

43

Uruchamianie usługi WWW

ZAGADNIENIA

- Funkcja usługi WWW
- Udostępnianie katalogów na serwerze WWW

Usługa WWW (ang. *World Wide Web*) jest hipertekstowym, multimedialnym, sieciowym systemem informacyjnym opartym na publicznie dostępnych, otwartych standardach. Podstawowym zadaniem WWW jest publikowanie informacji. Aby uzyskać dostęp do tak udostępnianej informacji, trzeba posłużyć się przeglądarką internetową. Przeglądarka łączy się z serwerem WWW, skąd pobiera pewien zbiór informacji, określany jako strona internetowa. Strona internetowa może zostać wyświetlona, zapisana w lokalnym systemie plików lub wydrukowana. Usługa ta działa na podstawie architektury klient-serwer.

43.1. Udostępniane katalogi

Serwer WWW może udostępniać pojedynczą witrynę internetową. Pobieranie danych z takiej witryny polega na wpisaniu w pole adresu przeglądarki internetowej nazwy serwera lub jego adresu IP. W serwerach z zainstalowanym systemem Linux katalogiem głównym, w którym domyślnie są przechowywane witryny, jest `/var/www`. Przeglądanie stron internetowych rozpoczyna się od pliku `index.html` (w niektórych systemach może to być: `index.htm`, `default.html`, `default.htm`, `index.php`). Jeżeli w katalogu głównym witryny znajduje się plik `index.html`, to po wpisaniu adresu witryny zostanie wyświetlony przez przeglądarkę. Jeżeli nazwa pliku jest inna, to należy w adresie podać nazwę pliku, który ma być wyświetlony. Inne witryny internetowe mogą być umieszczane w podkatalogach katalogu głównego. W takim wypadku, aby wyświetlić odpowiednią witrynę w polu adresu, trzeba podać nazwę komputera oraz ścieżkę dostępu (należy rozpocząć od katalogu głównego witryny WWW). Serwer WWW może udostępniać również wiele witryn internetowych, wykorzystując „wirtualne serwery”. Każdy z takich wirtualnych serwerów może być identyfikowany za pomocą innej nazwy, adresu IP lub numeru portu. Położenie katalogu głównego takiej witryny jest określone przez administratora serwera WWW. Na serwerze WWW mogą być udostępniane również witryny użytkowników mających konta w systemie. W serwerach z zainstalowanym systemem Linux każdy z użytkowników może udostępniać swoje witryny w specjalnie do tego celu przeznaczonym katalogu `public_html` (jeżeli konfiguracja serwera na to zezwala). Katalog `public_html` jest umieszczany w katalogu domowym użytkownika `/home/nazwa_uzytkownika`.

43.2. Instalacja serwera WWW

ZAGADNIENIA

■ Instalowanie serwera WWW

Większość dystrybucji Linuksa standardowo zawiera oprogramowanie umożliwiające uruchomienie serwera WWW. Podczas instalacji Linuksa serwer ten może być zainstalowany domyślnie. Jeżeli podczas instalacji serwer WWW nie został zainstalowany, można go doinstalować w dowolnym momencie. Do instalacji pakietów serwera należy wykorzystać menedżer pakietów **apt**. Polecenie:

```
sudo apt install apache2
```

spowoduje zainstalowanie serwera i dodatków niezbędnych do jego pracy.

43.3. Uruchamianie serwera WWW

ZAGADNIENIA

■ Uruchomienie serwera WWW

Serwer WWW można uruchomić za pomocą polecenia **service**, np. aby uruchomić serwer, należy wpisać polecenie:

```
sudo service apache2 start.
```

W celu zatrzymania usługi należy wydać polecenie:

```
sudo service apache2 stop.
```

Jeżeli usługa była wcześniej uruchomiona i dokonamy zmian w konfiguracji, należy uruchomić usługę ponownie. Można to zrobić dzięki poleceniu:

```
sudo service apache2 restart.
```

Konfiguracja dostarczona przez dystrybucję pozwala na wykorzystanie serwera przez większość użytkowników bez konieczności dokonywania jakichkolwiek zmian w konfiguracji. Jeżeli konfiguracja ta nie jest odpowiednia, można ją zmienić.

SPRAWDŹ SWOJE UMIEJĘTNOŚCI

1. Sprawdź w internecie, czy istnieje inny niż **Apache** serwer WWW przeznaczony do pracy w systemie Linux.
2. Wyszukaj w internecie informacje dotyczące procentowego udziału w rynku różnych serwerów WWW.

SPRAWDŹ SWOJĄ WIEDZĘ

1. Do czego służy usługa WWW?
2. Opisz procedurę instalacji serwera WWW.
3. Jakim poleceniem można uruchomić serwer WWW?

43.4. Konfiguracja serwera Apache

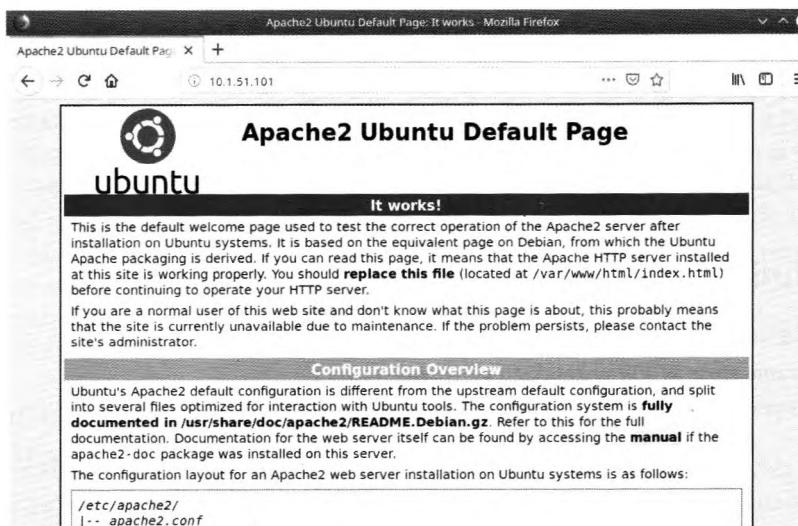
ZAGADNIENIA

- Najważniejsze pliki konfiguracyjne serwera httpd
- Zainstalowanie i uruchomienie serwera Apache2

Pliki konfiguracyjne serwera są zlokalizowane w katalogu **/etc/apache2**. Głównym plikiem konfiguracyjnym jest **apache2.conf**, w którym oprócz opcji konfiguracyjnych znajdują się odnośniki do innych plików konfiguracyjnych, np. **httpd.conf**, **ports.conf**. W pliku **httpd.conf** będą umieszczane wpisy użytkownika, plik **ports.conf** pozwala na zdefiniowanie wirtualnych serwerów. Wszystkie linie rozpoczynające się od znaku # są komentarzami, które nie mają wpływu na działanie serwera, a tylko wyjaśniają, do czego służą poszczególne opcje. W opcjach plików konfiguracyjnych litery małe i duże są rozróżniane. W niektórych przypadkach konieczne będzie nadanie nazwy serwerowi. W tym celu w pliku konfiguracyjnym **/etc/apache2/sites-available/000-default.conf** należy w opcji **ServerName** wprowadzić nazwę serwera, np.

