interfaces**. Plik możemy otworzyć w dowolnym edytorze tekstu (w trybie graficznym może to być np. Leafpad, w trybie tekstowym będzie to przykładowo Vi, Vim lub Nano – w przykładach wykorzystano edytor Nano). Poniżej przedstawiono wygląd

standardowej konfiguracji interfejsów sieciowych w systemie Raspbian dla platformy Banana Pi.



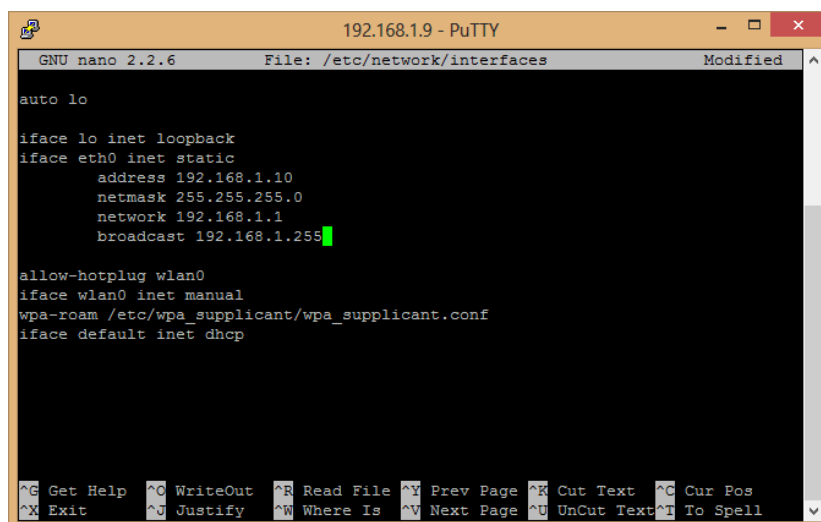
```
GNU nano 2.2.6 File: /etc/network/interfaces

auto lo

iface lo inet loopback
iface eth0 inet dhcp

allow-hotplug wlan0
iface wlan0 inet manual
wpa-roam /etc/wpa_supplicant/wpa_supplicant.conf
iface default inet dhcp
```

Poniżej przedstawiono skonfigurowane już ustawienia sieciowe.



```
GNU nano 2.2.6 File: /etc/network/interfaces Modified

auto lo

iface lo inet loopback
iface eth0 inet static
    address 192.168.1.10
    netmask 255.255.255.0
    network 192.168.1.1
    broadcast 192.168.1.255

allow-hotplug wlan0
iface wlan0 inet manual
wpa-roam /etc/wpa_supplicant/wpa_supplicant.conf
iface default inet dhcp
```

Zmianę ustawień sieciowych można też wykonać za pomocą polecenia

**Ifconfig eth0 192.168.1.10 netmask 255.255.255.0 up**

**Gdzie:**

- Eth0 – nazwa interfejsu sieciowego
- Up – włączenie interfejsu sieciowego

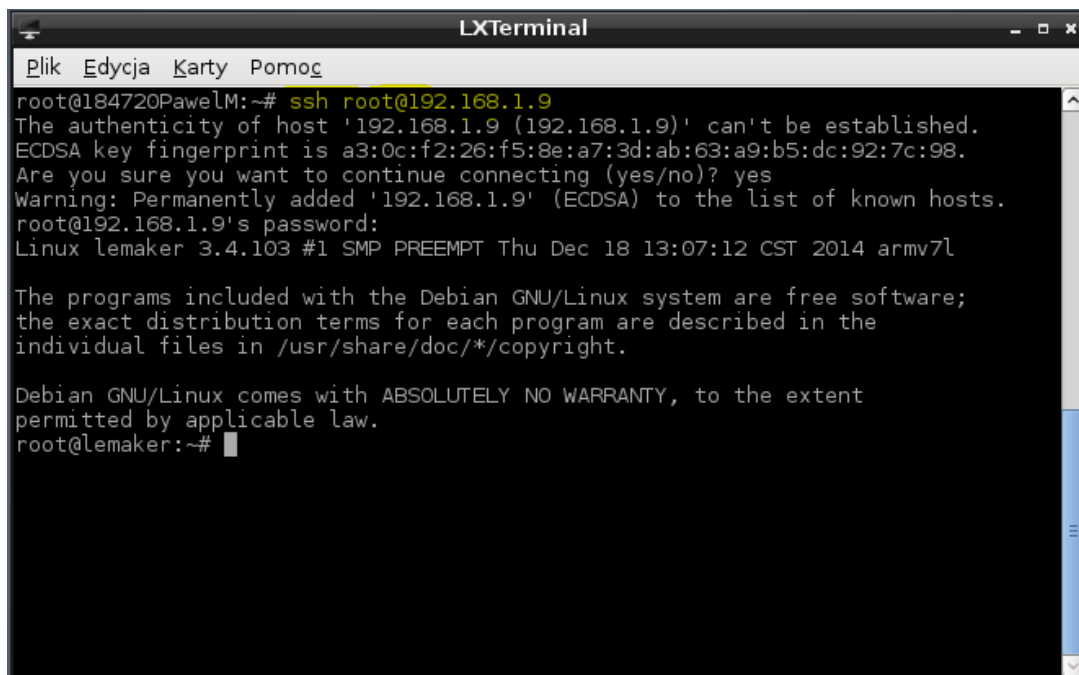
Restart karty sieciowej możemy wywołać poprzez wykonanie jednego z poleceń. Ponowny rozruch karty sieciowej będzie niezbędny w przypadku zmiany statycznego adresu IP lub jeśli

