

Michał Matak

nr indeksu: 304071

Paweł Müller

nr indeksu: 304080

Jakub Robaczewski

nr indeksu: 304119

Grzegorz Rusinek

nr indeksu: 304083

Programowanie Sieciowe

Laboratorium nr 1

Zestaw programów klient – serwer
wysyłające datagramy UDP



Część pierwsza - zadanie 1.1

Cel ćwiczenia

Celem ćwiczenia było stworzenie programów w języku C oraz Python, służących do komunikowania się między sobą przy wykorzystaniu gniazd sieciowych oraz protokołu UDP, zarówno przy użyciu adresów IPv4, jak i IPv6. Należy zaznaczyć, że programy te z założenia mają potrafić komunikować się ze sobą niezależnie od języka, w jakim zostały napisane – a więc serwer napisany w Pythonie potrafi komunikować się z klientem napisanym w C i vice versa.

Implementacja

W każdym z języków stworzone zostały dwa programy – jeden z nich jest implementacją klienta, a drugi serwera. Klient wysyła wiadomości do serwera – najpierw posługując się adresem IPv4, następnie IPv6. Serwer przez cały czas nasłuchuje na połączenia na dwóch gniazdach – jedno z nich obsługuje połączenia IPv4, drugie – IPv6. Oba gniazda nasłuchują na tym samym porcie. Serwer napisany w języku C działa na jednym wątku, a wyborem obsługiwanego gniazda zajmuje się funkcja `select()`. Serwer w języku Python został natomiast stworzony jako program dwuwątkowy, gdzie jeden z wątków nasłuchuje na połączenia z adresów IPv4, natomiast drugi – z adresów IPv6.

Zarówno klient, jak i serwer informują na bieżąco o przesyłanych i odbieranych wiadomościach poprzez stosowne komunikaty wysyłane na standardowe wyjście.

Wnioski

Mechanizm gniazd umożliwia relatywnie prostą realizację komunikacji sieciowej między dwoma programami. Dzięki temu, że programy korzystają z zapewnionej implementacji gniazd BSD, programista nie tylko może skorzystać z gotowych rozwiązań implementujących gniazda, ale nie musi się też martwić o niskopoziomowe różnice w formie zapisu poszczególnych zmiennych, ponieważ zajmują się tym za niego poszczególne warstwy modelu ISO/OSI.

Chcąc zrealizować program obsługujący kilka gniazd jednocześnie, nie musimy realizować obsługi każdego z gniazd w osobnym procesie, ani nawet w osobnym wątku – używane języki zapewniają konstrukcje pozwalające obsługiwać wiele gniazd w jednym wątku. Jedną z takich konstrukcji jest funkcja `select()`, która pozwala nasłuchiwać na wielu gniazdach równocześnie, nie tworząc przy tym ogromnej ilości wątków oczekujących na odbiór komunikatów sieciowych. Może to być przydatne szczególnie w sytuacjach, w których komunikaty przychodzą rzadko. Nawet jeżeli zdarzyłoby się, że w trakcie obsługi komunikatu na jednym z gniazd, drugie gniazdo zgłosiłoby komunikat do obsłużenia, to zostanie ono umieszczone w kolejce gniazd oczekujących na obsługę. Zakładając, że komunikaty są odbierane relatywnie rzadko, jest to optymalne rozwiązanie problemu.

```
mmatak@DESKTOP-FBMC4R5:/mnt/c/Users/micha/HOME/Studia/PSI/PSI$ python3 Z1/python/server.py
Server IPv4 is waiting for data...
Server IPv6 is waiting for data...
Received packet from 127.0.0.1:58605
Data: Message no. 0.
Data size: 14
Received packet from 127.0.0.1:58605
Data: Message no. 1.
Data size: 14
Received packet from 127.0.0.1:58605
Data: Message no. 2.
Data size: 14
Received packet from 127.0.0.1:58605
Data: Message no. 3.
Data size: 14
Received packet from 127.0.0.1:58605
Data: Message no. 4.
Data size: 14
```

```
mmatak@DESKTOP-FBMC4R5:/mnt/c/Users/micha/HOME/Studia/PSI/PSI/Z1/c$ ./clientv4
Client finished its job.mmatak@DESKTOP-FBMC4R5:/mnt/c/Users/micha/HOME/Studia/PSI/PSI/Z1/c$
```

