

Laboratorium Projektowania Aplikacji Internetowych 1
Wydział Elektrotechniki Automatyki i Informatyki
Politechnika Świętokrzyska

Studia: Stacjonarne I stopnia	Kierunek: Informatyka	
Data wykonania: 07.11.2017	Zespół: 06	Grupa: 3ID13B
Godzina wykonania: 8:00 – 9:30	Skład zespołu: 1. Bartłomiej Osak 2. Tomasz Pasternak	
Numer laboratorium: 2	Temat ćwiczenia: Protokoły	

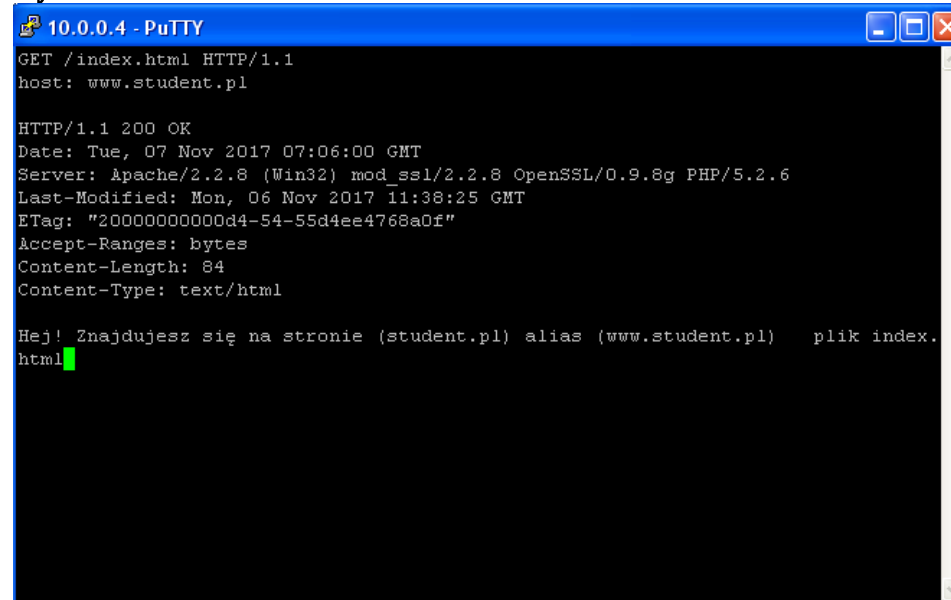
1. Połączyć się telnetem z serwerem www:

a) Połączyć się ze wszystkimi hostami wirtualnymi.

Wpisywane komendy:

```
GET /index.html HTTP/1.1  
host: www.student.pl
```

Wynik zadziałania:

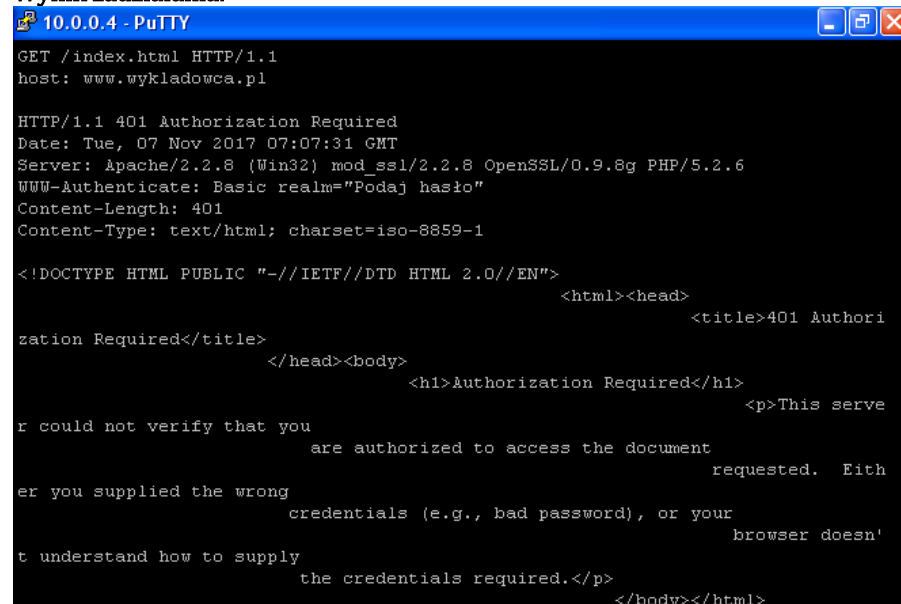


```
10.0.0.4 - PuTTY  
GET /index.html HTTP/1.1  
host: www.student.pl  
  
HTTP/1.1 200 OK  
Date: Tue, 07 Nov 2017 07:06:00 GMT  
Server: Apache/2.2.8 (Win32) mod_ssl/2.2.8 OpenSSL/0.9.8g PHP/5.2.6  
Last-Modified: Mon, 06 Nov 2017 11:38:25 GMT  
ETag: "2000000000d4-54-55d4ee4768a0f"  
Accept-Ranges: bytes  
Content-Length: 84  
Content-Type: text/html  
  
Hej! Znajdujesz się na stronie (student.pl) alias (www.student.pl) plik index.  
html
```

