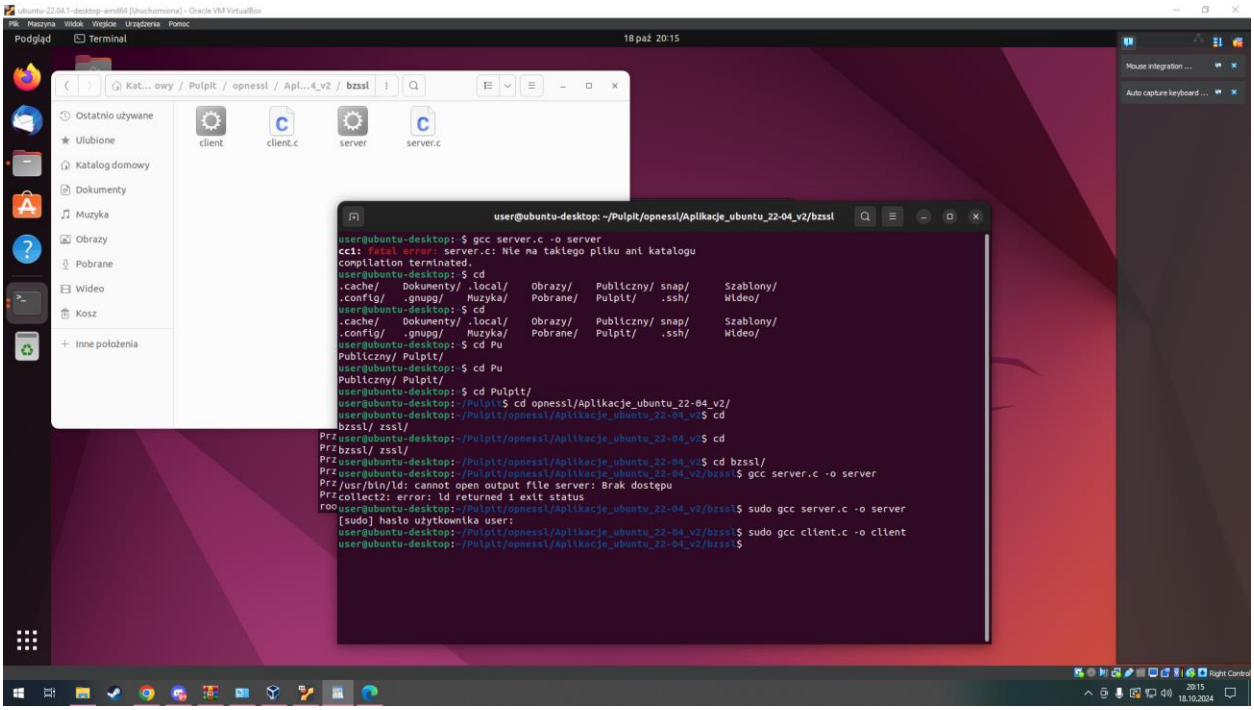
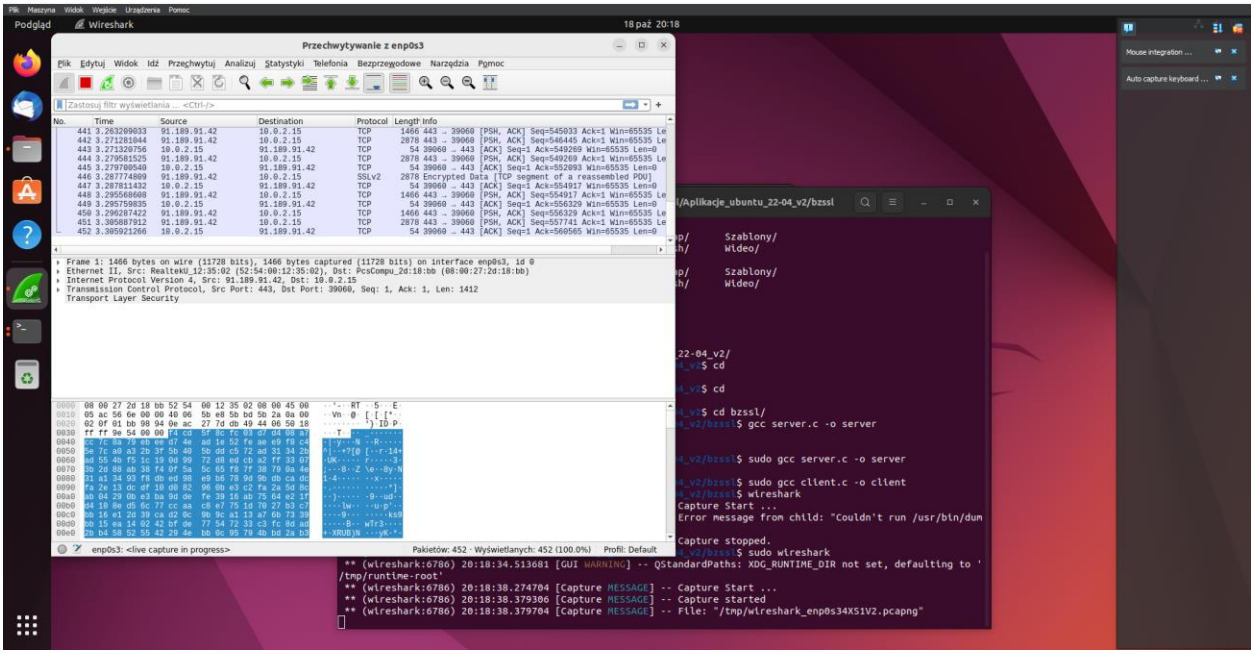
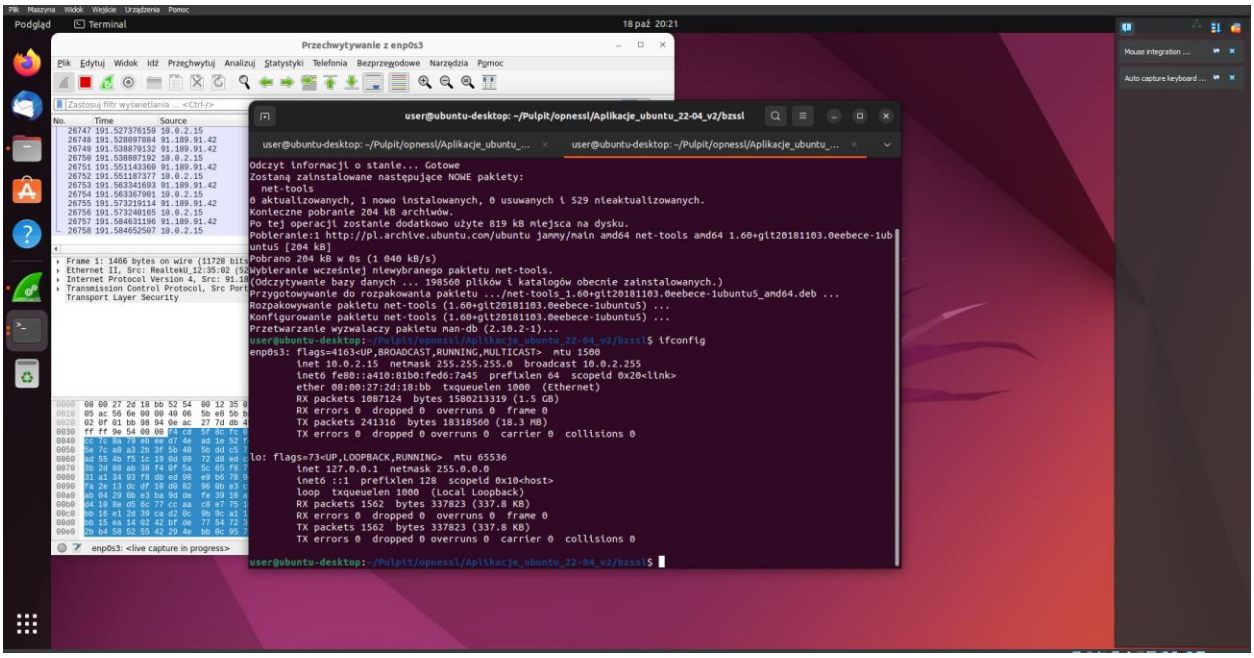
1. Skompiluj pliki client.c i server.c używając następujących komend: ~/Pulpit/openssl/Aplikacje/bzssl $gcc server.c –o server ~/Pulpit/openssl/Aplikacje/bzssl $gcc client.c –o client (jeżeli pojawią się ostrzeżenia, należy je zignorować)



