

## Jak zainstalować i skonfigurować serwer WWW Apache na Ubuntu

### Co to jest serwer WWW Apache?

Apache lub serwer HTTP Apache to darmowy serwer WWW typu open source, opracowany i obsługiwany przez Apache Software Foundation. Jego popularność można ocenić po tym, że około 46% stron internetowych na całym świecie jest obsługiwanych przez Apache. Apache umożliwia programistom witryn internetowych udostępnianie treści w Internecie. Służy jako dostawca, dostarczając pliki żądane przez użytkowników, gdy wprowadzają nazwę domeny w pasku adresu przeglądarki.

Ten samouczek dotyczy instalacji i konfiguracji Apache2 w systemie Ubuntu. Polecenia i procedury wymienione w tym artykule zostały uruchomione w systemie Ubuntu 18.04 LTS. Ponieważ używamy wiersza polecenia Ubuntu, terminala, w tym artykule; możesz go otworzyć poprzez Dash systemu lub skrót **Ctrl + Alt + T**.

### Zainstaluj Apache 2 na Ubuntu Linux

Wykonaj następujące kroki, aby zainstalować oprogramowanie Apache2 za pośrednictwem oficjalnych repozytoriów Ubuntu.

### NAJNOWSZE KOMENTARZE

---

**Sylwester** → [Jak zresetować Ubuntu](#)

**gonzoo** → [Jak włączyć / wyłączyć automatyczne logowanie w Debianie 10](#)

**KoxWUbuntu** → [Jak zainstalować Vanilla Gnome Desktop na Ubuntu](#)

**Konrad** → [Jak zainstalować i skonfigurować serwer LAMP na Ubuntu](#)

**jest dobrze** → [Dodawaj konta użytkowników i zarządzaj nimi w Ubuntu 18.04 LTS](#)

**Szaman** → [Jak używać Vokoscreen do nagrywania filmów na Ubuntu](#)

**Andrzej** → [Jak zostać rootem w linii poleceń Ubuntu za pomocą su i sudo](#)

**dzięki** → [Jak odinstalować programy z systemu Ubuntu](#)

**Traf** → [3 sposoby na wyłączenie / wyłączenie systemu Ubuntu](#)

**Dzięki za pomoc!** → [Jak otwierać i edytować pliki i foldery w Ubuntu Desktop jako administrator](#)

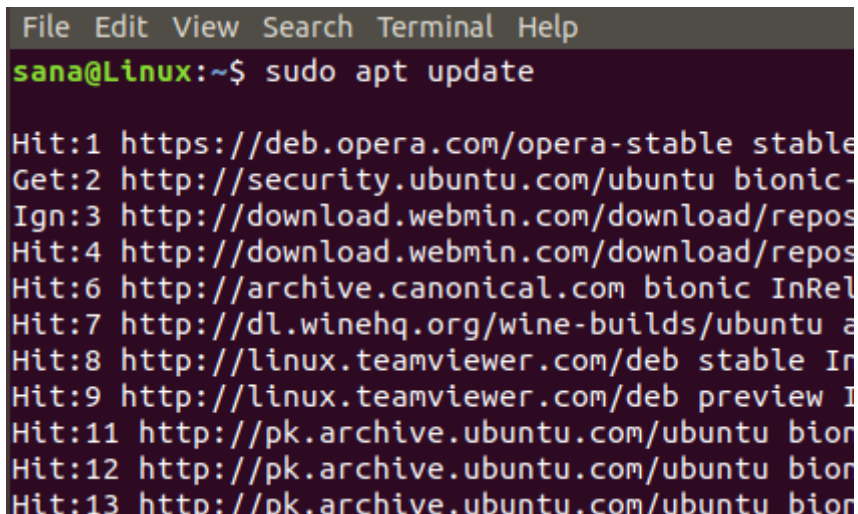
### POPULARNE POSTY

---

## Krok 1: Zaktualizuj repozytoria systemowe

Możesz pobrać najnowszą wersję oprogramowania, najpierw aktualizując indeks lokalnych pakietów repozytoriów Ubuntu. Otwórz Terminal i wprowadź w tym celu następujące polecenie:

```
$ sudo apt update
```



```
File Edit View Search Terminal Help
sana@Linux:~$ sudo apt update
Hit:1 https://deb.opera.com/opera-stable stable
Get:2 http://security.ubuntu.com/ubuntu bionic-
Ign:3 http://download.webmin.com/download/repos
Hit:4 http://download.webmin.com/download/repos
Hit:6 http://archive.canonical.com bionic InRel
Hit:7 http://dl.winehq.org/wine-builds/ubuntu a
Hit:8 http://linux.teamviewer.com/deb stable In
Hit:9 http://linux.teamviewer.com/deb preview I
Hit:11 http://pk.archive.ubuntu.com/ubuntu bion
Hit:12 http://pk.archive.ubuntu.com/ubuntu bion
Hit:13 http://pk.archive.ubuntu.com/ubuntu bion
```