Wpisywane komendy:

```
GET /index.html HTTP/1.1  
host: www.wykladowca.pl
```

Wynik zadziałania:



```
10.0.0.4 - PuTTY  
GET /index.html HTTP/1.1  
host: www.wykladowca.pl  
  
HTTP/1.1 401 Authorization Required  
Date: Tue, 07 Nov 2017 07:07:31 GMT  
Server: Apache/2.2.8 (Win32) mod_ssl/2.2.8 OpenSSL/0.9.8g PHP/5.2.6  
WWW-Authenticate: Basic realm="Podaj hasło"  
Content-Length: 401  
Content-Type: text/html; charset=iso-8859-1  
  
<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">  
    <html><head>  
        <title>401 Authori  
zation Required</title>  
    </head><body>  
        <h1>Authorization Required</h1>  
        <p>This serve  
r could not verify that you  
are authorized to access the document  
requested. Eith  
er you supplied the wrong  
credentials (e.g., bad password), or your  
browser doesn'  
t understand how to supply  
the credentials required.</p>  
    </body></html>
```

Wniosek:

Celem połączenia się z 10.0.0.4 należało ustawić port na wartość 80.

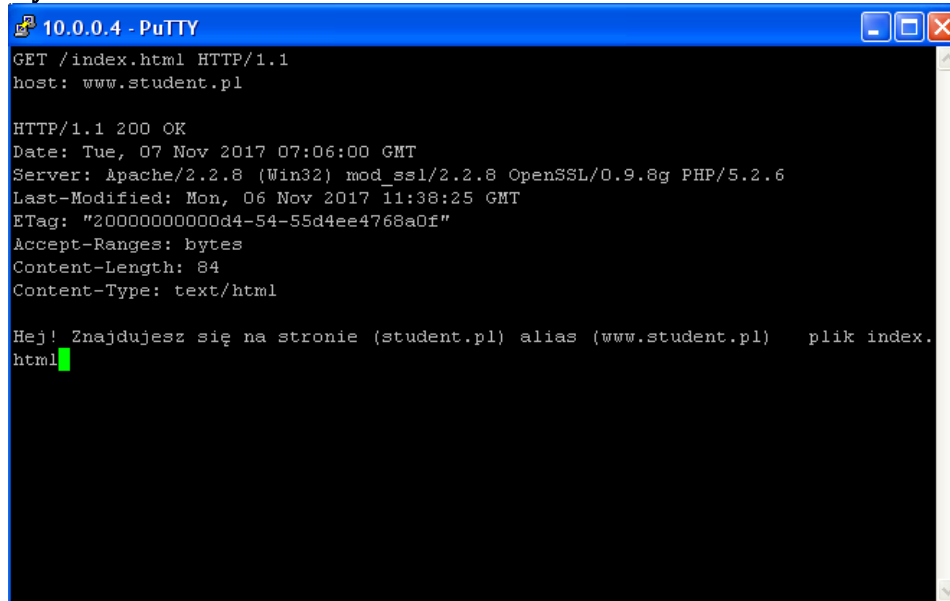
W przypadku pobrania strony index.html z www.student.pl serwer zwraca komunikat o numerze 200, który informuje o prawidłowym przebiegu zapytania. W przypadku www.wykladowca.pl serwer zwraca błąd 401, ponieważ dostęp na tą stronę wymaga podania hasła.

b) Pobrać stronę index.html.