Część druga - zadanie 1.2

Cel ćwiczenia

Zadanie polegało na modyfikacji stworzonego programu w celu sprawdzenia maksymalnej wielkości wysyłanego datagramu. Najpierw należało wysyłać kolejne datagramy o wielkości będącej kolejnymi potęgami dwójki, aż do niepowodzenia, a następnie znaleźć maksymalną wielkość wiadomości możliwej do wysłania.

Implementacja

Początkowo wysyłano datagramy, których wielkość była kolejnymi potęgami dwójki. Gdy wielkość pakietu osiągnęła wielkość większą niż maksymalna możliwa do wysłania, sprawdzono poprzez wyszukiwanie binarne graniczne wielkości datagramów możliwych do wysłania – zarówno dla IPv4, jak i IPv6. Klient informuje na bieżąco o powodzeniu, czy też niepowodzeniu wysłania datagramu poprzez stosowne komunikaty w konsoli.

Wnioski

Okazuje się, że maksymalna wielkość datagramu możliwego do wysłania różni się nieznacznie pomiędzy IPv4 i IPv6. Co więcej, otrzymano różne wartości w zależności od systemu, na którym program był uruchamiany (konkretnie: macOS i Windows 10). Wielkość datagramu w przypadku IPv4 jest mniejsza o 20 bajtów w stosunku do IPv6. Wyznaczony rozmiar pakietu dla IPv4, wynosi 65507, co jest spowodowane obecnością 20-bajtowego nagłówka IP i 8-bajtowego nagłówka UDP. Łącznie liczba ta sumuje się do 655035, która jest teoretyczną maksymalną wielkością datagramu UDP na systemie Windows. Na systemie macOS wartość ta wynosi 9264, co też zostało sprawdzone w trakcie testów. W przypadku protokołu IPv6 wartości te są różne, głównie przez różnice w wielkości nagłówków poszczególnych protokołów.

```
##### Client v4 #####

***** Doubling packet size every iteration *****

Successfully sent datagram of size 1
Successfully sent datagram of size 2
Successfully sent datagram of size 4
Successfully sent datagram of size 8
Successfully sent datagram of size 16
Successfully sent datagram of size 32
Successfully sent datagram of size 64
Successfully sent datagram of size 128
Successfully sent datagram of size 256
Successfully sent datagram of size 512
Successfully sent datagram of size 1024
Successfully sent datagram of size 2048
Successfully sent datagram of size 4096
Successfully sent datagram of size 8192
Successfully sent datagram of size 16384
Successfully sent datagram of size 32768
Failed to send datagram of size 65536

The greatest length of a message described as a power of 2 equals 32768

***** Looking for maximum message size for IPv4 *****

Successfully sent datagram of size 49152
Successfully sent datagram of size 57344
Successfully sent datagram of size 61440
Successfully sent datagram of size 63488
Successfully sent datagram of size 64512
Successfully sent datagram of size 65024
Successfully sent datagram of size 65280
Successfully sent datagram of size 65408
Successfully sent datagram of size 65472
Successfully sent datagram of size 65504
Failed to send datagram of size 65520
Failed to send datagram of size 65512
Failed to send datagram of size 65508
Successfully sent datagram of size 65506
Successfully sent datagram of size 65507

Maximum size of a message to send with IPv4 equals 65507
```

```
##### Client v6 #####

***** Doubling packet size every iteration *****

Successfully sent datagram of size 1
Successfully sent datagram of size 2
Successfully sent datagram of size 4
Successfully sent datagram of size 8
Successfully sent datagram of size 16
Successfully sent datagram of size 32
Successfully sent datagram of size 64
Successfully sent datagram of size 128
Successfully sent datagram of size 256
Successfully sent datagram of size 512
Successfully sent datagram of size 1024
Successfully sent datagram of size 2048
Successfully sent datagram of size 4096
Successfully sent datagram of size 8192