ServerName zspieradz.edu.pl

Po ponownym uruchomieniu serwera można wykonać próbę nawiązania połączenia: w oknie adresu przeglądarki wpisujemy adres IP lub nazwę komputera (rys. 43.1).



Rys. 43.1. Strona startowa serwera Apache

PRZYKŁAD 43.1

Instalowanie i uruchomienie serwera Apache2

Aby zainstalować i uruchomić serwer Apache2, należy:

1. Zalogować się na konto użytkownika.
2. Zainstalować serwer, w tym celu trzeba posłużyć się poleceniem:

sudo apt install apache2.

3. Sporządzić kopie zapasowe plików konfiguracyjnych, np. `/etc/apache2/apache2.conf`, `/etc/apache2/sites-available/000-default.conf` oraz innych przed ich modyfikacją za pomocą polecień:

```
sudo cp /etc/apache2/apache2.conf /etc/apache2/apache2.conf.old
```

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

4. Zmodyfikować plik konfiguracyjny przez dopisanie nazwy serwera (podając nazwę własnego serwera).

5. Uruchomić ponownie usługę za pomocą polecenia:

```
sudo service apache2 restart.
```

6. Sprawdzić, czy usługa jest uruchomiona za pomocą polecenia:

```
nmap localhost.
```

7. Uruchomić przeglądarkę internetową i w pole adresu wpisać adres IP serwera WWW. Jeżeli konfiguracja jest poprawna, zostanie wyświetlona strona testowa serwera **Apache2**.

SPRAWDŹ SWOJE UMIEJĘTNOŚCI

1. Zainstaluj serwer WWW. Otwórz stronę testową serwera – w pole adresu przeglądarki internetowej wpisz adres IP lub nazwę serwera, np. localhost. Odczytaj tekst ze strony startowej.
2. Wyświetl zawartość pliku `apache2.conf` i sprawdź wartość zmiennej `Timeout`. Na podstawie opisu umieszczonego w pliku wyjaśnij, do czego służy ta zmienna.
3. Wyświetl zawartość pliku `ports.conf` i sprawdź wartość zmiennej `Listen`. Wyjaśnij, do czego służy ta zmienna.

43.5. Umieszczanie stron na serwerze WWW

ZAGADNIENIA

- Umieszczanie stron WWW w katalogu głównym serwera
- Systemy operacyjne rozróżniające wielkości liter w nazwach plików i folderów

Położenie głównego katalogu zawierającego strony udostępniane użytkownikom jest określone w pliku konfiguracyjnym `/etc/apache2/sites-available/default` za pomocą zmiennej `DocumentRoot`, np. `DocumentRoot /var/www/html`

Jeżeli w tym katalogu umieści się plik `index.html` lub inny znajdujący się na liście domyślnie uruchamianych, to zostanie on wczytany i wyświetlony w oknie przeglądarki.

PRZYKŁAD 43.2

Umieszczanie stron WWW w katalogu głównym usługi

Aby umieścić stronę WWW w katalogu głównym usługi, należy:

1. Zalogować się na konto użytkownika.
2. W pliku konfiguracyjnym zlokalizować położenie katalogu głównego zawierającego strony WWW (opcja `DocumentRoot`).
3. Przygotować plik `index.html`, zawierający dokument w języku HTML.

4. Za pomocą usługi FTP skopiować plik do katalogu głównego usługi WWW.
5. Uruchomić przeglądarkę internetową i w pole adresu wpisać adres IP serwera WWW. Jeżeli konfiguracja jest poprawna, zostanie wyświetlona strona zawierająca przygotowany wcześniej dokument (rys. 43.2).



Rys. 43.2. Zmodyfikowana strona testowa serwera Apache

43.6. Wielkość liter, nazwy plików

ZAGADNIENIA

■ Rozróżnianie wielkości liter

Systemy operacyjne z rodziny Windows nie rozróżniają wielkości liter w nazwach plików i folderów. Ponadto systemy te nie zachowują konsekwencji w wyświetlaniu nazw (np. zamieniają pierwszą literę małą na dużą, ukrywając rozszerzenia nazw plików). Może to powodować pewne utrudnienia w korzystaniu z usług internetowych. Większość serwerów w internecie pracuje na podstawie systemów Unix/Linux. W tych systemach w nazwach plików i folderów są rozróżniane litery małe i duże (np. plik **index.html** i **INDEX.HTML** to zupełnie różne pliki). Ponieważ nie możemy przewidzieć, czy nasza witryna zostanie umieszczona na serwerze z systemem Windows czy Linux, aby uniknąć niespodzianek, należy ścisłe przestrzegać reguł dotyczących wielkości liter w nazwach.

43.7. Konfiguracja serwera Apache do obsługi stron użytkowników

ZAGADNIENIA

- Włączenie udostępniania katalogów domowych użytkowników
- Konfigurowanie serwera Apache2 do udostępniania stron domowych użytkowników

Użytkownicy lokalni mają na serwerze konta założone przez administratora i posiadają uprawnienia do zapisywania danych w swoich katalogach domowych. W katalogu domowym użytkownika może być utworzony podkatalog **public_html**, który będzie udostępniany za pośrednictwem usługi WWW. Za obsługę stron użytkowników odpowiada zmienna **UserDir**. Gotowe skrypty zawierające ustawienia konfiguracyjne są umieszczone w katalogu **mods-available**, w którym znajduje się wiele gotowych skryptów, możliwych do wykorzystania (ale nie wszystkie z nich są wykorzystywane). Aktywowanie skryptu odbywa się przez dołączenie go do listy. W tym celu w katalogu **mods-enabled** należy

utworzyć linki do odpowiednich skryptów. W ten sposób spośród dużej liczby przygotowanych skryptów można wybrać tylko te, które są potrzebne i mają być aktywowane. Włączenie udostępnianie katalogów domowych użytkowników można za pomocą poleceń:

```
cd /etc/apache2/mods-enabled  
sudo ln -s ../mods-available/userdir.conf userdir.conf  
sudo ln -s ../mods-available/userdir.load userdir.load
```

W pierwszym poleceniu zmieniamy katalog na **/etc/apache2/mods-enabled**. Polecenie drugie tworzy w katalogu bieżącym (**mods-enabled**) link symboliczny (**ln -s**) o nazwie **userdir.conf** do pliku **../mods-available/userdir.conf** (zapis **../mods-available/userdir.conf** oznacza, że należy wyjść o jeden poziom wyżej w strukturze katalogów i wejść do katalogu **mods-available**). Link ten aktywuje zmienne odpowiedzialne za udostępnianie katalogów domowych użytkowników. W ostatnim poleceniu jest tworzony link symboliczny do skryptu odpowiedzialnego za załadowanie modułu programu Apache2, zawierającego obsługę wymaganych funkcji.