będziemy chcieli uzyskać nowy adres z puli DHCP (np. w momencie kiedy zmienimy np. podsieć z której przydzielane są adresy IP):

- **Interface eth0 down** -> **interface eth0 up** (numery interfejsów można zweryfikować poprzez weryfikację pliku **/etc/udev/rules.d/70-persistent-network**)
- **/etc/init.d/networking restart**
- **Reboot** (restart całej maszyny)

### 3. Obsługa połączenia SSH

W celu umożliwienia zdalnej komunikacji z serwerem WWW (i całą platformą Banana Pi), system Raspbian ma domyślnie włączoną usługę serwera SSH. W celu połączenia się poprzez ten protokół potrzebujemy posiadać odpowiedni program, np. **putty**. W przypadku łączenia się z systemu Linux, możemy wykorzystać klient SSH wbudowany w powłokę systemową. W tym celu wywołujemy polecenie **ssh <użytkownik>@<adres\_ip>**



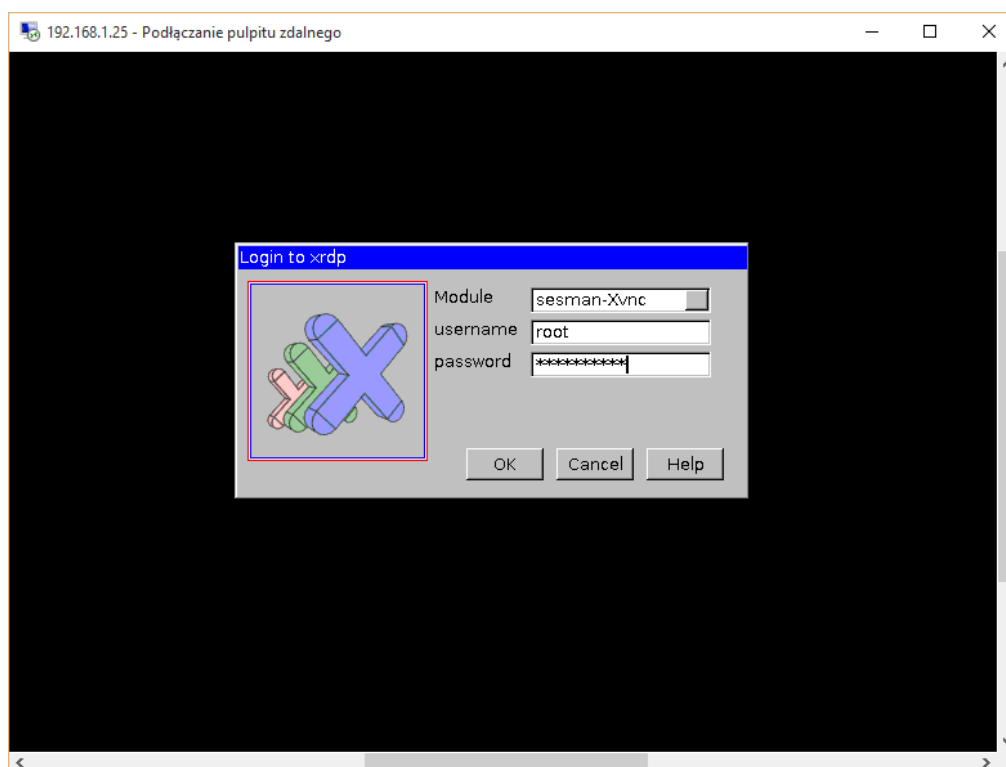
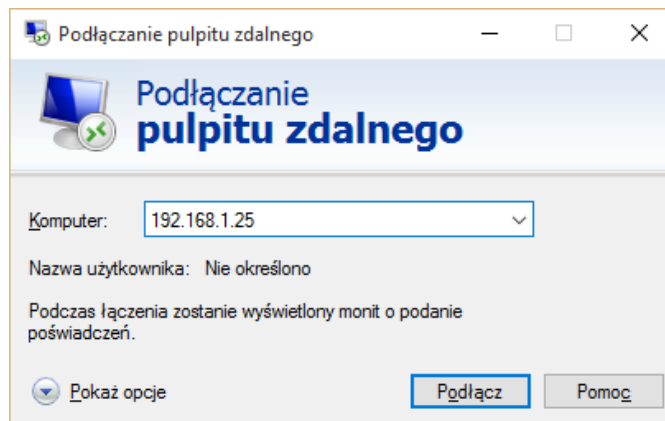
```
LXTerminal
Plik  Edycja  Karty  Pomoc
root@184720PawelM:~# ssh root@192.168.1.9
The authenticity of host '192.168.1.9 (192.168.1.9)' can't be established.
ECDSA key fingerprint is a3:0c:f2:26:f5:8e:a7:3d:ab:63:a9:b5:dc:92:7c:98.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '192.168.1.9' (ECDSA) to the list of known hosts.
root@192.168.1.9's password:
Linux lemaker 3.4.103 #1 SMP PREEMPT Thu Dec 18 13:07:12 CST 2014 armv7l

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

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
root@lemaker:~#
```