Wpisywane komendy:

```
GET /index.html HTTP/1.1
host: www.student.pl
```

Wynik zadziałania:



```
10.0.0.4 - PuTTY
GET /index.html HTTP/1.1
host: www.student.pl

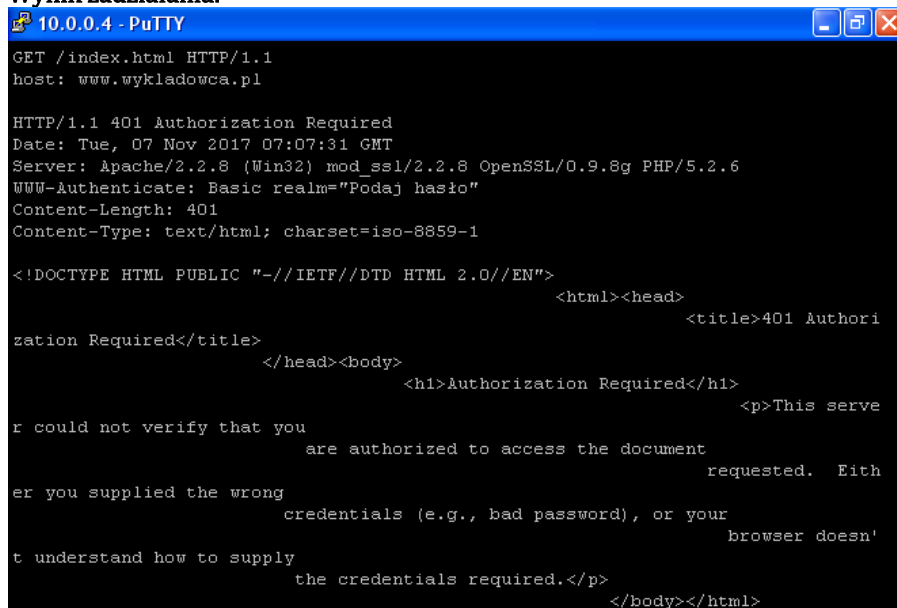
HTTP/1.1 200 OK
Date: Tue, 07 Nov 2017 07:06:00 GMT
Server: Apache/2.2.8 (Win32) mod_ssl/2.2.8 OpenSSL/0.9.8g PHP/5.2.6
Last-Modified: Mon, 06 Nov 2017 11:38:25 GMT
ETag: "20000000000d4-54-55d4ee4768a0f"
Accept-Ranges: bytes
Content-Length: 84
Content-Type: text/html

Hej! Znajdujesz się na stronie (student.pl) alias (www.student.pl) plik index.html
```

Wpisywane komendy:

```
GET /index.html HTTP/1.1
host: www.wykladowca.pl
```

Wynik zadziałania:



```
10.0.0.4 - PuTTY
GET /index.html HTTP/1.1
host: www.wykladowca.pl

HTTP/1.1 401 Authorization Required
Date: Tue, 07 Nov 2017 07:07:31 GMT
Server: Apache/2.2.8 (Win32) mod_ssl/2.2.8 OpenSSL/0.9.8g PHP/5.2.6
WWW-Authenticate: Basic realm="Podaj hasło"
Content-Length: 401
Content-Type: text/html; charset=iso-8859-1

<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
    <html><head>
        <title>401 Authori
zation Required</title>
    </head><body>
        <h1>Authorization Required</h1>
        <p>This serve
r could not verify that you
are authorized to access the document
requested. Eith
er you supplied the wrong
credentials (e.g., bad password), or your
browser doesn'
t understand how to supply
the credentials required.</p>
    </body></html>
```

Wniosek:

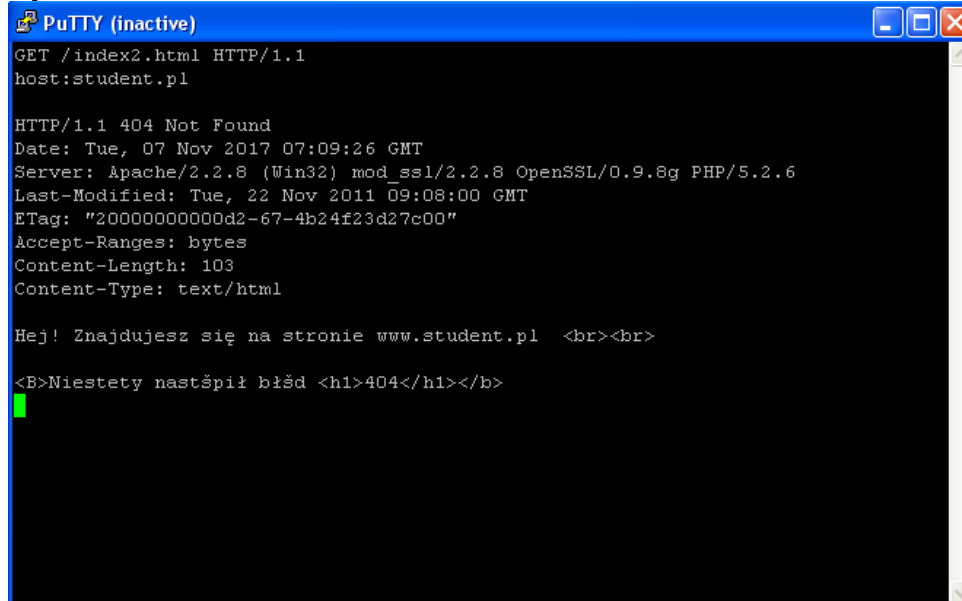
W przypadku pobrania strony index.html z www.student.pl serwer zwraca komunikat o numerze 200, który informuje o prawidłowym przebiegu zapytania. W przypadku www.wykladowca.pl serwer zwraca błąd 401, ponieważ dostęp na tą stronę wymaga podania hasła.

- c) Pobrać stronę, która istnieje na serwerze i odczytać kod błędu.

Wpisywane komendy:

```
GET /index2.html HTTP/1.1  
host: www.student.pl
```

Wynik zadziałania:



```
PuTTY (inactive)  
GET /index2.html HTTP/1.1  
host:student.pl  
  
HTTP/1.1 404 Not Found  
Date: Tue, 07 Nov 2017 07:09:26 GMT  
Server: Apache/2.2.8 (Win32) mod_ssl/2.2.8 OpenSSL/0.9.8g PHP/5.2.6  
Last-Modified: Tue, 22 Nov 2011 09:08:00 GMT  
ETag: "2000000000d2-67-4b24f23d27c00"  
Accept-Ranges: bytes  
Content-Length: 103  
Content-Type: text/html  
  
Hej! Znajdujesz się na stronie www.student.pl <br><br>  
<B>Niestety nastąpił błąd <h1>404</h1></b>
```

Wniosek:

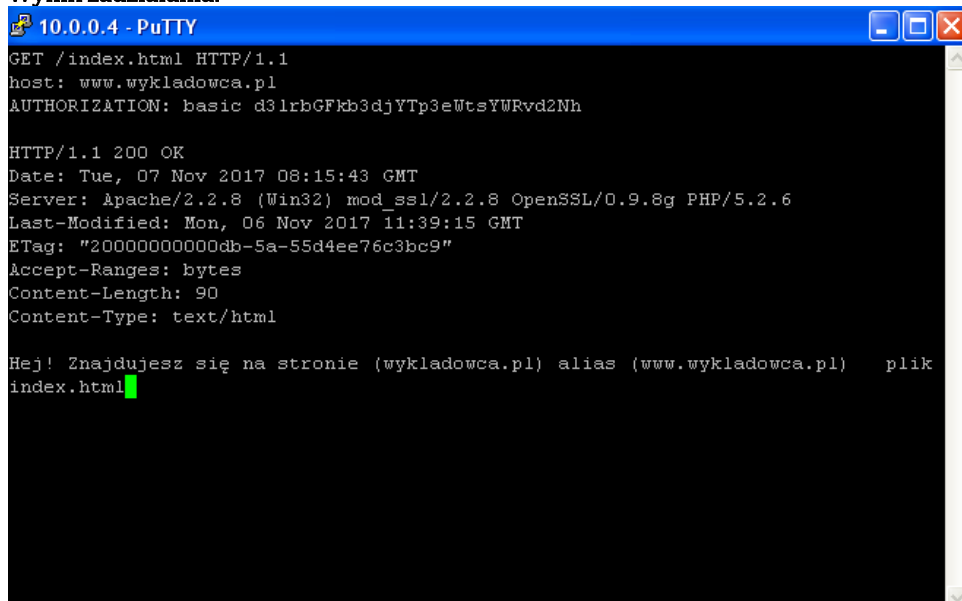
Chcąc pobrać stronę index2.html, która nie znajdowała się na serwerze zwracany jest błąd 404 – nie znaleziono.