PRZYKŁAD 43.3

Konfiguracja serwera Apache2 do udostępniania stron domowych użytkowników

Użytkownicy lokalni mogą udostępniać własne strony po wykonaniu poniższej procedury.

1. Zalogować się na konto użytkownika.
2. Utworzyć w katalogu domowym użytkownika podkatalog **public_html** za pomocą polecenia:

```
mkdir ~/public_html.
```

Powtórzyć tę operację dla każdego użytkownika, który będzie miał prawo udostępniania swojego katalogu.

3. Utworzyć w katalogu **mods-enabled** linki do plików konfiguracyjnych **userdir.conf** i **userdir.load**.
4. Utworzyć plik **index.html** w katalogu **public_html**.
5. Uruchomić ponownie serwer za pomocą polecenia:

```
sudo service apache2 restart.
```

6. Przetestować działanie usługi (rys. 43.3).



Rys. 43.3. Strona udostępniana przez użytkownika

Aby wyświetlić zawartość strony użytkownika, w pole adresu przeglądarki należy za adresem serwera wpisać **/~nazwa_uzytkownika**, np. **10.1.51.101/~uczen1**.

SPRAWDŹ SWOJE UMIEJĘTNOŚCI

- 1 Utwórz przykładową stronę w języku HTML zawierającą twoje imię i nazwisko.
- 2 Za pomocą protokołu FTP skopiuj stronę do katalogu **public_html** w swoim katalogu domowym.
- 3 Przetestuj działanie strony. Skorzystaj z własnego komputera (podaj nazwę komputera – localhost) oraz komputera koleżanki/kolegi (podaj adres IP komputera).

SPRAWDŹ SWOJĄ WIEDZĘ

- 1 Jaka zmienna wskazuje nazwę katalogu zawierającego strony domowe użytkowników?
- 2 Opisz procedurę konfiguracji serwera Apache2 do udostępniania stron domowych użytkowników.

43.8. Konfigurowanie wirtualnych serwerów WWW

ZAGADNIENIA

- Wirtualne serwery WWW
- Korzyści konfigurowania wirtualnych serwerów WWW
- Przebieg procedury konfigurowania wirtualnych serwerów WWW

Mechanizm wirtualnych serwerów umożliwia uruchomienie więcej niż jednej witryny internetowej na pojedynczym komputerze, np. można uruchomić stronę szkoły i samorządu uczniowskiego. Wirtualne serwery mogą być rozróżniane za pomocą adresów IP (ang. *IP-based*), oznacza to, że dla każdej witryny jest przedzielany inny adres IP, za pomocą portów (ang. *port-based*), oznacza to, że dla każdej witryny jest przedzielany inny numer portu lub za pomocą nazw (ang. *name-based*) – pojedynczy adres IP jest wykorzystywany przez wiele witryn. Udostępnianie wielu witryn na pojedynczym serwerze umożliwia obniżenie kosztów instalacji i administracji (zamiast kupować, instalować i administrować odrębnym serwerem dla każdej witryny, wystarczy jeden serwer obsługujący wszystkie witryny).

PRZYKŁAD 43.4

Konfigurowanie i testowanie wirtualnych serwerów bazujących na adresach IP

W rozwiązaniu tym konieczne jest skonfigurowanie innego adresu IP dla każdej udostępnianej witryny. W tym przykładzie serwer zostanie skonfigurowany do obsługi witryny szkoły (**10.1.51.101**), samorządu szkolnego (**10.1.51.121**) i szkolnego koła sportowego (**10.1.51.141**).

Aby skonfigurować wirtualne serwery, należy:

1. Skonfigurować interfejs sieciowy tak, aby obsługiwał wiele adresów IP (aliasy adresów IP). W tym celu dla każdego dodawanego adresu należy wydać polecenie:

```
sudo ip addr add 10.1.51.121 dev enp5s2.
```

W poleceniu tym do interfejsu enp5s2 jest przypisywany dodatkowy adres **10.1.51.121**. Aby sprawdzić, czy adresy zostały poprawnie dodane, można wyświetlić listę adresów za pomocą polecenia:

```
sudo ip addr show.
```

oraz przetestować adres za pomocą polecenia **ping**. Sposób wykonania tych operacji pokazano na rys. 43.4.



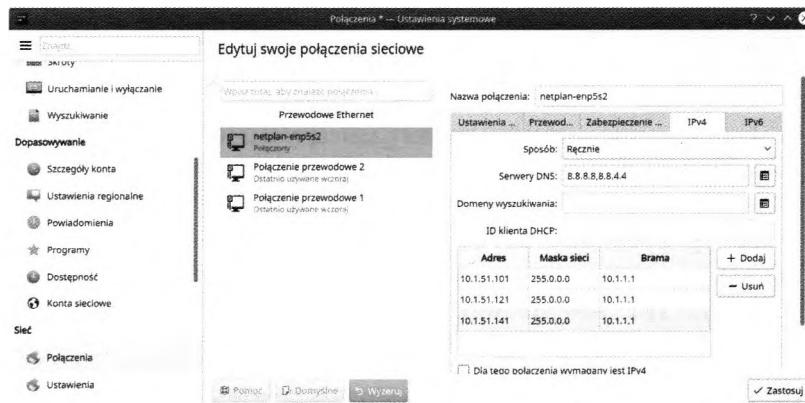
```

uczen : bash — Konsola
Plik Edycja Widok Zakładki Ustawienia Pomoc
::: $ sudo ip addr add 10.1.51.121 dev enp5s2
[sudo] hasło użytkownika uczen:
::: $ sudo ip addr add 10.1.51.141 dev enp5s2
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 brd 127.255.255.255 scope host
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: enp5s2: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc fq_codel state DOWN group default qlen 1000
    link/ether 00:22:15:17:c8:a9 brd ff:ff:ff:ff:ff:ff
    inet 10.1.51.101/24 brd 10.1.51.255 scope global dynamic noprefixroute enp5s2
        valid_lft 75755sec preferred_lft 75755sec
        inet 10.1.51.121/32 scope global enp5s2
            valid_lft forever preferred_lft forever
        inet 10.1.51.141/32 scope global enp5s2
            valid_lft forever preferred_lft forever
        inet6 fe80::222:15ff:fe17:dd75/64 scope link
            valid_lft forever preferred_lft forever
::: $ ping 10.1.51.101
PING 10.1.51.101 (10.1.51.101) 56(84) bytes of data.
64 bytes from 10.1.51.101: icmp_seq=1 ttl=64 time=0.045 ms
64 bytes from 10.1.51.101: icmp_seq=2 ttl=64 time=0.040 ms
^C
--- 10.1.51.101 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1016ms
rtt min/avg/max/mdev = 0.040/0.042/0.045/0.007 ms
::: $ ping 10.1.51.121
PING 10.1.51.121 (10.1.51.121) 56(84) bytes of data.
64 bytes from 10.1.51.121: icmp_seq=1 ttl=64 time=0.037 ms
64 bytes from 10.1.51.121: icmp_seq=2 ttl=64 time=0.042 ms
^C
--- 10.1.51.121 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1011ms
rtt min/avg/max/mdev = 0.037/0.039/0.042/0.006 ms
::: $ ping 10.1.51.141
PING 10.1.51.141 (10.1.51.141) 56(84) bytes of data.
64 bytes from 10.1.51.141: icmp_seq=1 ttl=64 time=0.040 ms
64 bytes from 10.1.51.141: icmp_seq=2 ttl=64 time=0.042 ms
^C
--- 10.1.51.141 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1003ms
rtt min/avg/max/mdev = 0.040/0.041/0.042/0.001 ms
::: $