2. Uruchom program Wireshark z prawami administratora i rozpocznij podsłuchiwanie swojej karty sieciowej. Z menu Capture wybierz opcję “start”, a następnie wybierz interfejs sieciowy i kliknij „ok.”. W tym momencie program Wireshark odczytuje wszystkie informacje przechodzące przez kartę sieciową Twojego komputera.

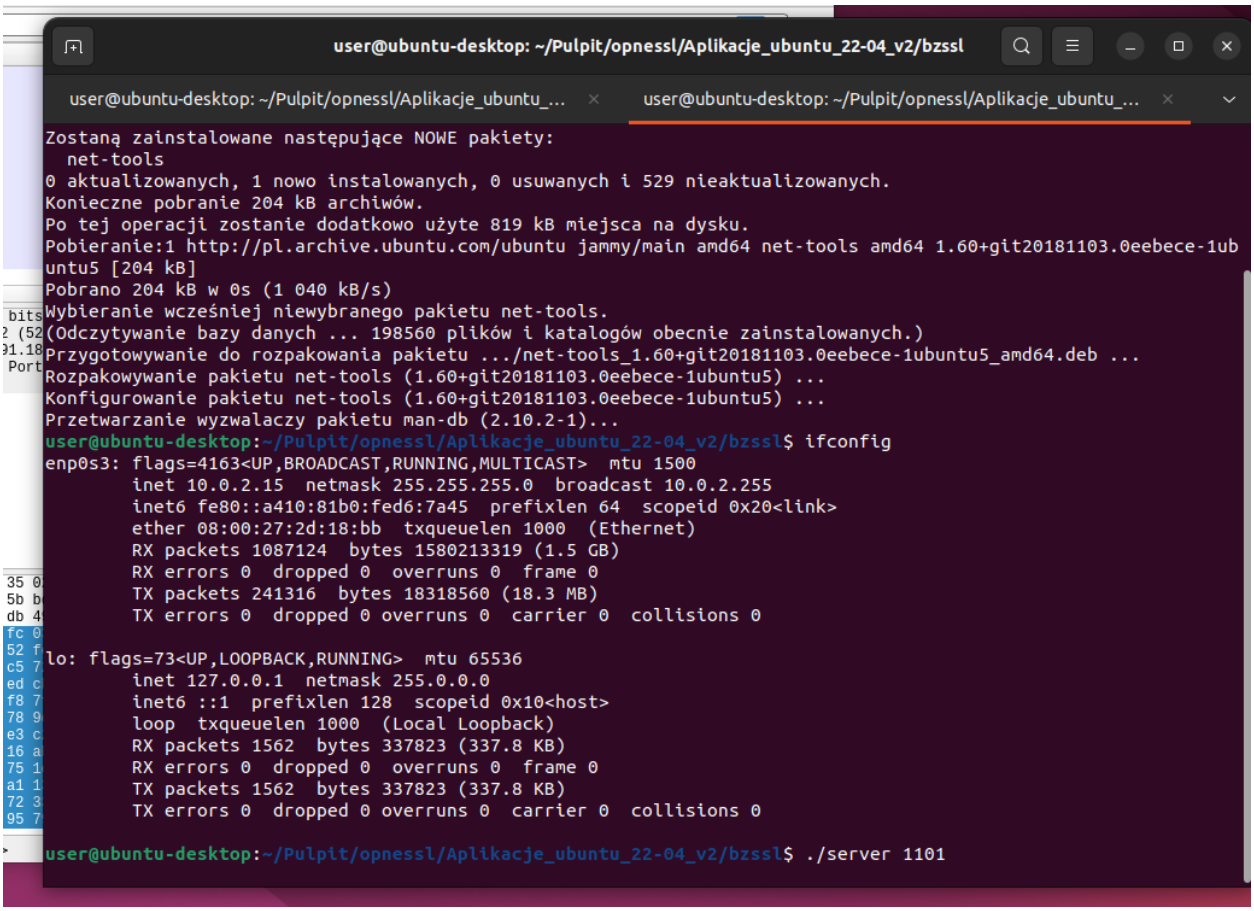


3. Odczytaj i zapisz adres IP swojego komputera wpisując polecenie ifconfig (jeżeli brakuje polecenia to zainstaluj: sudo apt install net-tools) Aby móc uruchomić program, należy zmienić tryb dostępu dla pliku – dodać tryb wykonywania chmod +x server Czynność tą należy wykonać tylko w przypadku, gdy plik wykonywalny nie posiada odpowiednio ustawionych praw wykonania. UWAGA! Aby uruchomić program w systemie Linux należy przed jego nazwą dodać znaki ./, np. ./server



4. Dwa pierwsze pliki client.c i server.c są odpowiednio oprogramowaniem klienta umożliwiającego połączenie się z podanym adresem IP i portem, na którym nasłuchuje serwer. Po nawiązaniu połączenia można odczytać wiadomość, która jest przesyłana tekstem otwartym do serwera.