## Krok 2: Zainstaluj Apache 2 za pomocą polecenia apt

Następnie wprowadź następujące polecenie jako sudo, aby zainstalować Apache2 i jego wymagane zależności:

```
$ sudo apt install apache2
```

Użyj skrótu Ctrl + Alt + Del do zamknięcia systemu Ubuntu

Jak zainstalować i używać języka programowania R w systemie Ubuntu 18.04 LTS

Jak zainstalować Audacious Audio Player na Ubuntu

Jak wyświetlić zapisane hasła WiFi w Ubuntu

Słuchaj darmowej muzyki dzięki Spotify na Ubuntu

Jak skompresować pliki / foldery do formatu .zip, tar.xz i 7z w Ubuntu Desktop

Jak zainstalować i skonfigurować serwer WWW Apache na Ubuntu

Jak wykonać pliki .bin i .run w Ubuntu

Jak zainstalować Apache Maven na Ubuntu 18.04 LTS

Jak zainstalować usługę Gogs GIT w systemie Ubuntu 18.04 LTS

Jak zainstalować Matomo Web Analytics na Ubuntu 18.04 LTS

Zabezpiecz Ubuntu za pomocą ClamAV Antivirus

```
sana@Linux:~$ sudo apt install apache2
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are n
  gyp libhttp-parser2.7.1 libjs-async libjs-inherits libjs-jq
node-ansi node-ansi-color-table node-archy node-balanced-ma
node-forever-agent node-fs.realpath node-github-url-from-gi
node-isexe node-lockfile node-mime node-mute-stream node-no
node-qs node-read node-retry node-semver node-sha node-slid
node-tunnel-agent node-underscore node-validate-npm-package
Use 'sudo apt autoremove' to remove them.
Suggested packages:
  apache2-doc apache2-suexec-pristine | apache2-suexec-custom
The following NEW packages will be installed:
  apache2
0 upgraded, 1 newly installed, 0 to remove and 169 not upgrad
```

Może pojawić się monit z opcją y / n, aby kontynuować instalację. Wpisz Y, po czym rozpocznie się procedura instalacji.

### Krok 3: Sprawdź instalację Apache

Po zakończeniu instalacji możesz sprawdzić numer wersji, a tym samym sprawdzić, czy Apache2 jest rzeczywiście zainstalowany w systemie, wprowadzając następujące polecenie:

```
$ apache2 -version
```

Zainstaluj i skonfiguruj Wildfly (JBoss) na Ubuntu 18.04 LTS

Jak sprawdzić zainstalowaną pamięć RAM w systemie Ubuntu

Jak zainstalować Android Studio IDE na Ubuntu 18.04 LTS

```
sana@Linux:~$ apache2 -version
Server version: Apache/2.4.29 (Ubuntu)
Server built:   2018-10-03T14:41:08
```

## Skonfiguruj ustawienia zapory

Aby skonfigurować Apache, najpierw musimy zezwolić na zewnętrzny dostęp do niektórych portów internetowych naszego systemu i zezwolić Apache na twojej zaporze UFW.

### Krok 1: Wymień profile aplikacji UFW

Aby skonfigurować zaporę, wypiszmy najpierw profile aplikacji, które będą nam potrzebne, aby umożliwić dostęp do Apache. Użyj poniższego polecenia, aby wyświetlić listę takich dostępnych aplikacji:

```
$ sudo ufw app list
```

```
sana@Linux:~$ sudo ufw app list
Available applications:
  Apache
  Apache Full
  Apache Secure
  CUPS
```

W powyższym pliku wyjściowym możesz zobaczyć trzy profile Apache, z których każdy zapewnia różny poziom bezpieczeństwa; Apache jest tym, który zapewnia maksymalne ograniczenie przy wciąż otwartym porcie 80.