```

Rys. 43.4. Dodawanie i testowanie aliasów IP

Wprowadzone adresy będą aktywne do czasu ponownego uruchomienia systemu. Aby dodać je w sposób trwały, można skorzystać z narzędzia **Połączenia/Ustawienia systemowe**, uruchamianego przez **Start/Ustawienia/Ustawienia systemowe**. W oknie **Sieć/Połączenia** należy wybrać zakładkę **Przewodowe Ethernet** i wskazać połączenie. W zakładce **IPv4** ustawić sposób konfiguracji na **Ręcznie** oraz dodać adresy. Po wprowadzeniu dodatkowych adresów (rys. 43.5) kliknąć przycisk **Zastosuj** i w razie konieczności wprowadzić hasło administratora.



Rys. 43.5. Wprowadzanie dodatkowych adresów IP

2. Utworzyć katalogi, w których będą przechowywane pliki stron. Można to zrobić za pomocą polecen:

```
sudo mkdir /var/www/samorzad
sudo mkdir /var/www/sks
```

3. Przydzielić uprawnienia do katalogów i zmienić ich właściciela, np. za pomocą polecen:

```
sudo chown -R uczen:uczen /var/www/samorzad
sudo chown -R uczen:uczen /var/www/sks
sudo chmod -R 755 /var/www/samorzad
sudo chmod -R 755 /var/www/sks
```

4. Utworzyć przykładowe strony dla samorządu i SKS, np. dla witryny samorządu utworzyć plik **index.html** za pomocą polecenia:

```
sudo vi /var/www/samorzad/index.html
```

i wpisać zawartość

```
<html>
  <head>
    <title>Samorzad ZSP</title>
  </head>
  <body>
    <h1>Strona Samorządu Szkolnego</h1>
  </body>
</html>
```

5. Utworzyć pliki konfiguracyjne wirtualnych serwerów. Można skopiować przykładowy plik konfiguracyjny wirtualnego serwera, a następnie go zmodyfikować

```
sudo cp /etc/apache2/sites-available/000-default.conf
/etc/apache2/sites-available/samorzad.conf
sudo cp /etc/apache2/sites-available/000-default.conf
/etc/apache2/sites-available/sks.conf
```

6. Wykonać konfigurację wirtualnych serwerów. Otworzyć plik konfiguracyjny witryny samorządu za pomocą polecenia:

```
sudo vi /etc/apache2/sites-available/samorzad.
```

Wprowadzić w pliku następujące zmiany:

- w wierszu **<VirtualHost *:80>** symbol "*" zamienić na adres IP wirtualnego serwera, np. **<VirtualHost 10.1.51.121:80>**
- określić położenie katalogu zawierającego witrynę

```
DocumentRoot /var/www/samorzad
```

Przykładowy plik z wprowadzonymi zmianami pokazano na rys. 43.6.

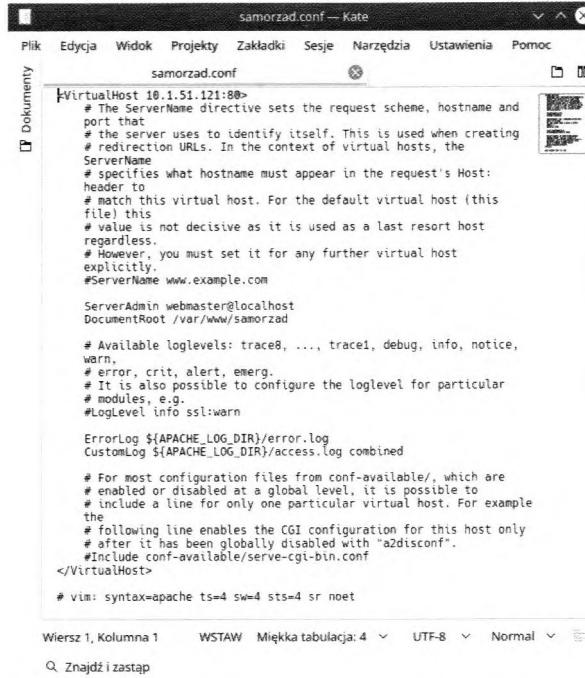
W analogiczny sposób zmodyfikować plik konfiguracyjny witryny SKS.

7. Dokonać aktywacji witryny przez utworzenie odpowiednich skrótów, wykonując polecenia:

```
sudo a2ensite samorzad
sudo a2ensite sks.
```

8. Uruchomić ponownie serwer **Apache2** poleceniem:

```
sudo service apache2 restart.
```



```

VirtualHost 10.1.51.121:80>
    # 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
    # reach 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/samorzad

    # 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

```

Wiersz 1, Kolumna 1 WSTAW Miękka tabulacja: 4 UTF-8 Normal

Q Znajdź i zastąp

Rys. 43.6. Plik konfiguracji wirtualnego serwera opartego na adresach IP

9. Przetestować działanie wirtualnych serwerów przez uruchomienie udostępnianych stron za pomocą adresów IP wirtualnych serwerów (rys. 43.7).



Rys. 43.7. Testowanie działania wirtualnych serwerów rozróżnianych przez adresy IP

PRZYKŁAD 43.5

Konfigurowanie i testowanie wirtualnych serwerów opierających się na nazwach (name-based)

W rozwiążaniu tym konieczne jest skonfigurowanie innej nazwy dla każdej udostępnianej witryny. W tym przykładzie serwer zostanie skonfigurowany do obsługi witryny szkoły (**szkola.zsp.local**), samorządu szkolnego (**samorzad.zsp.local**) i szkolnego koła sportowego (**sko.zsp.local**). Aby strony mogły być uruchamiane z wszystkich komputerów w sieci lokalnej, należy w konfiguracji serwera DNS dodać wpisy (rekordy A lub aliasy) łączące nazwy **szkola.zsp.local**, **sko.zsp.local** i **samorzad.zsp.local** z adresem serwera.

