

## Ćwiczenia 22 – instalacja i konfiguracja serwera Apache



1. Zaloguj się na swoje konto.
2. Sprawdź czy zainstalowany jest pakiet openssl

```
andrzej@servubu:~/apache/download$ whereis openssl
openssl: /usr/bin/openssl /usr/include/openssl /usr/
```

3. Utwórz katalog /home/twoje\_konto/**apache**/, a w nim podkatalog **download**
4. Źródła można pobrać ze strony <https://dldn.apache.org/httpd/httpd-2.4.63.tar.bz2> narzędziem wget.
5. Sumę kontrolną gpg pobieramy ze strony <https://downloads.apache.org/httpd/httpd-2.4.63.tar.bz2.asc> narzędziem wget.

6. Sumę kontrolną pobieramy: <https://downloads.apache.org/httpd/httpd-2.4.63.tar.bz2.sha256> narzędziem curl.

```
mkdir apache/download -p
cd apache/download
wget https://dldn.apache.org/httpd/httpd-2.4.54.tar.bz2
wget https://downloads.apache.org/httpd/httpd-2.4.54.tar.bz2.asc
curl -O https://downloads.apache.org/httpd/httpd-2.4.54.tar.bz2.sha256
```

7. Sprawdzić sumę kontrolną narzędziem sha256sum httpd-2.4.63.tar.bz2 i cat httpd-2.4.63.tar.bz2.sha256 lub shasum -a 256 -c httpd-2.4.63.tar.bz2.sha256

```
andrzej@servubu:~/apache/download$ ls
apr-1.7.0.tar.gz      httpd-2.4.54      httpd-2.4.54.tar.bz2.asc  pcre-8.45
apr-util-1.6.1.tar.gz httpd-2.4.54.tar.bz2 httpd-2.4.54.tar.bz2.sha256 pcre-8.45.tar.bz2
andrzej@servubu:~/apache/download$ shasum -a 256 -c httpd-2.4.54.tar.bz2.sha256
httpd-2.4.54.tar.bz2: OK
```

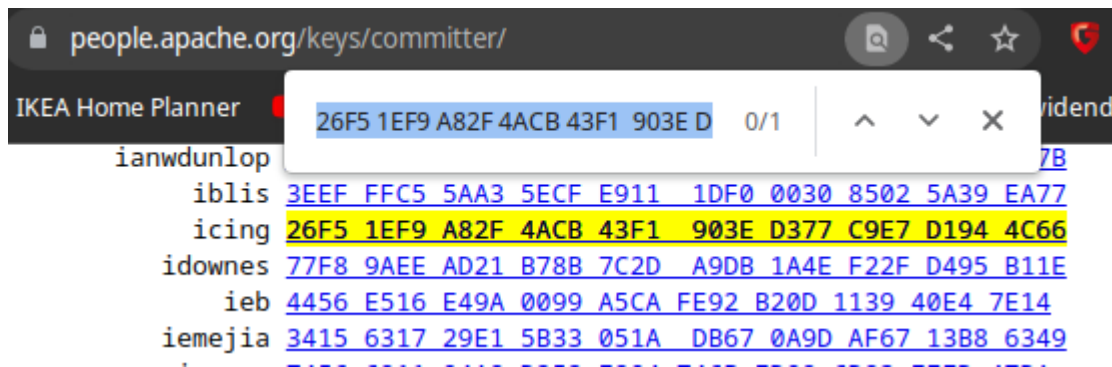
8. Sprawdzić numer klucza (pierwsza poniższa komenda go wyświetli ) i go ściągnąć:

```
gpg --verify httpd-2.4.54.tar.bz2.asc httpd-2.4.54.tar.bz2
gpg --keyserver pgpkeys.mit.edu --recv-key 26F51EF9A82F4ACB43F1903ED377C9E7D1944C66
```