5. Uruchom program server: ./server 1101 Program server nasłuchuje na porcie określonym w argumencie wykonania.

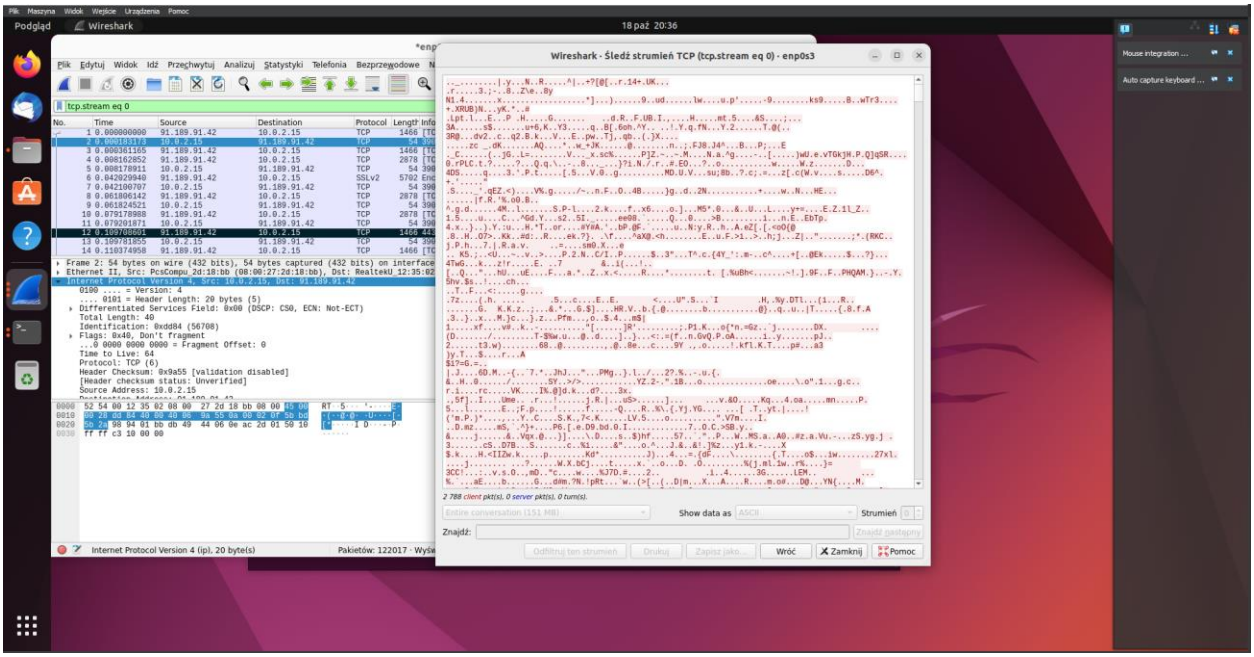


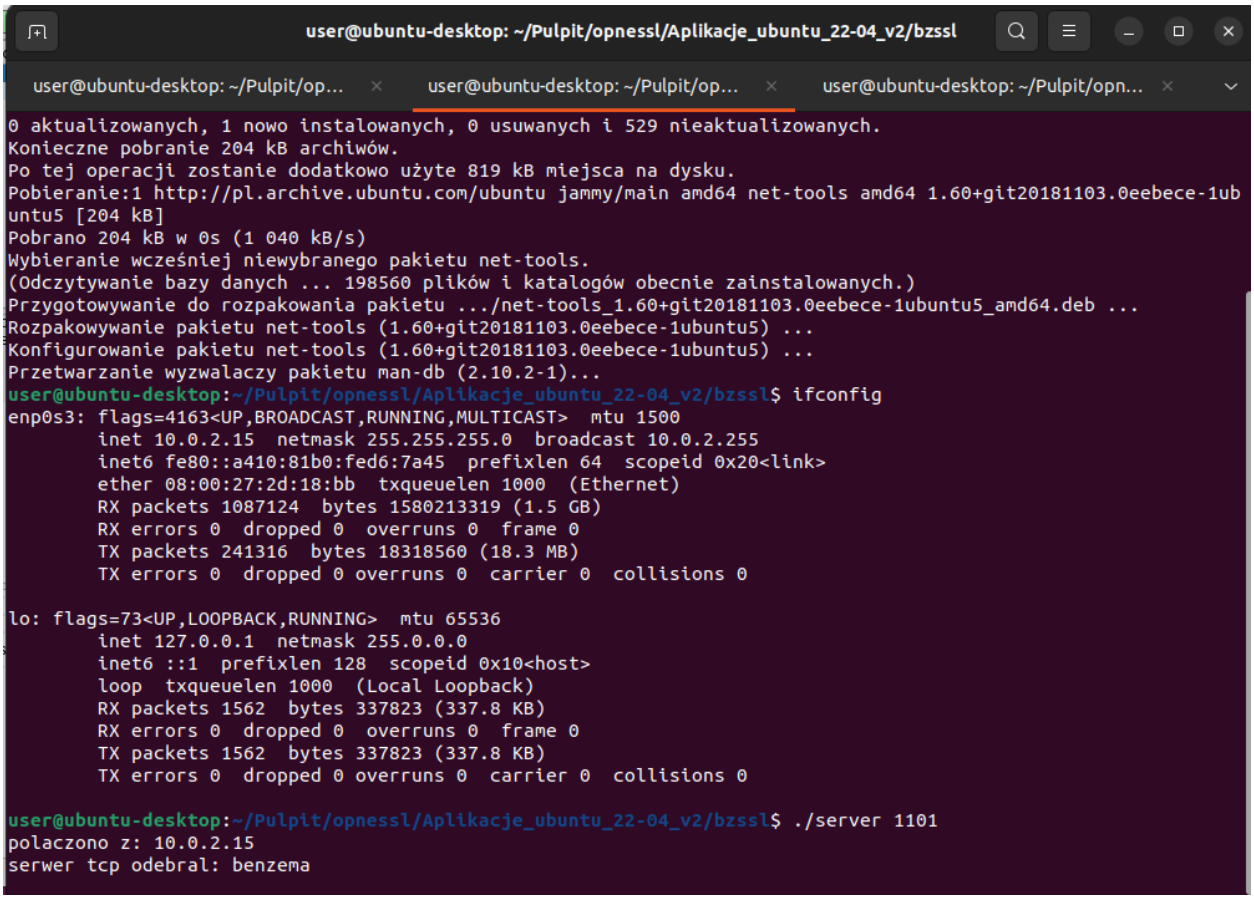
6. Utwórz nową konsolę. Aby móc uruchomić program klienta, należy zmienić tryb dostępu dla pliku – dodać tryb wykonywania: chmod +x client

7. Uruchom program klienta i wyślij wiadomość na serwer: ./client 127.0.0.1 1101 8. W programie zakończ przechwytywanie. Spróbuj odczytać jakie informacje zostały przesłane pomiędzy klientem a serwerem. Czy jest to możliwe? Możesz kliknąć prawym przyciskiem myszy na jednym z pakietów i wybrać opcję Podążaj->Strumień TCP to Wireshark połączy i wyświetli dane przesyłane w ramach połączenia.