Aby skonfigurować wirtualne serwery, należy tak jak w poprzednim ćwiczeniu:

1. Utworzyć katalogi, w których będą przechowywane pliki stron.
2. Przydzielić uprawnienia do katalogów.
3. Utworzyć przykładowe strony dla samorządu i SKS.
4. Utworzyć pliki konfiguracyjne wirtualnych serwerów.
5. Wykonać konfigurację wirtualnych serwerów. Otworzyć plik konfiguracyjny witryny samorządu za pomocą polecenia:

```
sudo vi /etc/apache2/sites-available/samorzad.
```

Wprowadzić w pliku następujące zmiany:

- określić dla wirtualnego serwera nazwę (ang. *ServerName*) i alias nazwy (ang. *ServerAlias*), w opcjach wprowadzić

```
ServerName samorzad.zsp.local  
ServerAlias www.samorzad.zsp.local
```

- określić położenie katalogu zawierającego witrynę

```
DocumentRoot /var/www/samorzad
```

Przykładowy plik z wprowadzonymi zmianami pokazano na rys. 43.8.

W analogiczny sposób zmodyfikować plik konfiguracyjny witryny SKS.

6. Aktywować witrynę przez utworzenie odpowiednich skrótów i uruchomić ponownie serwer **Apache2** (patrz: punkty 7. i 8. w poprzednim ćwiczeniu).
7. Przetestować działanie wirtualnych serwerów przez uruchomienie udostępnianych stron za pomocą nazw wirtualnych serwerów (rys. 43.9).

```

<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 samorzad.zsp.local
    ServerAlias www.samorzad.zsp.local

    ServerAdmin webmaster@localhost
    DocumentRoot /var/www/samorzad

    # 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>

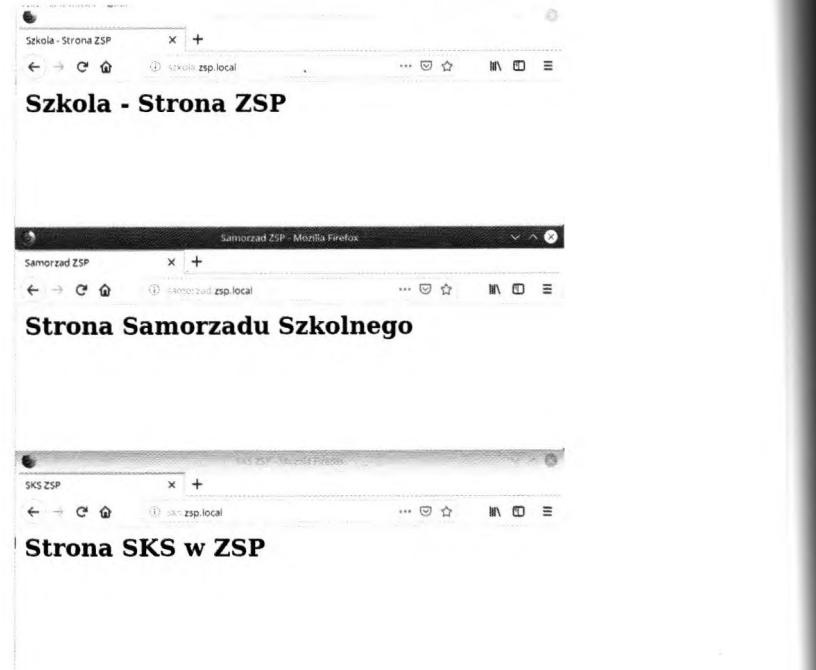
# vim: syntax=apache ts=4 sw=4 sts=4 sr noet

```

Wiersz 1, Kolumna 1 WSTAW Miękka tabulacja: 4 UTF-8 Normal

Znajdź następ

Rys. 43.8. Plik konfiguracyjny wirtualnego serwera opartego na nazwach



Rys. 43.9. Testowanie działania wirtualnych serwerów rozróżnianych przez nazwy

SPRAWDŹ SWOJE UMIEJĘTNOŚCI

1. Zaplanuj liczbę i nazwy dla wirtualnych serwerów w twojej szkole.
2. Utwórz system wirtualnych serwerów dla twojej szkoły i przetestuj ich działanie.

SPRAWDŹ SWOJĄ WIEDZĘ

1. Wyjaśnij, jaka jest różnica między działaniem wirtualnych serwerów opartych na adresach IP i nazwach.
2. Gdybyś była/będzie administratorem szkolnego serwera WWW, to czy utworzyłabyś/utworzyłbyś wirtualne serwery? Jeśli tak, to jakie? Uzasadnij decyzję.

43.9. Blokowanie przeglądania zawartości katalogu

ZAGADNIENIA

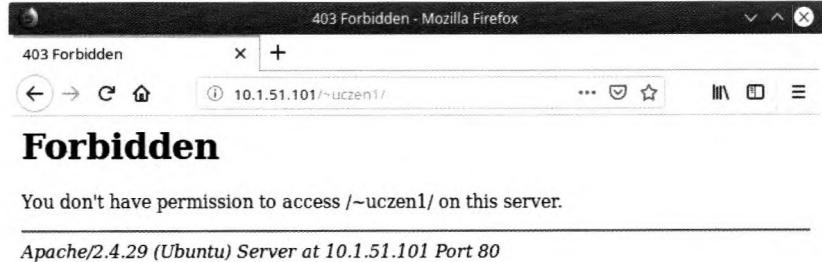
- Konfigurowanie serwera Apache2 do blokowania listowania zawartości katalogu zawierającego strony domowe użytkowników
- Wprowadzenie ochrony dostępu do udostępnianego katalogu za pomocą hasła

W serwerze **Apache2** standardowo jest włączone wyświetlanie zawartości katalogów domowych użytkownika. Dla użytkownika oznacza to, że jeżeli w pole adresu przeglądarki wpisze nazwę pliku, to zostanie wyświetlony wskazany plik. Jeżeli wpisze się ścieżkę do katalogu, zostanie wyświetlona zawartość pliku startowego (np. **index.html**). Jeżeli nie istnieje plik o nazwie domyślnej lub wskazanej przez użytkownika, jest wyświetlana zawartość katalogu (rys. 43.10).



Rys. 43.10. Odblokowane wyświetlanie zawartości katalogu

Opcja taka może być przydatna, gdy użytkownik chce udostępnić pliki za pomocą WWW oraz aby inni użytkownicy mogli sprawdzić nazwy plików. Wyświetlanie zawartości katalogów może stanowić zagrożenie bezpieczeństwa. Przez zmianę konfiguracji serwera można zmienić jego zachowanie tak, aby w takim przypadku wyświetlał komunikat o błędzie (rys. 43.11).



Rys. 43.11. Zablokowane wyświetlanie zawartości katalogu

W tym celu w pliku `/etc/apache2/mods-available/userdir.conf` należy odszukać fragment

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

W pliku konfiguracyjnym trzeba postawić znak `#` na początku każdej linii między znacznikami `<Directory /home/*public_html>` oraz `</Directory>`.

PRZYKŁAD 43.6

Konfigurowanie serwera Apache2 do blokowania listowania zawartości katalogu zawierającego strony domowe użytkowników.