9. gpg --verify httpd-2.4.63.tar.bz2.asc httpd-2.4.63.tar.bz2

```
andrzej@servubu:~/apache/download$ gpg --verify httpd-2.4.54.tar.bz2.asc httpd-2.4.54.tar.bz2
gpg: Signature made Mon 06 Jun 2022 04:23:05 PM CEST
gpg: using RSA key 26F51EF9A82F4ACB43F1903ED377C9E7D1944C66
gpg: Good signature from "Stefan Eissing (icing) <stefan@eissing.org>" [unknown]
gpg: WARNING: This key is not certified with a trusted signature!
gpg: There is no indication that the signature belongs to the owner.
Primary key fingerprint: 26F5 1EF9 A82F 4ACB 43F1 903E D377 C9E7 D194 4C66
```

10. Sprawdzamy na stronie <https://people.apache.org/keys/commmitter/> odcisk palca.



11. Rozpakować plik **httpd-2.4.63.tar.bz2** komendą tar.

12. następnie przejść do katalogu httpd-2.4.63

13. Wydać komendę:

**./configure --prefix=/home/twoje\_konto/apache --with-ssl=/usr/bin/openssl --enable-ssl --enable-so**

14. W przypadku braku APR zainstaluj potrzebne pakiety komendą:

**sudo apt-get install libapr1-dev libaprutil1-dev -y**

15. W przypadku braku kompilatora c: build-essential lub gcc i g++

16. W przypadku braku pcre3: zainstaluj pakiet pcre: **sudo apt install libpcre3-dev libpcre3 -y**

17. libssl-dev - > openssl is too old

18. Wydać komendę: **make -j\$(nproc)**

19. Wydać komendę: **make install**

```
tar xvfj httpd-2.4.54.tar.bz2
cd httpd-2.4.54/
./configure --prefix=/home/andrzej/apache --with-ssl=/usr/bin/openssl --enable-ssl --enable-so
sudo apt-get install libapr1-dev libaprutil1-dev
./configure --prefix=/home/andrzej/apache --with-ssl=/usr/bin/openssl --enable-ssl --enable-so
make
sudo make install
```

20. Sprawdź czy jest APR: **apache/bin/httpd -V**

21. Przejdź do katalogu /home/twoje\_konto/apache i sprawdź czy powstały katalogi

```
andrzej@servubu:~/apache$ ls
bin build cgi-bin conf _download error htdocs icons include logs man manual modules
```

22. Uruchomić serwer komendą: **sudo bin/apachectl restart**

23. Jeśli wystąpią błędy odkomentuj i zmień wpis

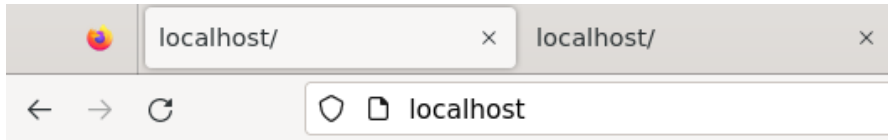
```
/home/andrzej/apache/conf/httpd.conf
# If your host doesn't have a registered domain name, use the following line.
#
ServerName www.example.net:80
```

24. Sprawdź czy istnieje proces dla serwera

komendą: **ps aux | grep apache**

```
andrzej@servubu:~/apache$ sudo ps aux | grep apache
root      130722  0.0  0.2 10508 8452 ?        Ss   19:42   0:00 /home/andrzej/apache/bin/httpd -k start
daemon    132054  0.0  0.2 1215588 7100 ?        Sl   20:18   0:00 /home/andrzej/apache/bin/httpd -k start
```