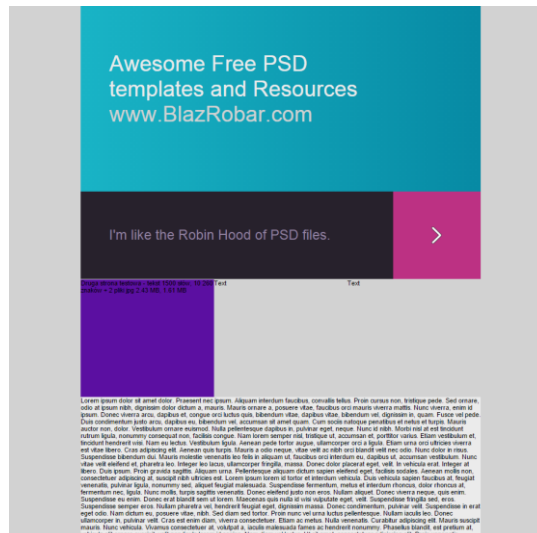
### 4. Obsługa połączenia RDP (zdalny pulpit)

Aby nawiązać połączenie z komputera z systemem Windows należy znać IP maszyny do której chcemy się połączyć (tutaj maszyną tą jest minikomputer Banana Pi). Kiedy znamy już IP możemy połączyć się z maszyną poprzez domyślny klient systemu Windows (**Podłączenie pulpitu Zdalnego**)



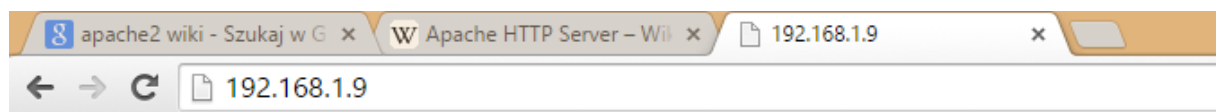
## 5. Obsługa serwera www - Apache

Do wykonania zadań przedstawionych w tej instrukcji niezbędne jest utworzenie (lub wykorzystanie przykładowej) strony WWW, która posłuży jako treść udostępniana na testowanym serwerze Apache.



Aby serwer Apache poprawnie wyświetlał stronę, należy umieścić ją w odpowiedniej lokalizacji. Dla serwera Apache w domyślnej konfiguracji, strony zlokalizowane są w **/var/www/** lub w nowszych wersjach **/var/www/html** (lokalizacje te oczywiście można zmieniać). Domyślna strona testowa znajduje się na pulpicie maszyny testowej **/home/bananapi/Desktop/strona\_testowa** lub **/home/strona\_testowa** (kopiujemy zawartość folderu bezpośrednio do lokalizacji!). Aby strona była wyświetlana razem z stylami CSS, po skopiowaniu strony testowej do lokalizacji **/var/www** należy nadać uprawnienia do folderu **css** – operację tą wykonujemy poprzez wywołanie polecenia **chmod 755 /var/www/css** (uprawnienia w systemach Linux to – rwx – odczyt/zapis/wykonanie – zapis liczbowy 4 2 1)

Aby test był miarodajny, strona WWW powinna zawierać sporą liczbę danych tekstowych, zdjęcia, itp. Poniżej przedstawiono stronę domyślną dla serwera Apache:



## It works!

This is the default web page for this server.

The web server software is running but no content has been added, yet.