Aby umożliwić blokowanie listowania zawartości katalogu zawierającego strony domowe użytkowników, należy:

1. Zalogować się na konto użytkownika.
2. Zmodyfikować plik konfiguracyjny `/etc/apache2/mods-available/user-dir.conf` zgodnie z opisem.
3. Usunąć plik `index.html` (lub zmienić jego nazwę, np. na `index.txt`) z katalogu `public_html`.
4. Uruchomić ponownie serwer za pomocą polecenia:
`sudo service apache restart.`

5. Przetestować działanie usługi.

Dostęp do katalogu lub całej witryny udostępnianej przez WWW można ograniczyć przez pliki `.htaccess` oraz `.htpasswd`. Metoda ta pozwala na ograniczenie dostępu do katalogu tylko dla osób znających login i hasło. Możliwość ta jest ważna w przypadku, gdy część opublikowanych informacji powinna być dostępna tylko dla uprawnionych użytkowników (np. oprogramowanie tylko dla klientów, dokumenty dla pracowników).

PRZYKŁAD 43.7

Konfigurowanie funkcji kontroli dostępu do udostępnianego katalogu

W tym przykładzie zostanie ograniczony dostęp do katalogu **dokumenty**. Tylko osoby znające nazwę konta i hasło będą mogły przeglądać jego zawartość. Aby skonfigurować konto i hasło niezbędne do uzyskania dostępu do katalogu, należy:

1. W katalogu, w którym są przechowywane pliki witryny szkoły, utworzyć podkatalog **dokumenty**, np. za pomocą polecenia:

```
sudo mkdir /var/www/html/dokumenty.
```

2. W katalogu **dokumenty** utworzyć plik **.htaccess** zawierający:

```
AuthName „Katalog z ograniczonym dostępem”
```

```
AuthType Basic
```

```
AuthUserFile /hasla/.htpasswd
```

```
Require valid-user
```

W pliku tym, w opcji **AuthName**, można wpisać dowolny tekst (np. informacje o ograniczeniu dostępu). W linii **AuthUserFile** należy podać pełną ścieżkę do pliku **.htpasswd**, w którym są przechowywane nazwy kont i hasła dostępu. Plik ten powinien zostać umieszczony w miejscu, do którego zwykły użytkownik nie będzie miał dostępu.

3. Utworzyć plik **.htpasswd** w miejscu wskazanym w pliku **.htaccess**. Plik **.htpasswd** będzie zawierał nazwy użytkowników oraz ich hasła dostępu. Zawartość pliku **.htpasswd** tworzy się według schematu:

```
nazwa_uzytkownika:haslo_dostepu
```

Hasło dostępu jest odpowiednio zakodowane funkcją **crypt()**, **MD5** lub **SHA-1**. Przykładowo, użytkownik **uczen1** może mieć hasło **Qwerty123**. W każdej linijce pliku wprowadza się informacje dotyczące jednego konta. Przykładowy plik zawierający konta użytkowników **uczen1**, **uczen2** i **uczen3** z hasłem **Qwerty123** zakodowanym odpowiednio funkcją MD5, SHA-1 i **crypt()**.

```
uczen1:$apr1$c/vGOHV$C/QQGFwte0rW1DUGlyW3L1
```

```
uczen2:{SHA}zJ+BakJDHPhSzcej+tQqb2X/zjQ=
```

```
uczen3:9NE3e1mWz/eSc
```

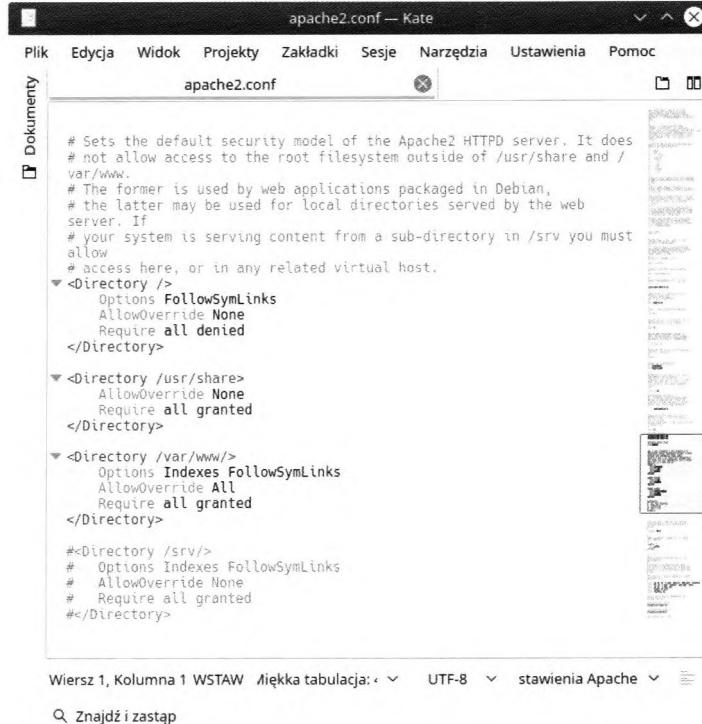
Plik z hasłami można utworzyć za pomocą polecenia:

```
sudo htpasswd -c /hasla/.htpasswd uczen1.
```

Hasła do pliku **.htpasswd** dla kont użytkowników można zakodować również za pomocą generatora haseł działającego pod adresem http://aspirine.org/htpasswd_en.html.

4. Zezwolić na nadpisywanie ustawień w pliku konfiguracyjnym witryny. Jeżeli zabezpieczany katalog jest częścią witryny głównej, należy zmodyfikować plik **/etc/apache2/apache2.conf**. Jeżeli katalog jest częścią wirtualnego hosta, należy zmodyfikować plik konfiguracyjny tego hosta. W pliku konfiguracyjnym trzeba odszukać w sekcję opisującą reguły dostępu do katalogu **/var/www** i zmienić opcję **AllowOverride None** na **AllowOverride All**.

Na rys. 43.12 pokazano plik konfiguracyjny i zaznaczono właściwy fragment. W konfiguracji zezwolono na nadpisywanie ustawień w katalogu **/var/www** oraz wszystkich katalogach podrzędnych.

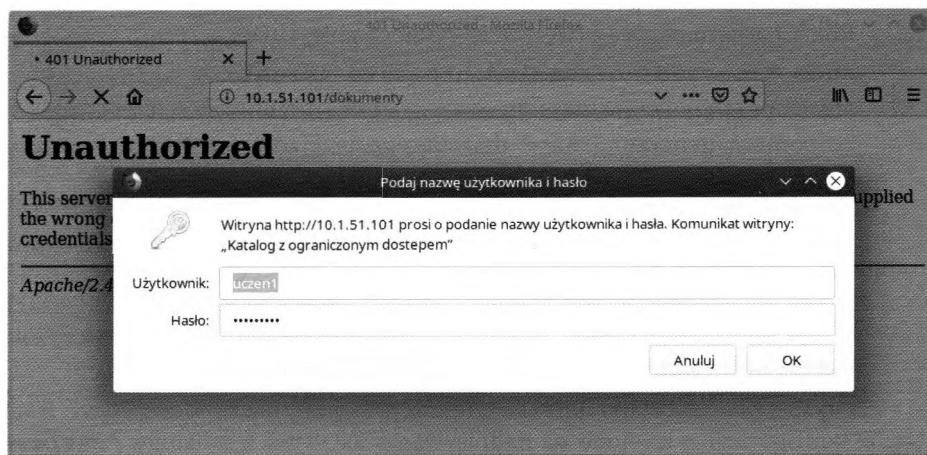


Rys. 43.12. Zezwolenie na nadpisywanie ustawień w katalogu

5. Uruchomić ponownie serwer apache, np. za pomocą polecenia:

sudo /etc/init.d/apache2 restart.