25. Uruchomić przeglądarkę i sprawdzić działanie wpisując: **localhost lub ip serwera**



**It works! from /home/andrzej/**

26. Sprawdź konfigurację poleceniem: `apachectl configtest`.

27. W chrome w górnym pasku DevTools kliknij i wybierz **Lighthouse**

28. Sprawdź działanie strony narzędziem curl:

```
andrzej@servubu:~/apache$ sudo curl -v localhost
* Trying 127.0.0.1:80...
* TCP_NODELAY set
* Connected to localhost (127.0.0.1) port 80 (#0)
> GET / HTTP/1.1
> Host: localhost
> User-Agent: curl/7.68.0
> Accept: */*
>
* Mark bundle as not supporting multiuse
< HTTP/1.1 200 OK
< Date: Fri, 09 Dec 2022 20:34:40 GMT
< Server: Apache/2.4.54 (Unix) OpenSSL/1.1.1f
< Last-Modified: Fri, 09 Dec 2022 19:21:59 GMT
```

29. Sprawdź logi: `cat ~/apache/logs/error_log` i `cat ~/apache/logs/access_log`

```
andrzej@servubu:~/apache$ cat logs/access_log
127.0.0.1 - - [09/Dec/2022:19:44:06 +0100] "GET / HTTP/1.1" 200 45
127.0.0.1 - - [09/Dec/2022:19:45:12 +0100] "GET / HTTP/1.1" 200 45
127.0.0.1 - - [09/Dec/2022:19:45:13 +0100] "GET /favicon.ico HTTP/1.1" 404 196
```

30. Jeśli nie działa serwer to sprawdź uprawnienia na katalogu `/home/twoje_konto` powinno być 751

31. Analogicznie przetestuj serwer linkowy sąsiada.(wpisz ip sąsiada w przeglądarce lub narzędziem `curl -v ip`)

32. Jeśli nie możesz skorzystać z serwera sąsiada to należy otworzyć porty **80 i 443**.

33. Sprawdź połączenie z pomocą wireshark.

34. Sprawdź zawartość logów.

35. Popraw wygląd swojej strony. ( plik: `~/apache/htdocs/index.html` )

36. Wygeneruj certyfikat dla serwera z pomocą openssl ( patrz wykład).

37. Klucz prywatny serwera:

```
andrzej@servubu:~/apache/conf$ sudo openssl genrsa --aes256 -out server.key 2048
Enter PEM pass phrase:
Verifying - Enter PEM pass phrase:
```

38. Certyfikat:

```
andrzej@servubu:~/apache/conf$ sudo openssl req -new -x509 -nodes -sha1 -days 365 -key server.key -out server.crt -extensions usr_cert
Enter pass phrase for server.key:
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:PL
```

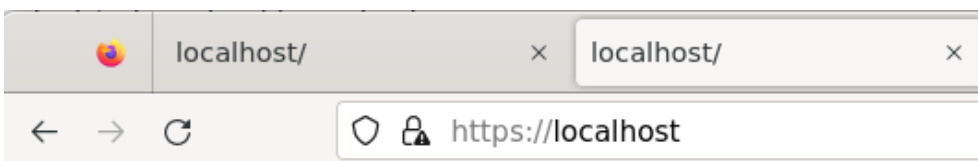
39. Zdjęcie hasła:

```
andrzej@servubu:~/apache/conf$ sudo openssl rsa -in server.key -out server.key
Enter pass phrase for server.key:
writing RSA key
```

40. Odkomentuj linię: *Include conf/extra/httpd-ssl.conf* w pliku *~/apache/conf/httpd.conf*

41. Zrestartuj serwer : `sudo bin/apachectl restart`

42. Uruchomić przeglądarkę i sprawdzić działanie wpisując: `https://localhost`



## Dziwne u mnie działa !!!

43. Sprawdź działanie strony narzędziem curl:

```

andrzej@servubu:~/apache$ sudo curl -v https://localhost
* Trying 127.0.0.1:443...
* TCP_NODELAY set
* Connected to localhost (127.0.0.1) port 443 (#0)
* ALPN, offering h2
* ALPN, offering http/1.1
* successfully set certificate verify locations:
*   CAfile: /etc/ssl/certs/ca-certificates.crt
*   Cpath: /etc/ssl/certs
* TLSv1.3 (OUT), TLS handshake, Client hello (1):
* TLSv1.3 (IN), TLS handshake, Server hello (2):
* TLSv1.3 (IN), TLS handshake, Encrypted Extensions (8):
* TLSv1.3 (IN), TLS handshake, Certificate (11):
* TLSv1.3 (OUT), TLS alert, unknown CA (560):
* SSL certificate problem: self signed certificate
* Closing connection 0