## Krok 2: Zezwól Apache na UFW i sprawdź jego status

Zezwolenie Apache na UFW otworzy port 80 dla ruchu sieciowego, zapewniając jednocześnie maksymalne bezpieczeństwo serwera. Skonfiguruj UFW, aby zezwolić Apache za pomocą następującego polecenia:

```
$ sudo ufw allow 'Apache'
```

```
sana@Linux:~$ sudo ufw allow 'Apache'  
Rule updated  
Rule updated (v6)
```

Status UFW wyświetli teraz włączony Apache w zaporze.

```
$ sudo ufw status
```

## Skonfiguruj ustawienia serwera WWW Apache

### Krok 1: Sprawdź, czy usługa Apache działa

Pierwszym krokiem jest sprawdzenie, czy usługa Apache2 jest uruchomiona w systemie za pomocą następującego polecenia :

```
$ sudo systemctl status apache2
```

```
sana@Linux:~$ sudo systemctl status apache2
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset: enabled)
   Drop-In: /lib/systemd/system/apache2.service.d
            └─apache2-systemd.conf
   Active: active (running) since Wed 2018-10-31 11:33:20 PKT; 1min 45s ago
     Process: 28804 ExecStop=/usr/sbin/apachectl stop (code=exited, status=0/SUCCESS)
     Process: 28560 ExecReload=/usr/sbin/apachectl graceful (code=exited, status=0/SUCCESS)
     Process: 28809 ExecStart=/usr/sbin/apachectl start (code=exited, status=0/SUCCESS)
   Main PID: 28837 (apache2)
     Tasks: 6 (limit: 4580)
   CGroup: /system.slice/apache2.service
           └─28837 /usr/sbin/apache2 -k start
             └─28842 /usr/sbin/apache2 -k start
               └─28843 /usr/sbin/apache2 -k start
                 └─28844 /usr/sbin/apache2 -k start
                   └─28845 /usr/sbin/apache2 -k start
                     └─28846 /usr/sbin/apache2 -k start

Oct 31 11:33:20 Linux systemd[1]: Starting The Apache HTTP Server...
Oct 31 11:33:20 Linux systemd[1]: Started The Apache HTTP Server.
```

Status „aktywny (uruchomiony) sprawdza, czy usługa apache2 jest uruchomiona.

## Krok 2: Sprawdź, czy Apache działa poprawnie i nasłuchuje na twoim adresie IP

Możesz również sprawdzić, czy Apache działa, żądając strony z serwera Apache. W tym celu możesz użyć adresu IP swojego serwera, aby uzyskać dostęp do strony docelowej Apache.

Użyj następującego polecenia, aby dowiedzieć się o adresie IP swojego serwera:

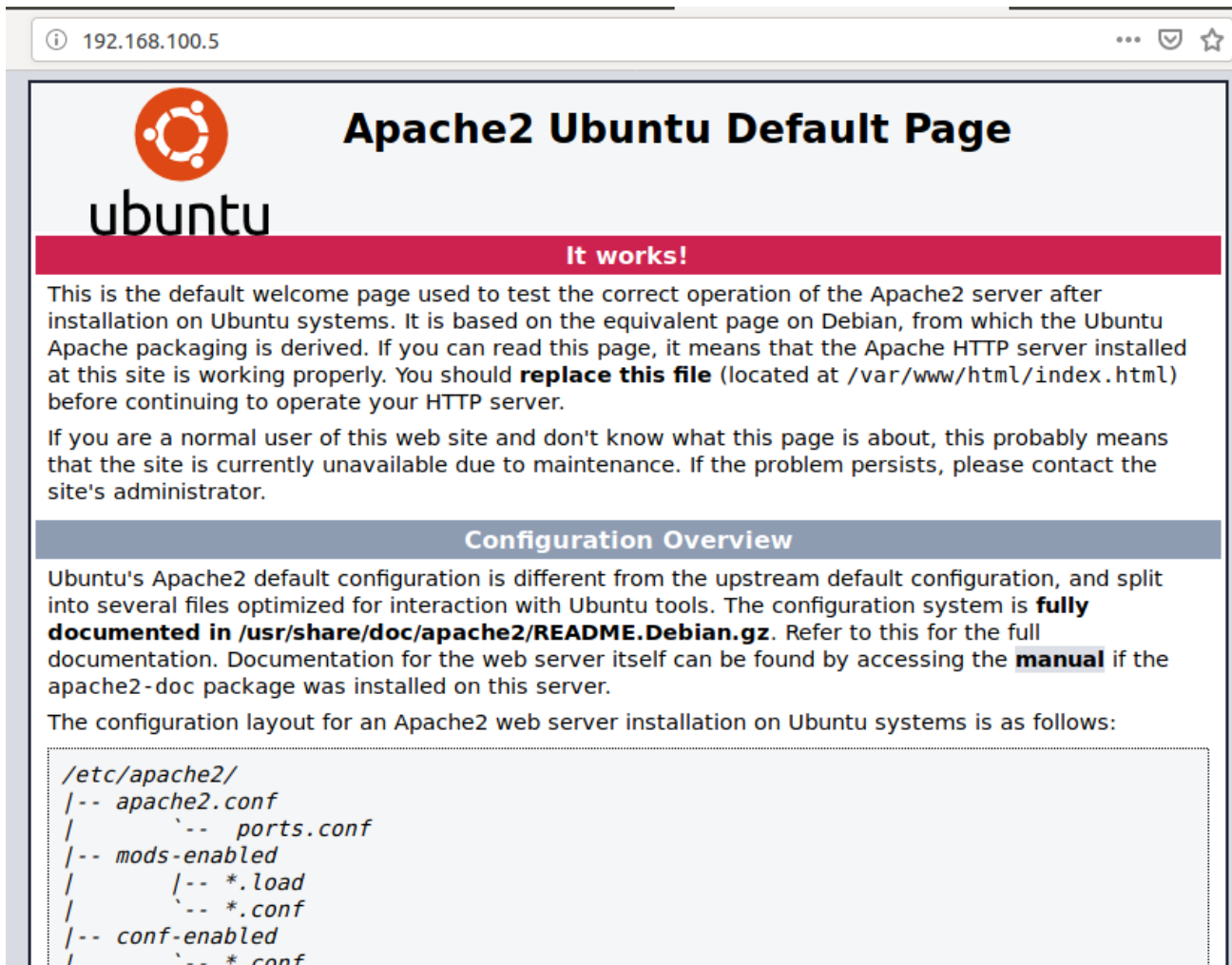
```
$ hostname -I
```

```
sana@Linux:~$ hostname -I  
192.168.100.5 192.168.100.4 172.17.0.1
```

Następnie wypróbuj adresy IP, jeden po drugim z wyjścia, w przeglądarce internetowej w następujący sposób:


***http: // server\_IP***

W moim przypadku , `http://192.168.100.4` i `http://192.168.100.5`. Spowoduje to wyświetlenie następującej strony internetowej Apache dla systemu Ubuntu, sprawdzając, czy serwer Apache działa poprawnie.



192.168.100.5

# Apache2 Ubuntu Default Page



ubuntu

**It works!**

This is the default welcome page used to test the correct operation of the Apache2 server after installation on Ubuntu systems. It is based on the equivalent page on Debian, from which the Ubuntu Apache packaging is derived. If you can read this page, it means that the Apache HTTP server installed at this site is working properly. You should **replace this file** (located at `/var/www/html/index.html`) before continuing to operate your HTTP server.

If you are a normal user of this web site and don't know what this page is about, this probably means that the site is currently unavailable due to maintenance. If the problem persists, please contact the site's administrator.

## Configuration Overview

Ubuntu's Apache2 default configuration is different from the upstream default configuration, and split into several files optimized for interaction with Ubuntu tools. The configuration system is **fully documented in `/usr/share/doc/apache2/README.Debian.gz`**. Refer to this for the full documentation. Documentation for the web server itself can be found by accessing the **manual** if the `apache2-doc` package was installed on this server.

The configuration layout for an Apache2 web server installation on Ubuntu systems is as follows:

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

## Konfigurowanie wirtualnych hostów w Apache

Wirtualny host jest podobny do bloków serwerów w Nginx. Służy do zarządzania konfiguracjami dla więcej niż jednej domeny z jednego serwera. Przedstawimy przykład konfiguracji wirtualnego hosta za pośrednictwem serwera Apache. Skonfigurujemy witrynę o nazwie `sampledomain.com`, używając bloku serwera, który jest domyślnie włączony w Apache dla Ubuntu 18.



## Krok 1: Skonfiguruj nazwę domeny

Blok serwera, który jest domyślnie włączony, może wyświetlać dokumenty z / var / www / html. Jednak utworzymy katalog pod adresem / var / www / pozostawiając nienaruszony katalog domyślny.