Domyślna strona edytowana przez Nano:



```
192.168.1.9 - PuTTY
GNU nano 2.2.6 File: /var/www/index.html
<html><body><h1>It works!</h1>
<p>This is the default web page for this server.</p>
<p>The web server software is running but no content has been added, yet.</p>
</body></html>
[ Read 4 lines ]
^G Get Help ^O WriteOut ^R Read File ^Y Prev Page ^K Cut Text ^C Cur Pos
^X Exit ^J Justify ^W Where Is ^V Next Page ^U UnCut Text ^T To Spell
```

Po umieszczeniu strony www w odpowiedniej lokalizacji, należy sprawdzić poprawność działania serwera. Można tego dokonać poprzez wpisanie w przeglądarce adresu serwera WWW.

#### 6. Testy aplikacją Siege, monitoring zasobów serwera aplikacją htop

Mając już działający serwer WWW można przystąpić do testów aplikacją Siege. Pierwszym krokiem jest test serwera z domyślnymi parametrami. Na maszynie testowej uruchamiamy program **Siege** z wybranymi przez siebie parametrami (zaleca się testowanie przynajmniej przez co najmniej 4 minuty z wykorzystaniem minimum 100 symulowanych użytkowników – opis obsługi aplikacji **Siege** znajduje się w sekcji Przygotowanie platformy – pkt 7). Wyniki testu zapisujemy. Następnie należy zmodyfikować parametry serwera Apache (dowolność – np. możemy zmniejszyć lub zwiększyć maksymalną liczbę klientów) i powtórzyć testy z takimi samymi parametrami aplikacji Siege i zweryfikować, czy pojawiły się różnice to znaczy:

- jak zmieniła się liczba transakcji
- ile danych udało się pobrać
- jak zmieniła się dostępność i czas odpowiedzi serwera

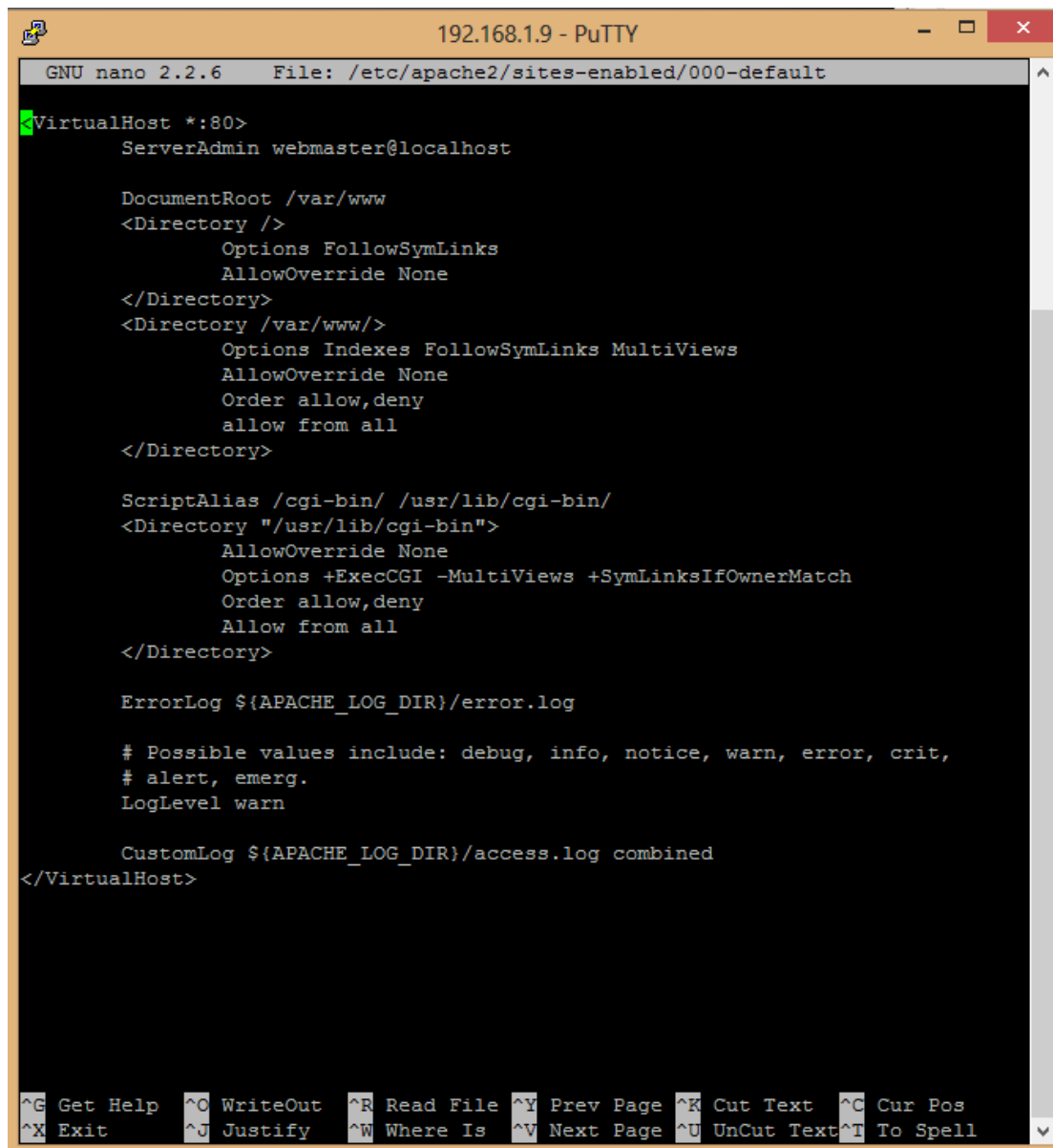
Konfiguracja serwera Apache znajduje się w lokalizacji **/etc/apache2/apache2.conf**

```
GNU nano 2.2.6 File: /etc/apache2/apache2
Timeout 300
#
# KeepAlive: Whether or not to allow persistent connections (more than
# one request per connection). Set to "Off" to deactivate.
#
KeepAlive On