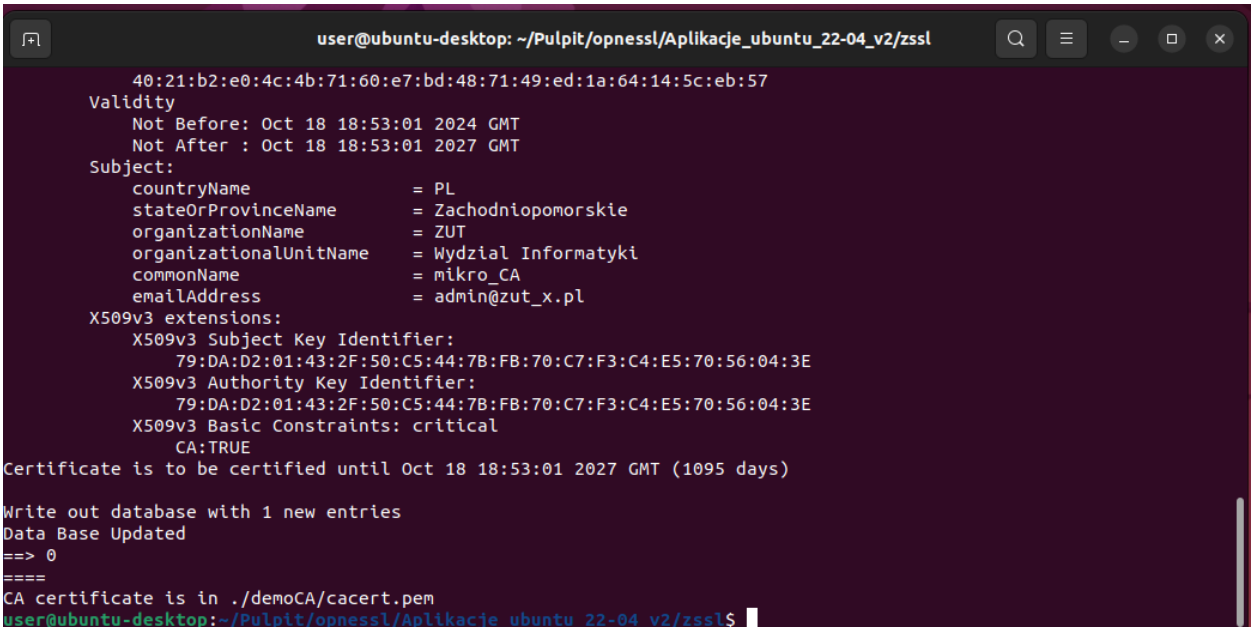
```
user@ubuntu-desktop: ~/Pulpit/opnessl/Aplikacje_ubuntu_22-04_v2/bzssl
user@ubuntu-desktop: ~/Pulpit/op...   user@ubuntu-desktop: ~/Pulpit/op...   user@ubuntu-desktop: ~/Pulpit/opn...

user@ubuntu-desktop:~/Pulpit/opnessl/Aplikacje_ubuntu_22-04_v2/bzssl$ chmod +x client
chmod: nie można zmienić uprawnień do 'client': Operacja niedozwolona
user@ubuntu-desktop:~/Pulpit/opnessl/Aplikacje_ubuntu_22-04_v2/bzssl$ sudo chmod +x client
[sudo] hasło użytkownika user:
user@ubuntu-desktop:~/Pulpit/opnessl/Aplikacje_ubuntu_22-04_v2/bzssl$ ./client 10.0.2.15 1101
Polaczenie nawiazane
Podaj tekst do wyslania: benzema
Wyslano 7 bajtow
user@ubuntu-desktop:~/Pulpit/opnessl/Aplikacje_ubuntu_22-04_v2/bzssl$
```