Utwórz ten katalog za pomocą następującego polecenia, zastępując sampledomain.com odpowiednią nazwą domeny.

```
sudo mkdir -p /var/www/sampledomain.com/html
```

```
sana@Linux:~$ sudo mkdir -p /var/www/sampledomain.com/html  
[sudo] password for sana:
```

Następnie przypisz własność katalogu za pomocą następujących poleceń:

```
sudo chown -R $USER:$USER /var/www/sampledomain.com/html
```

```
sudo chmod -R 755 /var/www/sampledomain.com
```

```
sana@Linux:~$ sudo chown -R $USER:$USER /var/www/sampledomain.com/html  
sana@Linux:~$ sudo chmod -R 755 /var/www/sampledomain.com
```

Utwórzmy teraz stronę indeksu, do której możemy później uzyskać dostęp w celu przetestowania, czy Apache obsługuje naszą nazwę domeny. Utwórz plik HTML za pomocą edytora Nano lub dowolnego ulubionego edytora tekstu.

```
$ nano /var/www/sampldomain.com/html/index.html
```

Wpisz następujący kod HTML strony indeksu:

```
<html>  
<head>  
<title>Welcome to the page sampldomain.com!</title>  
</head>  
<body>  
<h1>You got Lucky! Your sampldomain.com server block is up!</h1>  
</body>  
</html>
```

```
sana@Linux: ~
File Edit View Search Terminal Help
GNU nano 2.9.3 /var/www/sampldomain.com/html/index.html Modified
<html>
  <head>
    <title>Welcome to the page sampldomain.com!</title>
  </head>
  <body>
    <h1>You got Lucky! Your sampldomain.com server block is up!</h1>
  </body>
</html>
Save modified buffer? (Answering "No" will DISCARD changes.)
Y Yes
N No ^C Cancel
```

Używamy edytora nano do tworzenia pliku HTML.

Możesz zapisać plik w nano, używając Ctrl + X, a następnie wpisz Y i wciśnij Enter.

Apache potrzebuje wirtualnego pliku hosta do obsługi zawartości twojego serwera. Domyślny plik konfiguracyjny do tego celu jest już utworzony, ale utworzymy nowy dla naszych niestandardowych konfiguracji.

```
$ sudo nano /etc/apache2/sites-available/sampldomain.com.conf
```

Wprowadź następujące niestandardowe szczegóły konfiguracji dla naszej nazwy domeny:

```
<VirtualHost *:80>
ServerAdmin admin@sampledomain.com
ServerName sampledomain.com
ServerAlias www.sampledomain.com
DocumentRoot /var/www/sampledomain.com/html
ErrorLog ${APACHE_LOG_DIR}/error.log
CustomLog ${APACHE_LOG_DIR}/access.log combined
</VirtualHost>
```

```
sana@Linux: ~
File Edit View Search Terminal Help
GNU nano 2.9.3 /etc/apache2/sites-available/sampledomain.com.conf Modified

<VirtualHost *:80>
  ServerAdmin admin@sampledomain.com
  ServerName sampledomain.com
  ServerAlias www.sampledomain.com
  DocumentRoot /var/www/sampledomain.com/html
  ErrorLog ${APACHE_LOG_DIR}/error.log
  CustomLog ${APACHE_LOG_DIR}/access.log combined
</VirtualHost>

save modified buffer? (Answering "No" will DISCARD changes.)
Y Yes
N No      ^C Cancel
```

Używamy edytora nano do utworzenia tego pliku .conf.

Możesz zapisać plik w nano, używając **Ctrl + X**, a następnie wpisz Y i wciśnij Enter.

## Krok 2: Włącz plik konfiguracyjny domeny

Pozwól nam włączyć plik konfiguracyjny utworzony za pomocą narzędzia a2ensite:

```
$ sudo a2ensite sampledomain.com.conf
```

```
sana@Linux:~$ sudo a2ensite sampledomain.com.conf
Enabling site sampledomain.com.
To activate the new configuration, you need to run:
  systemctl reload apache2
sana@Linux:~$
```

Dane wyjściowe zasugerują aktywację nowej konfiguracji, ale możemy to wszystko zrobić zbiorowo po uruchomieniu następujących czynności polecenie, które wyłącza oryginalny plik konfiguracyjny:

```
$ sudo a2dissite 000-default.conf
```

```
sana@Linux:~$ sudo a2dissite 000-default.conf
Site 000-default disabled.
To activate the new configuration, you need to run:
  systemctl reload apache2
```