- d) Pobrać stronę zabezpieczoną na hasło.

Wpisywane komendy:

```
GET /index.html HTTP/1.1  
host: www.wykladowca.pl  
AUTHORIZATION: basic d3lrbGFkb3djYTp3eWtsYWVvd2Nh
```

Wynik zadziałania:



```
10.0.0.4 - PuTTY  
GET /index.html HTTP/1.1  
host: www.wykladowca.pl  
AUTHORIZATION: basic d3lrbGFkb3djYTp3eWtsYWVvd2Nh  
  
HTTP/1.1 200 OK  
Date: Tue, 07 Nov 2017 08:15:43 GMT  
Server: Apache/2.2.8 (Win32) mod_ssl/2.2.8 OpenSSL/0.9.8g PHP/5.2.6  
Last-Modified: Mon, 06 Nov 2017 11:39:15 GMT  
ETag: "2000000000db-5a-55d4ee76c3bc9"  
Accept-Ranges: bytes  
Content-Length: 90  
Content-Type: text/html  
  
Hej! Znajdujesz się na stronie (wykladowca.pl) alias (www.wykladowca.pl) plik  
index.html
```

Wniosek:

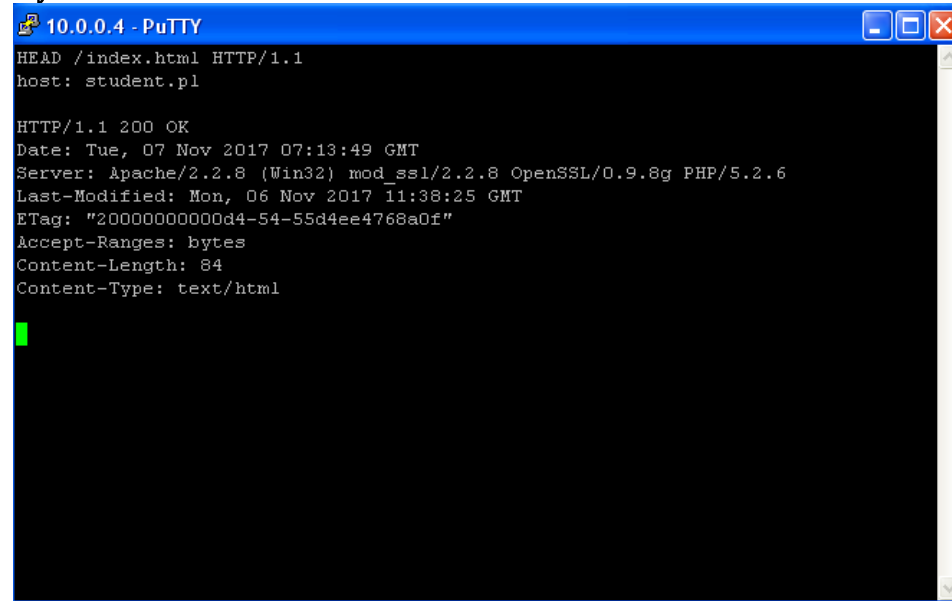
W celu pobrania strony zabezpieczonej na hasło należało na początku zakodować ciąg: wyklawowca:wykladowca na kodowanie base64. Następnie w terminalu należało użyć nagłówka żądania AUTHORIZATION celem podania odpowiedniego zakodowanego hasła. Po wykonaniu się serwer zwraca komunikat o numerze 200 oraz zawartość strony index.html.

e) Pobrać tylko nagłówek odpowiedzi z serwera.