Successfully sent datagram of size 16384
Successfully sent datagram of size 32768
Failed to send datagram of size 65536

The greatest length of a message described as a power of 2 equals 32768

***** Looking for maximum message size for IPv6 *****

Successfully sent datagram of size 49152
Successfully sent datagram of size 57344
Successfully sent datagram of size 61440
Successfully sent datagram of size 63488
Successfully sent datagram of size 64512
Successfully sent datagram of size 65024
Successfully sent datagram of size 65280
Successfully sent datagram of size 65408
Successfully sent datagram of size 65472
Successfully sent datagram of size 65504
Successfully sent datagram of size 65520
Failed to send datagram of size 65528
Successfully sent datagram of size 65524
Successfully sent datagram of size 65526
Successfully sent datagram of size 65527

Maximum size of a message to send with IPv6 equals 65527
```

```
python - -zsh - 80x40
~/PycharmProject/PSI/Z1/python - -zsh
##### Client v4 #####

***** Doubling packet size every iteration *****

Successfully sent datagram of size 1
Successfully sent datagram of size 2
Successfully sent datagram of size 4
Successfully sent datagram of size 8
Successfully sent datagram of size 16
Successfully sent datagram of size 32
Successfully sent datagram of size 64
Successfully sent datagram of size 128
Successfully sent datagram of size 256
Successfully sent datagram of size 512
Successfully sent datagram of size 1024
Successfully sent datagram of size 2048
Successfully sent datagram of size 4096
Successfully sent datagram of size 8192
Failed to send datagram of size 16384

The greatest length of a message described as a power of 2 equals 8192

***** Looking for maximum message size for IPv4 *****

Failed to send datagram of size 12288
Failed to send datagram of size 10240
Successfully sent datagram of size 9216
Failed to send datagram of size 9728
Failed to send datagram of size 9472
Failed to send datagram of size 9344
Failed to send datagram of size 9280
Failed to send datagram of size 9248
Failed to send datagram of size 9232
Failed to send datagram of size 9224
Failed to send datagram of size 9220
Failed to send datagram of size 9218
Failed to send datagram of size 9217

Maximum size of a message to send with IPv4 equals 9216
```

```
python - -zsh - 80x40
~/PycharmProject/PSI/Z1/python - -zsh
##### Client v6 #####

***** Doubling packet size every iteration *****

Successfully sent datagram of size 1
Successfully sent datagram of size 2
Successfully sent datagram of size 4
Successfully sent datagram of size 8
Successfully sent datagram of size 16
Successfully sent datagram of size 32
Successfully sent datagram of size 64
Successfully sent datagram of size 128
Successfully sent datagram of size 256
Successfully sent datagram of size 512
Successfully sent datagram of size 1024
Successfully sent datagram of size 2048
Successfully sent datagram of size 4096
Successfully sent datagram of size 8192
Failed to send datagram of size 16384

The greatest length of a message described as a power of 2 equals 8192

***** Looking for maximum message size for IPv6 *****

Failed to send datagram of size 12288
Failed to send datagram of size 10240
Successfully sent datagram of size 9216
Failed to send datagram of size 9728
Failed to send datagram of size 9472
Failed to send datagram of size 9344
Failed to send datagram of size 9280
Failed to send datagram of size 9248
Failed to send datagram of size 9232
Failed to send datagram of size 9224
Failed to send datagram of size 9220
Failed to send datagram of size 9218
Failed to send datagram of size 9217

Maximum size of a message to send with IPv6 equals 9216
```