#
# MaxKeepAliveRequests: The maximum number of requests to allow
# during a persistent connection. Set to 0 to allow an unlimited amount.
# We recommend you leave this number high, for maximum performance.
#
MaxKeepAliveRequests 100
#
# KeepAliveTimeout: Number of seconds to wait for the next request from the
# same client on the same connection.
#
KeepAliveTimeout 5
##
## Server-Pool Size Regulation (MPM specific)
##
# prefork MPM
# StartServers: number of server processes to start
# MinSpareServers: minimum number of server processes which are kept spare
# MaxSpareServers: maximum number of server processes which are kept spare
# MaxClients: maximum number of server processes allowed to start
# MaxRequestsPerChild: maximum number of requests a server process serves
<IfModule mpm_prefork_module>
    StartServers      5
    MinSpareServers   5
    MaxSpareServers   10
    MaxClients        150
    MaxRequestsPerChild 0
</IfModule>
# worker MPM
# StartServers: initial number of server processes to start
# MinSpareThreads: minimum number of worker threads which are kept spare
# MaxSpareThreads: maximum number of worker threads which are kept spare
# ThreadLimit: ThreadsPerChild can be changed to this maximum value during a
#               graceful restart. ThreadLimit can only be changed by stopping
#               and starting Apache.
# ThreadsPerChild: constant number of worker threads in each server process
# MaxClients: maximum number of simultaneous client connections
# MaxRequestsPerChild: maximum number of requests a server process serves
<IfModule mpm_worker_module>
    StartServers      2
    MinSpareThreads   25
    MaxSpareThreads   75
    ThreadLimit       64
    ThreadsPerChild   25
    MaxClients        150
    MaxRequestsPerChild 0
</IfModule>
^G Get Help      ^O WriteOut      ^R Read File
^X Exit          ^J Justify       ^W Where Is
```