Teraz uruchom ponownie usługę Apache:

```
$ sudo systemctl restart apache2
```

### Krok 3: Test na błędy

Na koniec przetestujemy, czy występują jakieś błędy konfiguracji za pomocą następującego polecenia:

```
$ sudo apache2ctl configtest
```

Jeśli nie otrzymasz żadnych błędów, otrzymasz następujące dane wyjściowe:

```
sana@Linux:~$ sudo apache2ctl configtest  
Syntax OK
```

Jednak następujący błąd występuje często w Ubuntu 18.04

```
sana@Linux:~$ sudo apache2ctl configtest  
AH00558: apache2: Could not reliably determine the server's fully qualified domain  
name, using 192.168.100.5. Set the 'ServerName' directive globally to suppress  
this message  
Syntax OK  
sana@Linux:~$
```

### Rozwiąż błąd:

Wpisz następujące polecenie, aby rozwiązać wyżej wymieniony błąd:

```
$ echo "ServerName sampledomain.com" | sudo tee /etc/apache2/conf-available/se
```

```
sana@Linux:~$ echo "ServerName sampledomain.com" | sudo tee /etc/apache2/conf-av  
ailable/servername.conf  
ServerName sampledomain.com
```

A potem:

```
$ sudo a2enconf servername
```

```
sana@Linux:~$ sudo a2enconf servername
Enabling conf servername.
To activate the new configuration, you need to run:
systemctl reload apache2
```

Teraz, gdy ponownie sprawdzisz, czy nie ma błędów, zobaczysz to błąd rozwiązany przez następujące dane wyjściowe:

```
sana@Linux:~$ sudo apache2ctl configtest
Syntax OK
```

#### Krok 4: Sprawdź, czy Apache obsługuje twoją nazwę domeny

Serwer Apache jest teraz skonfigurowany do obsługi twojej nazwy domeny. Można to zweryfikować, wprowadzając nazwę serwera w następujący sposób w dowolnej przeglądarce internetowej uruchomionej w systemie:

***<http://sampledomain.com>***

Strona indeksu powinna być wyświetlana w następujący sposób, wskazując, że Apache jest teraz gotowy do obsługi twojego blok serwera!





## Niektóre typowe polecenia zarządzania Apache

Po skonfigurowaniu serwera WWW może być konieczne wykonanie podstawowych operacji zarządzania na Apache. Oto polecenia, które można wprowadzić w aplikacji Terminal dla tych operacji.

```
sudo systemctl start apache2
```

Użyj tego polecenia jako sudo, aby uruchomić serwer Apache.

```
sudo systemctl stop apache2
```

Użyj tego polecenia jako sudo, aby zatrzymać serwer Apache, gdy jest on w trybie startowym.

```
sudo systemctl restart apache2
```

Użyj tego polecenia jako sudo, aby zatrzymać, a następnie ponownie uruchomić usługę Apache.

```
sudo systemctl reload apache2
```

Użyj tego polecenia jako sudo, aby zastosować zmiany konfiguracji bez ponownego uruchamiania połączenia.

```
sudo systemctl enable apache2
```

Użyj tego polecenia jako sudo, aby umożliwić uruchamianie Apache przy każdym uruchomieniu systemu.

```
sudo systemctl disable apache2
```

Użyj tego polecenia jako sudo, aby wyłączyć, jeśli skonfigurowałeś uruchamianie Apache przy każdym uruchomieniu systemu.

## Wniosek

W tym artykule nauczyłeś się instalować i konfigurować serwer WWW Apache w systemie Ubuntu. Obejmuje to wprowadzenie zmian w zaporze UFW, a następnie skonfigurowanie serwera WWW pod kątem adresu IP. Zalecamy również skonfigurowanie wirtualnego hosta za pośrednictwem Apache; da ci to podstawę do używania Apache do hostowania twoich plików w Internecie. Podstawowe polecenia zarządzania Apache pomogą również jako administrator sieci w optymalnym zarządzaniu serwerem WWW.

## KOMENTARZE

---



**klesk**

```
root@klesk:~# $ echo "ServerName historiewikingow.pl | sudo tee  
/etc/apache2/conf-available/servername.conf"
```

\$: nie znaleziono polecenia



**nie wiem**

dolara przed echo usuń i dodaj cudzysłów po nazwie serwera

Nazwa

Komentarz

Opublikuj komentarz