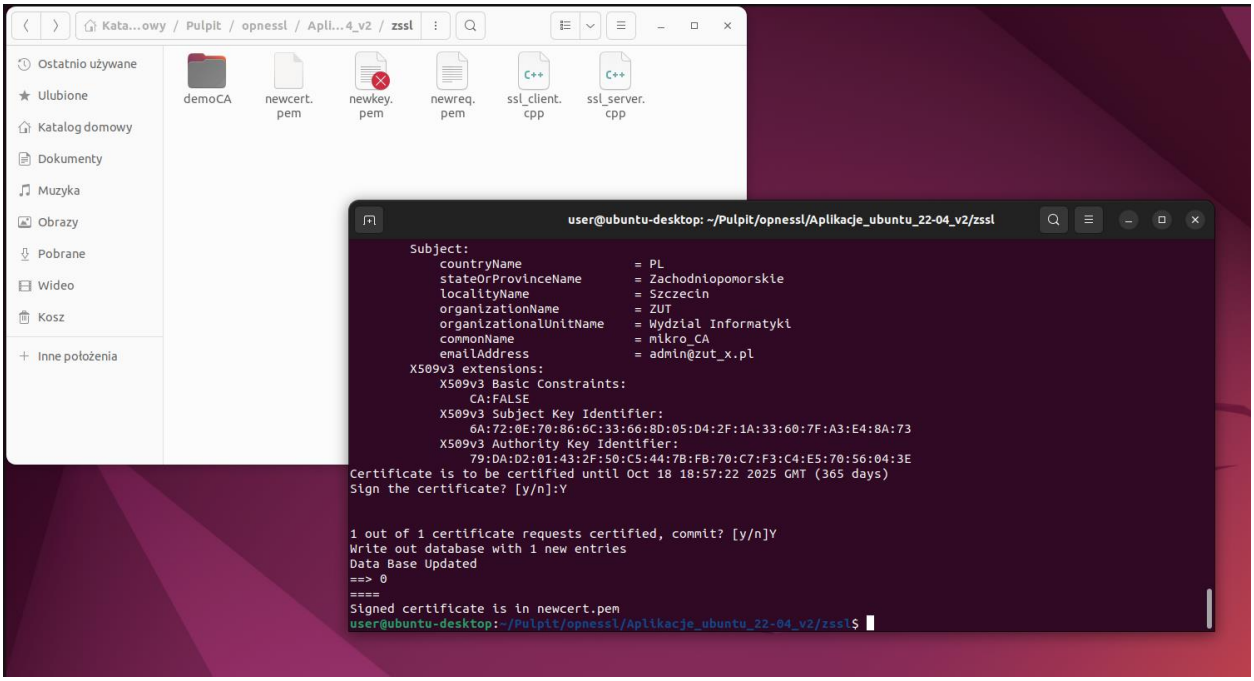


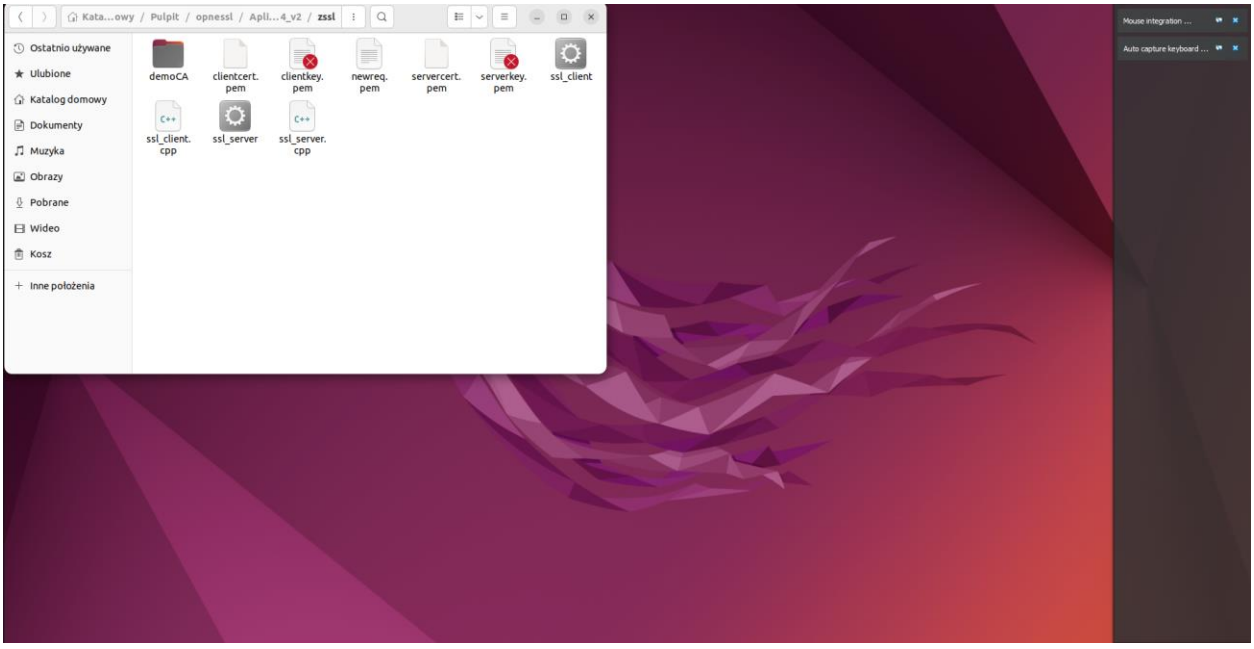
1. Utwórz własny mikro urząd certyfikacji przy pomocy biblioteki OpenSSL (patrz rozdz. 4.1).
Desktop/openssl/Aplikacje/zssl#/usr/lib/ssl/misc/CA.pl –newca