6. Przetestować działanie wprowadzonych zabezpieczeń. Jeżeli konfiguracja przebiegła poprawnie, to podczas próby uzyskania dostępu do zabezpieczonego katalogu pojawi się okno z żądaniem podania nazwy konta i hasła (rys. 43.13).



Rys. 43.13. Testowanie systemu ochrony katalogu za pomocą hasła

SPRAWDŹ SWOJĄ WIEDZĘ

1. Wyjaśnij, na czym polega zagrożenie bezpieczeństwa spowodowane możliwością listowania katalogów.
2. Podaj przykłady, w jakich sytuacjach ty, jako administratorka/administrator serwera WWW, wprowadziłbyś/wprowadziłbyś kontrolę dostępu do zawartości katalogu.

43.10. Zmiana domyślnie uruchamianego

ZAGADNIENIA

- Doinstalowanie modułu PHP5
- Skonfigurowanie serwera Apache2 do obsługi języka PHP

Jeżeli w pole adresu przeglądarki zostanie wpisana ścieżka katalogu, to we wskazanym katalogu będzie poszukiwany plik o domyślnej nazwie **index.html**. Za nazwę domyślnego pliku odpowiada zmienna **DirectoryIndex** w pliku **/etc/apache2/mods-enabled/dir.conf**, np.

```
DirectoryIndex index.php index.html default.html.
```

Po nazwie tej zmiennej można podać listę plików (oddzielonych spacjami wraz z rozszerzeniami), które mają być domyślnie uruchamiane przez przeglądarkę (ważna jest kolejność). Jeżeli zamierza się korzystać np. ze skryptów języka PHP, to warto dodać do tej listy nazwę **index.php**.

Jeżeli domyślna konfiguracja serwera WWW nie umożliwia obsługi stron zawierających skrypty PHP, można doinstalować moduł PHP5 za pomocą polecenia:

```
sudo apt install php php-cgi libapache2-mod-php php-common  
php-pear php-mbstring.
```

Polecenie to zainstaluje obsługę języka PHP5 oraz dodatkowe moduły.

PRZYKŁAD 43.8

Konfigurowanie serwera Apache2 do obsługi języka PHP

Aby umożliwić wykonywanie skryptów języka PHP na serwerze, należy:

1. Zalogować się na konto użytkownika.
2. Zainstalować moduł PHP5 za pomocą polecenia

```
sudo apt install php php-cgi libapache2-mod-php php-common  
php-pear php-mbstring.
```

3. Utworzyć plik **index.php** w katalogu **/var/www/html**.

Przykładowa zawartość pliku **index.php**

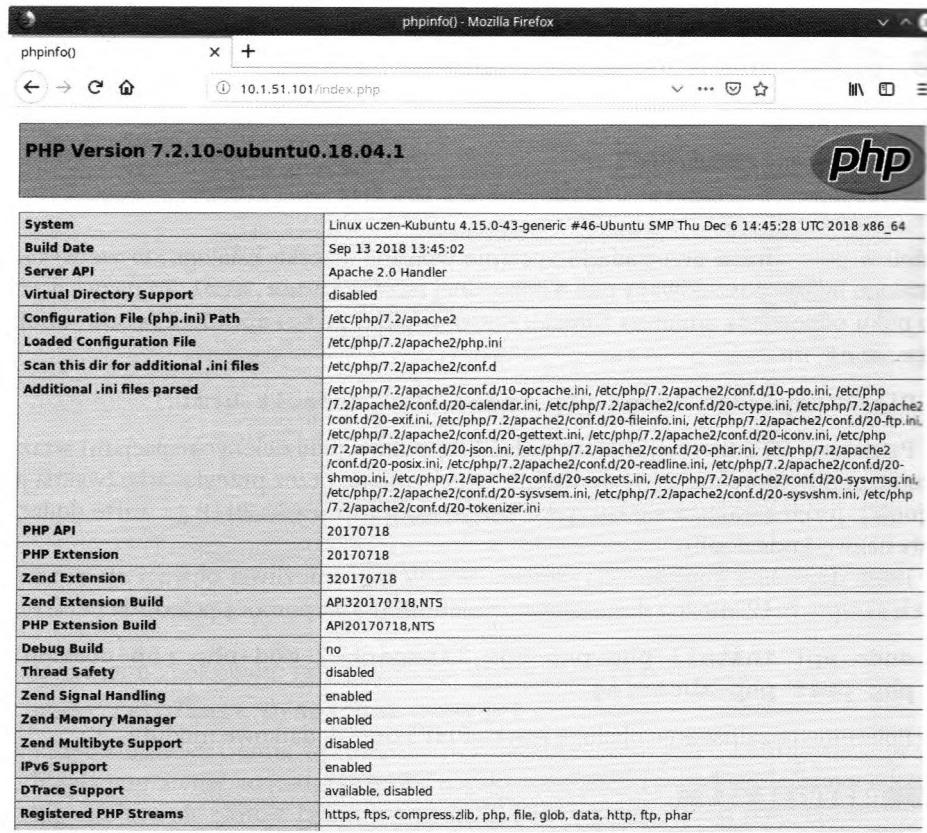
```
<html>  
<body>  
<?php phpinfo()  
?>  
</body>  
</html>
```

4. Zmodyfikować plik konfiguracyjny **etc/apache2/mods-enabled/dir.conf** zgodnie z opisem.

5. Uruchomić ponownie serwer za pomocą polecenia:

sudo service apache2 restart.

6. Przetestować działanie usługi. Jeżeli usługa działa właściwie, to po wczytaniu strony powinna się pojawić informacja o PHP, jak na rys. 43.14.



Rys. 43.14. Testowanie działania PHP na serwerze

SPRAWDŹ SWOJE UMIEJĘTNOŚCI

1. Zainstaluj moduł PHP w serwerze. Za pomocą skryptu języka PHP wyświetl informacje o zainstalowanej wersji PHP oraz położeniu pliku konfiguracyjnego **php.ini**.
2. Wyszukaj w internecie informacje o innych niż PHP językach skryptowych używanych do tworzenia stron internetowych.

SPRAWDŹ SWOJĄ WIEDZĘ

1. Do czego służy moduł PHP i jakie są jego zadania?
2. W jaki sposób doinstalować moduł PHP?
3. Opisz procedurę konfiguracji serwera Apache2 do obsługi języka PHP.