Dodatkowo, w trakcie trwania testu, należy monitorować zużycie zasobów na serwerze WWW (aplikacja **htop** uruchomiona np. w drugiej sesji ssh).

## 7. Konfiguracja wirtualnych hostów na serwerze Apache, testy aplikacja Siege, monitoring zasobów

Wirtualny host na platformie Apache umożliwia propagowanie danej strony (serwisu itp.) WWW. W celu umożliwienia jednoczesnego dystrybuowania kilku takich stron WWW, można skonfigurować kolejnych wirtualnych klientów. W tym celu dodajemy kolejny blok instrukcji w konfiguracji wyświetlanych stron serwera Apache. Konfiguracja zawarta jest w pliku `/etc/apache2/sites-enabled/000-default` Poniżej przedstawiono konfigurację podstawową, gdzie skonfigurowano jednego domyślnego wirtualnego hosta (konfiguracja domyślna – **port 80**).



The image shows a PuTTY terminal window titled "192.168.1.9 - PuTTY". Inside the terminal, the GNU nano 2.2.6 text editor is open, editing the file "/etc/apache2/sites-enabled/000-default". The editor displays the configuration for a VirtualHost. The content is as follows:

```
VirtualHost *:80>
    ServerAdmin webmaster@localhost

    DocumentRoot /var/www
    <Directory />
        Options FollowSymLinks
        AllowOverride None
    </Directory>
    <Directory /var/www/>
        Options Indexes FollowSymLinks MultiViews
        AllowOverride None
        Order allow,deny
        allow from all
    </Directory>

    ScriptAlias /cgi-bin/ /usr/lib/cgi-bin/
    <Directory "/usr/lib/cgi-bin">
        AllowOverride None
        Options +ExecCGI -MultiViews +SymLinksIfOwnerMatch
        Order allow,deny
        Allow from all
    </Directory>

    ErrorLog ${APACHE_LOG_DIR}/error.log

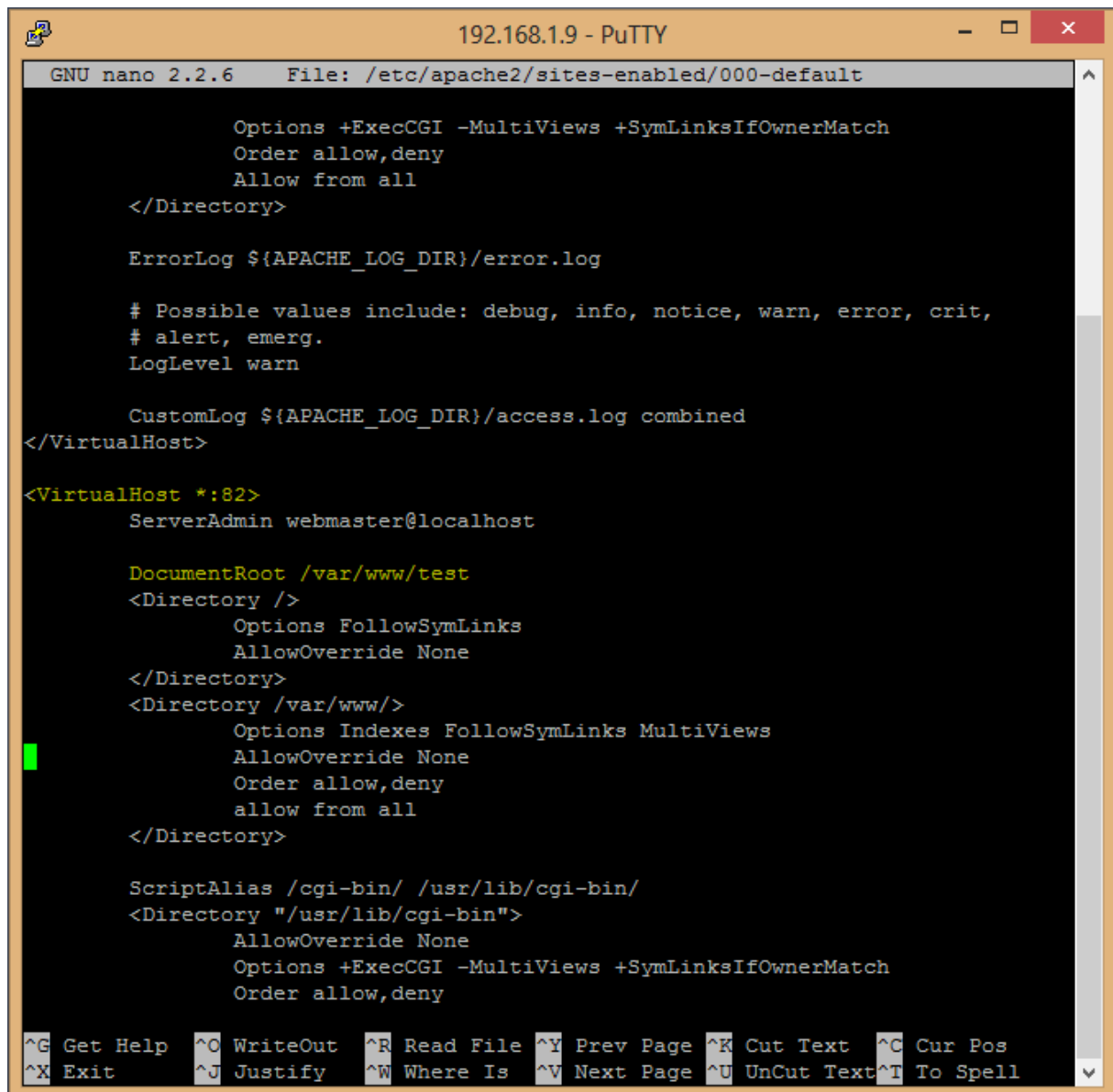
    # Possible values include: debug, info, notice, warn, error, crit,
    # alert, emerg.
    LogLevel warn

    CustomLog ${APACHE_LOG_DIR}/access.log combined
</VirtualHost>
```

At the bottom of the terminal, there is a status bar with various keyboard shortcuts for the nano editor, such as ^G Get Help, ^O WriteOut, ^R Read File, ^Y Prev Page, ^K Cut Text, ^C Cur Pos, ^X Exit, ^J Justify, ^W Where Is, ^V Next Page, ^U UnCut Text, and ^T To Spell.