2. Wygeneruj żądania wystawienia certyfikatu dla serwera (program ssl_server), patrz rozdz. 4.2. Desktop/openssl/Aplikacje/zssl#/usr/lib/ssl/misc/CA.pl –newreq 3. Wystaw certyfikat dla serwera (patrz rozdz. 4.3).

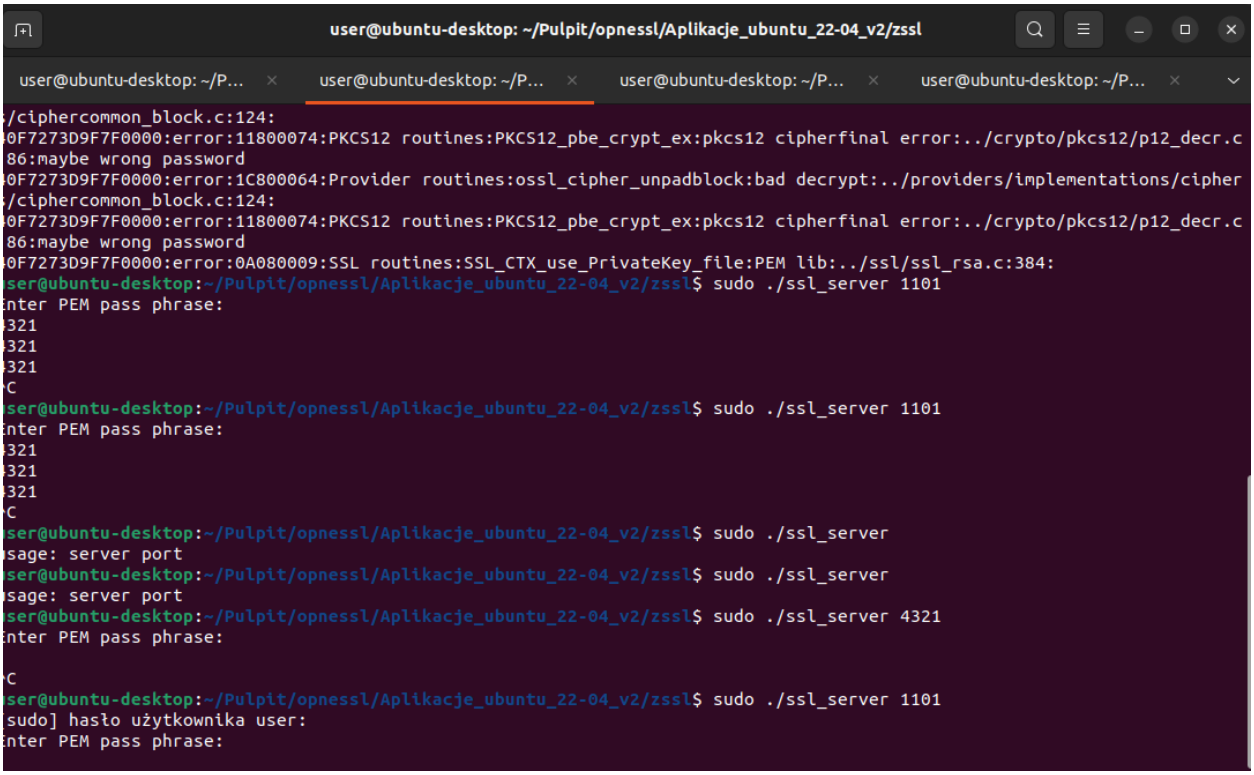
Desktop/openssl/Aplikacje/zssl#/usr/lib/ssl/misc/CA.pl –sign