43.11. Przeglądanie logów serwera

ZAGADNIENIA

- Logi serwera
- Funkcje zmiennych ErrorLog oraz LogLevel
- Przeglądanie logów serwera

Serwer może być tak skonfigurowany, aby zapisywał wszystkie przypadki uzyskania dostępu do jego zasobów (ang. *access.log*) oraz wszystkie przypadki błędów (ang. *error.log*), np. gdy użytkownik wpisze nazwę pliku, który nie istnieje na serwerze. Ścieżki dostępu do plików zawierających logi błędów opisują zmienne **ErrorLog** oraz **LogLevel** w pliku **/etc/apache2/apache2.conf**. Zmienna **ErrorLog** wskazuje położenie plików logów, np. **ErrorLog \${APACHE_LOG_DIR}/error.log**.

Zmienna **LogLevel** kontroluje liczbę logowanych komunikatów. Może zawierać wartości (według malejącej liczby logowanych komunikatów): **debug, info, notice, warn, error, crit, alert, emerg**, np. **LogLevel warn**.

Przykłady plików logów z pliku **/var/log/apache2/error.log** pokazano na rys. 43.15.

```

apache2 : bash — Konsola
Plik Edycja Widok Zakładki Ustawienia Pomoc
Uczeń>ls -l /var/log/apache2/
Uczeń>cd /var/log/apache2
Uczeń>ls
access.log error.log other_vhosts_access.log
Uczeń>sudo tail error.log
[Sun Jan 06 13:48:31.681113 2019] [core:notice] [pid 9855:tid 140344966536128] AH00094: Command line: '/usr/sbin/apache2'
[Sun Jan 06 13:57:34.278088 2019] [mpm_event:notice] [pid 9855:tid 140344966536128] AH00491: caught SIGTERM, shutting down
[Sun Jan 06 13:57:34.367915 2019] [mpm_event:notice] [pid 10062:tid 140303338425280] AH00489: Apache/2.4.29 (Ubuntu) configured -- resuming normal operations
[Sun Jan 06 13:57:34.368153 2019] [core:notice] [pid 10062:tid 140303338425280] AH00094: Command line: '/usr/sbin/apache2'
[Sun Jan 06 14:13:08.266414 2019] [mpm_event:notice] [pid 10062:tid 140303338425280] AH00491: caught SIGTERM, shutting down
[Sun Jan 06 14:13:08.343243 2019] [mpm_prefork:notice] [pid 16941] AH00163: Apache/2.4.29 (Ubuntu) configured -- resuming normal operations
[Sun Jan 06 14:13:08.343468 2019] [core:notice] [pid 16941] AH00094: Command line: '/usr/sbin/apache2'
[Sun Jan 06 14:13:08.897192 2019] [mpm_prefork:notice] [pid 16941] AH00169: caught SIGTERM, shutting down
[Sun Jan 06 14:13:09.006943 2019] [mpm_prefork:notice] [pid 17002] AH00163: Apache/2.4.29 (Ubuntu) configured -- resuming normal operations
[Sun Jan 06 14:13:09.007041 2019] [core:notice] [pid 17002] AH00094: Command line: '/usr/sbin/apache2'
Uczeń>

```

Rys. 43.15. Przykładowy log z pliku **error.log**

Przykłady plików logów z pliku **/var/log/apache2/access.log** pokazano na rys. 43.16.

Jednym z podstawowych obowiązków administratora jest przeglądanie logów serwera. Z plików można dowiedzieć się o błędach występujących w kodach stron internetowych, np. o brakujących lub uszkodzonych plikach. Można również przeanalizować, które strony i pliki są pobierane i z jaką częstotliwością oraz jacy klienci i kiedy odwiedzali serwer.

```

Plik Edycja Widok Zakładki Ustawienia Pomoc
u:zyczek@kubu: ~ :/var/log/apache2$ sudo tail access.log
10.1.51.101 - uczen1 [06/Jan/2019:13:57:14 +0100] "GET /dokumenty/ HTTP/1.1" 200 658 "-" "Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:64.0) Gecko/20100101 Firefox/64.0"
10.1.51.101 - - [06/Jan/2019:13:57:14 +0100] "GET /icons/blank.gif HTTP/1.1" 304 179 "http://10.1.51.101/dokumenty/" "Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:64.0) Gecko/20100101 Firefox/64.0"
10.1.51.101 - - [06/Jan/2019:13:57:14 +0100] "GET /icons/back.gif HTTP/1.1" 304 179 "http://10.1.51.101/dokumenty/" "Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:64.0) Gecko/20100101 Firefox/64.0"
10.1.51.101 - uczen1 [06/Jan/2019:13:57:15 +0100] "GET /dokumenty/ HTTP/1.1" 200 657 "-" "Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:64.0) Gecko/20100101 Firefox/64.0"
10.1.51.101 - - [06/Jan/2019:13:57:15 +0100] "GET /icons/blank.gif HTTP/1.1" 304 179 "http://10.1.51.101/dokumenty/" "Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:64.0) Gecko/20100101 Firefox/64.0"
10.1.51.101 - - [06/Jan/2019:13:57:15 +0100] "GET /icons/back.gif HTTP/1.1" 304 179 "http://10.1.51.101/dokumenty/" "Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:64.0) Gecko/20100101 Firefox/64.0"
10.1.51.101 - - [06/Jan/2019:13:59:01 +0100] "GET /dokumenty HTTP/1.1" 401 743 "-" "Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:64.0) Gecko/20100101 Firefox/64.0"
10.1.51.101 - - [06/Jan/2019:13:59:40 +0100] "GET /favicon.ico HTTP/1.1" 404 503 "-" "Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:64.0) Gecko/20100101 Firefox/64.0"
10.1.51.101 - - [06/Jan/2019:13:59:49 +0100] "GET /dokumenty HTTP/1.1" 401 743 "-" "Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:64.0) Gecko/20100101 Firefox/64.0"
10.1.51.101 - - [06/Jan/2019:14:33:33 +0100] "GET /index.php HTTP/1.1" 200 21576 "-" "Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:64.0) Gecko/20100101 Firefox/64.0"
u:zyczek@kubu: ~ :/var/log/apache2$ 

```

Rys. 43.16. Przykładowy log z pliku **access.log**

SPRAWDŹ SWOJE UMIEJĘTNOŚCI

- Skonfiguruj serwer WWW, umieść na nim przynajmniej jedną stronę i przetestuj działanie usługi.
- Włącz udostępnianie stron WWW użytkowników i przetestuj działanie serwera.
- Zablokuj możliwość przeglądania zawartości katalogów zawierających strony użytkowników i przetestuj działanie serwera.
- Przetestuj obsługę skryptów PHP na twoim serwerze, dołącz plik **index.php** do listy przeszukiwanych plików.
- Przeanalizuj logi serwera powstałe podczas testowania serwera. Na ich podstawie określ, które pliki były najczęściej pobierane i czy w czasie pracy serwera pojawiły się błędy. Jeśli tak, to czym były spowodowane.