W najprostszym wariantcie wystarczy skopiować cały blok dotyczący wirtualnego hosta (fragment zaczynający się znacznikiem **<VirtualHost \*:80>** a kończący na znaczniku **</VirtualHost>**) Należy także zmienić port i ścieżkę zawierającą nową stronę WWW (katalog ten trzeba utworzyć i umieścić tam kolejną stronę WWW, np. modyfikację wcześniej wykorzystanej strony).





The screenshot shows a PuTTY terminal window titled "192.168.1.9 - PuTTY". Inside, the GNU nano 2.2.6 text editor is open, editing the file `/etc/apache2/sites-enabled/000-default`. The configuration is for an Apache virtual host. The first `</VirtualHost>` block is empty except for logging. The second `<VirtualHost *:82>` block is being edited and contains the following configuration:

```
ServerAdmin webmaster@localhost

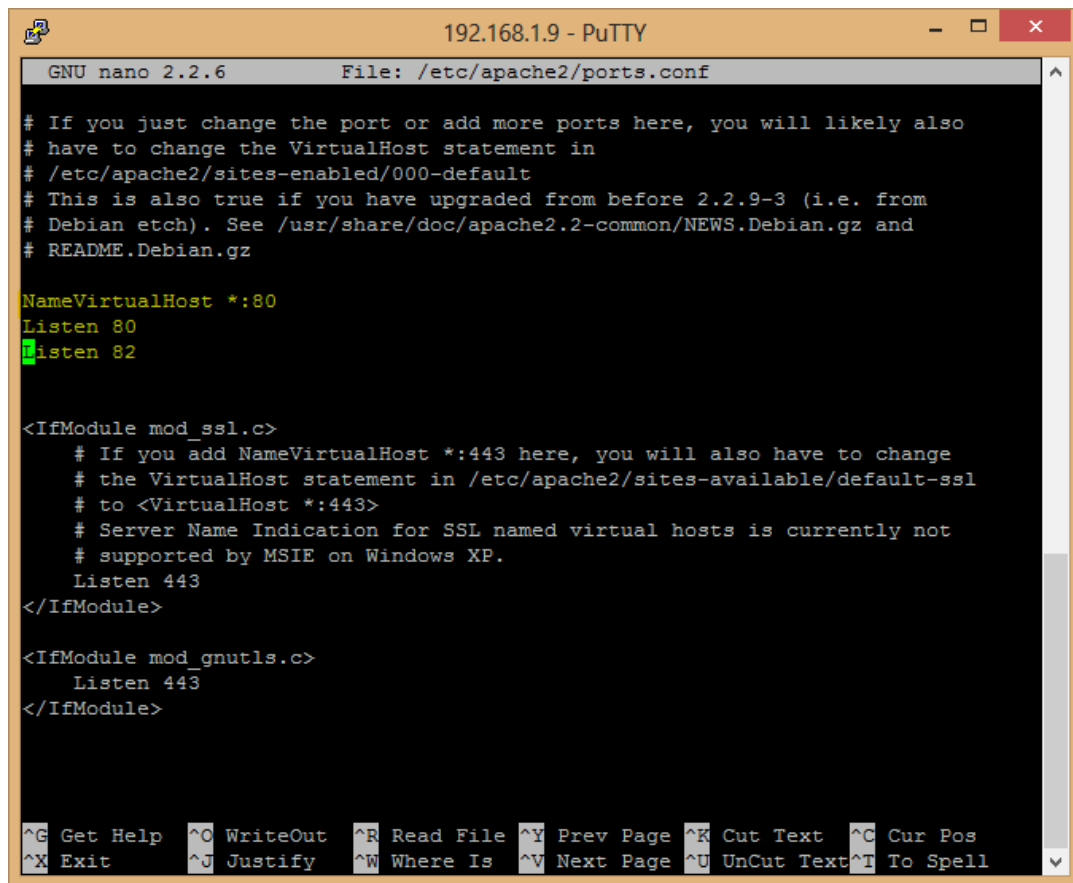
DocumentRoot /var/www/test
<Directory />
    Options FollowSymLinks
    AllowOverride None
</Directory>
<Directory /var/www/>
    Options Indexes FollowSymLinks MultiViews
    AllowOverride None
    Order allow,deny
    allow from all
</Directory>

ScriptAlias /cgi-bin/ /usr/lib/cgi-bin/
<Directory "/usr/lib/cgi-bin">
    AllowOverride None
    Options +ExecCGI -MultiViews +SymLinksIfOwnerMatch
    Order allow,deny
```

At the bottom of the terminal, a status bar shows various keyboard shortcuts: `^G Get Help`, `^O WriteOut`, `^R Read File`, `^Y Prev Page`, `^K Cut Text`, `^C Cur Pos`, `^X Exit`, `^J Justify`, `^W Where Is`, `^V Next Page`, `^U UnCut Text`, and `^T To Spell`.