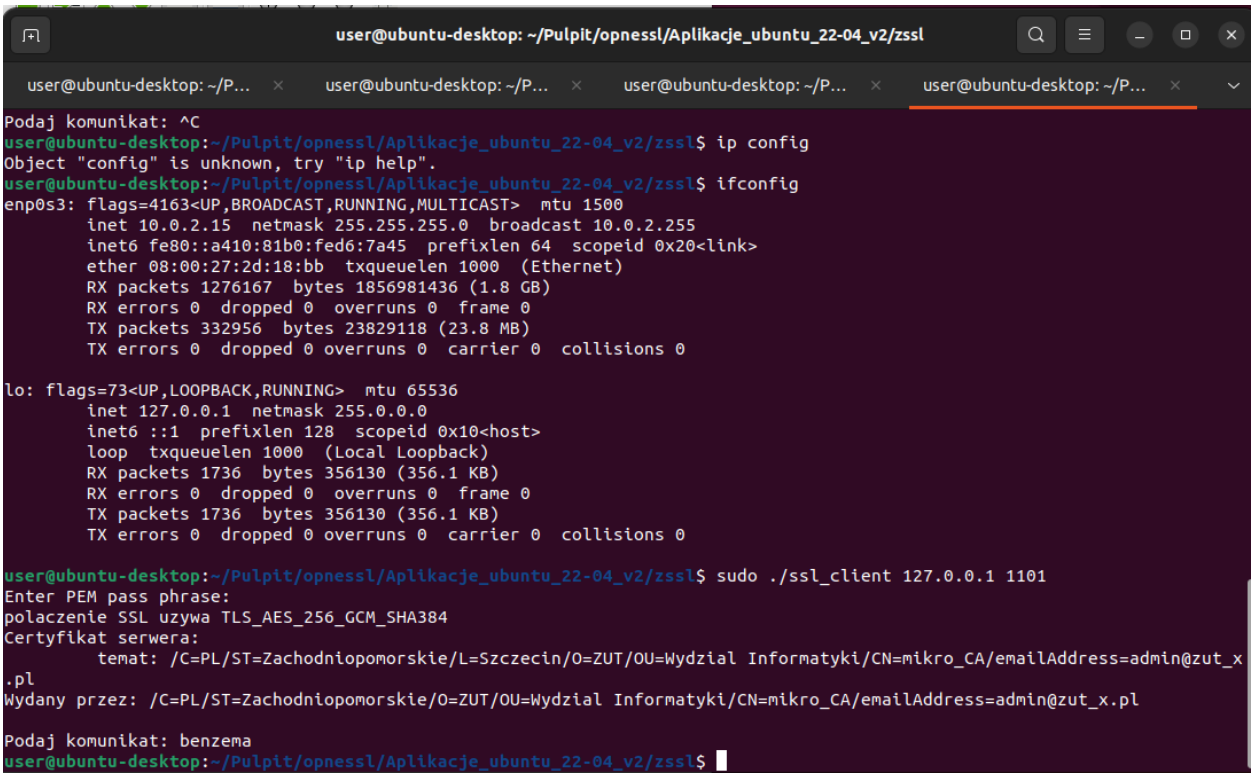


4. Zamień nazwy newcert.pem na servercert.pem oraz newkey.pem na serverkey.pem 5. Kroki 2 i 3 powtórz także dla oprogramowania ssl_klient. Dwa pliki newkey.pem i newcert.pem zamień na clientkey.pem i clientcert.pem