```

44. Powinien powstać plik ~/apache/logs/ssl\_request\_log

```

andrzej@servubu:~/apache$ cat logs/ssl_request_log
[09/Dec/2022:20:19:08 +0100] 127.0.0.1 TLSv1.3 TLS_AES_256_GCM_SHA384 "GET / HTTP/1.1" 78
[09/Dec/2022:20:19:08 +0100] 127.0.0.1 TLSv1.3 TLS_AES_256_GCM_SHA384 "GET /favicon.ico HTTP/1.1" 196
[09/Dec/2022:20:22:05 +0100] 127.0.0.1 TLSv1.3 TLS_AES_256_GCM_SHA384 "GET / HTTP/1.1" 60

```

45. Sprawdź aktywne połączenia ze swoim serwerem komendą: netstat lub ss -anp | grep 443

```

andrzej@servubu:~/apache$ ss -anp | grep 443
u_str  ESTAB      0          0                                     * 34080
                                     * 33443
u_str  ESTAB      0          0                                     /run/user/1001/bus 33443
                                     * 34080
                                     users: ( ("dbus-daemon",pid=1383,fd=12) )

```

46. Sprawdź aktywne połączenia ze swoim serwerem

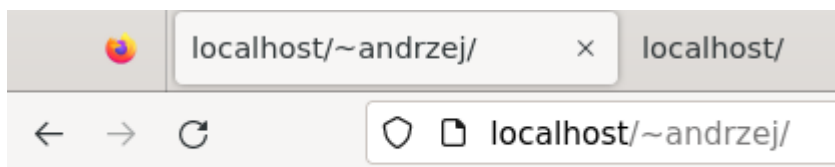
```

andrzej@servubu:~/apache$ openssl s_client -connect localhost:443
CONNECTED(00000003)

```

47. Dodać możliwość tworzenia stron www przez użytkowników systemowych: np.

https://localhost/~twoje\_konto (wskazówka: public\_html)



moja prwatna strona

wskazówka 2:

```
LoadModule userdir_module modules/mod_userdir.so
```

wskazówka 3:

```
andrzej@servubu:~/public_html$ pwd
/home/andrzej/public_html
andrzej@servubu:~/public_html$ ls
index.html
andrzej@servubu:~/public_html$ cat index.html
moja prwatna stronaandrzej@servubu:~/public_ht
```

#### 48. Dodatkowe:

- a) Utwórz własne usługi dla apache: apache-twoje-imię
  - Utwórz nowy plik serwisowy: `sudo nano /etc/systemd/system/apache_imie.service`
  - Zawartość pliku `apache_imie.service`:

```
[Unit]
Description=Apache HTTP Server (apache_imie)
After=network.target

[Service]
Type=forking
ExecStart=/opt/apache/bin/apachectl -k start
ExecReload=/opt/apache/bin/apachectl -k graceful
ExecStop=/opt/apache/bin/apachectl -k stop
PIDFile=/opt/apache/logs/httpd.pid
User=www-data
Group=www-data
Restart=always

[Install]
WantedBy=multi-user.target
```

- nadaj uprawnienia i odśwież systemd:

```
sudo chmod 644 /etc/systemd/system/apache_imie.service
sudo systemctl daemon-reexec
sudo systemctl daemon-reload
```

- Uruchom nową usługę: `sudo systemctl start apache_imie`

- b) dsd

#### 49. KONIEC