Kolejnym krokiem jest uruchomienie nasłuchiwanie serwera Apache na nowym porcie. Plik konfiguracyjny zlokalizowany jest `/etc/apache2/ports.conf`

Poniżej przedstawiono wygląd pliku konfiguracyjnego **ports.conf** (dodano już nasłuchiwanie związane z nowym wirtualnym hostem)



The screenshot shows a terminal window titled "192.168.1.9 - PuTTY" running GNU nano 2.2.6. The file being edited is /etc/apache2/ports.conf. The configuration includes comments about changing ports and VirtualHost statements, and lists listening ports: 80, 82, and 443 (for SSL and gnutls). The bottom of the terminal shows various keyboard shortcuts for nano.

```
GNU nano 2.2.6 File: /etc/apache2/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
# This is also true if you have upgraded from before 2.2.9-3 (i.e. from
# Debian etch). See /usr/share/doc/apache2.2-common/NEWS.Debian.gz and
# README.Debian.gz

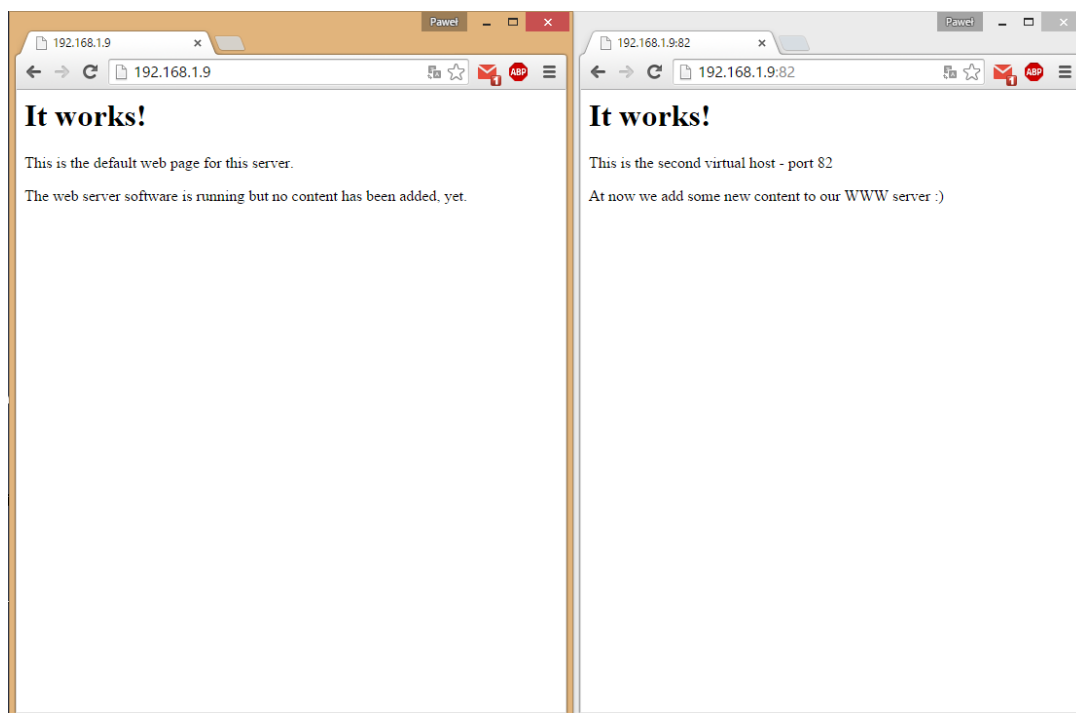
NameVirtualHost *:80
Listen 80
Listen 82

<IfModule mod_ssl.c>
    # If you add NameVirtualHost *:443 here, you will also have to change
    # the VirtualHost statement in /etc/apache2/sites-available/default-ssl
    # to <VirtualHost *:443>
    # Server Name Indication for SSL named virtual hosts is currently not
    # supported by MSIE on Windows XP.
    Listen 443
</IfModule>

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

^G Get Help  ^O WriteOut  ^R Read File  ^Y Prev Page  ^K Cut Text   ^C Cur Pos
^X Exit      ^J Justify   ^W Where Is   ^V Next Page  ^U UnCut Text ^T To Spell
```

Poniżej przedstawiono poprawne współdziałanie wirtualnych hostów.



Po wykonaniu konfiguracji, ponownie należy przeprowadzić testy aplikacją **Siege**. Tym razem należy uruchomić kolejny instancję programu (dzięki temu można zweryfikować

działanie drugiego wirtualnego hosta czyli np. **192.168.1.9:82**) w trakcie trwania poprzedniego i sprawdzić jak tym razem poradzi sobie serwer WWW mając zwielokrotnione zadania do przetworzenia. Ponownie w trakcie trwania testów należy monitorować zasoby na serwerze WWW (platforma Banana Pi).