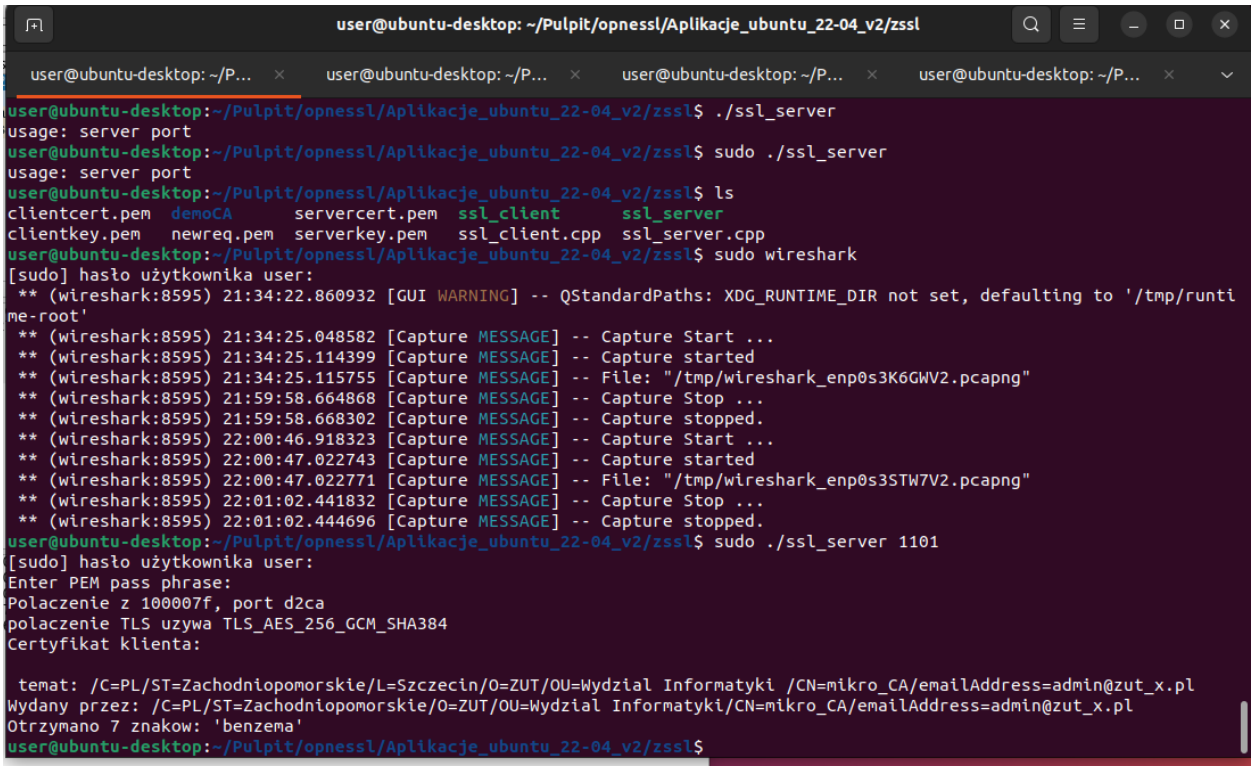
1.Odpalam serwer na porcie 1101

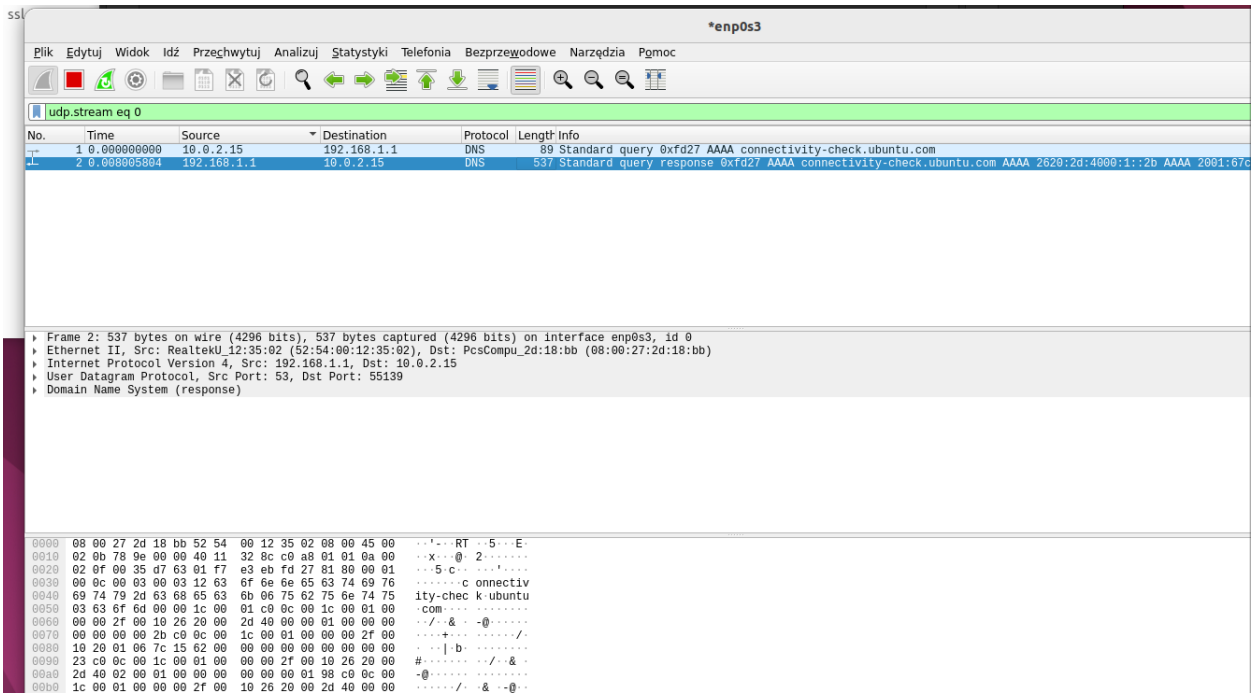


2.Podajemy komunikat do przesłania



3. Otrzymujemy wysłany komunikat.




*enp0s3
Plik Edytuj Widok Idź Przechwytuj Analizuj Statystyki Telefonia Bezprzewodowe Narzędzia Pomoc
udp.stream eq 0
No. Time Source Destination Protocol Length Info
1 0.000000000 10.0.2.15 192.168.1.1 DNS 89 Standard query 0xfd27 AAAA connectivity-check.ubuntu.com
2 0.008005804 192.168.1.1 10.0.2.15 DNS 537 Standard query response 0xfd27 AAAA connectivity-check.ubuntu.com AAAA 2620:2d:4000:1::2b AAAA 2001:67c
Frame 2: 537 bytes on wire (4296 bits), 537 bytes captured (4296 bits) on interface enp0s3, id 0
Ethernet II, Src: RealtekU_12:35:02 (52:54:00:12:35:02), Dst: PcsCompu_2d:18:bb (08:00:27:2d:18:bb)
Internet Protocol Version 4, Src: 192.168.1.1, Dst: 10.0.2.15
User Datagram Protocol, Src Port: 53, Dst Port: 55139
Domain Name System (response)
0000 08 00 27 2d 18 bb 52 54 00 12 35 02 08 00 45 00
0010 02 0b 78 9e 00 00 40 11 32 8c c0 a8 01 01 0a 00
0020 02 0f 00 35 d7 63 01 f7 e3 eb fd 27 81 80 00 01
0030 00 0c 00 03 00 03 12 63 6f 6e 6e 65 63 74 69 76
0040 69 74 79 2d 63 68 65 63 6b 06 75 62 75 6e 74 75
0050 03 63 6f 6d 00 00 1c 00 01 c0 0c 00 1c 00 01 00
0060 00 00 2f 00 10 26 20 00 2d 40 00 00 01 00 00 00
0070 00 00 00 00 2b c0 0c 00 1c 00 01 00 00 00 2f 00
0080 10 20 01 06 7c 15 62 00 00 00 00 00 00 00 00 00
0090 23 c0 0c 00 1c 00 01 00 00 00 2f 00 10 26 20 00
00a0 2d 40 02 00 01 00 00 00 00 00 00 01 98 c0 0c 00
00b0 1c 00 01 00 00 00 2f 00 10 26 20 00 2d 40 00 00