Wpisywane komendy:

```
HEAD /index.html HTTP/1.1  
host: student.pl
```

Wynik zadziałania:



```
10.0.0.4 - PuTTY  
HEAD /index.html HTTP/1.1  
host: student.pl  
  
HTTP/1.1 200 OK  
Date: Tue, 07 Nov 2017 07:13:49 GMT  
Server: Apache/2.2.8 (Win32) mod_ssl/2.2.8 OpenSSL/0.9.8g PHP/5.2.6  
Last-Modified: Mon, 06 Nov 2017 11:38:25 GMT  
ETag: "2000000000d4-54-55d4ee4768a0f"  
Accept-Ranges: bytes  
Content-Length: 84  
Content-Type: text/html
```

Wniosek:

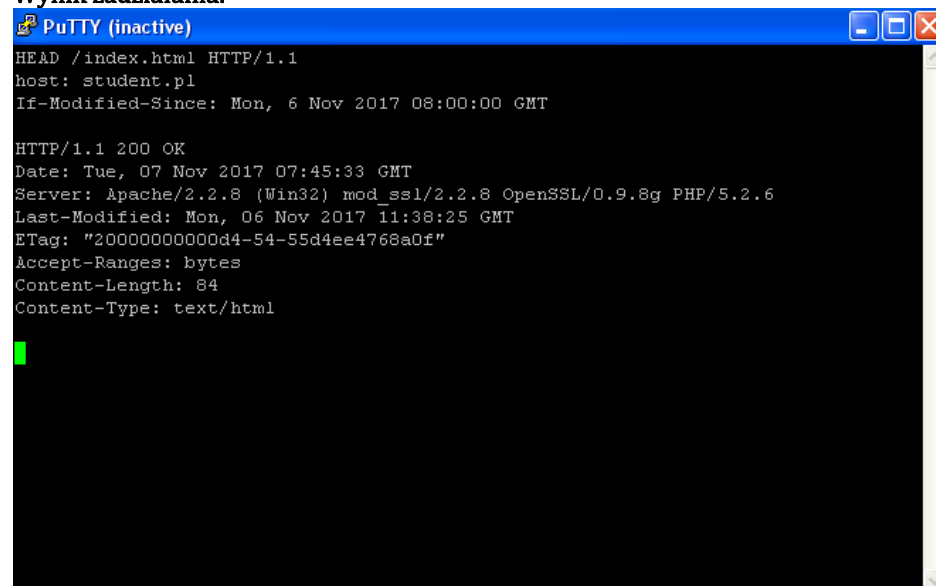
Dzięki użyciu dyrektywy HEAD możemy odczytać najważniejsze informacje zwracane przez serwer bez zawartości pobieranej strony, tu: index.html. W naszym przypadku serwer zwrócił kod o numerze 200, czyli zapytanie zostało prawidłowo obsłużone.

f) Sprawdzenie ważności strony. Sprawdzić, jakie jest pole Last-Modified.

Wpisywane komendy:

```
HEAD /index.html HTTP/1.1  
host: student.pl  
If-Modified-Since: Mon, 6 Nov 2017 08:00:00 GMT
```

Wynik zadziałania:



```
PuTTY (inactive)  
HEAD /index.html HTTP/1.1  
host: student.pl  
If-Modified-Since: Mon, 6 Nov 2017 08:00:00 GMT  
  
HTTP/1.1 200 OK  
Date: Tue, 07 Nov 2017 07:45:33 GMT  
Server: Apache/2.2.8 (Win32) mod_ssl/2.2.8 OpenSSL/0.9.8g PHP/5.2.6  
Last-Modified: Mon, 06 Nov 2017 11:38:25 GMT  
ETag: "2000000000d4-54-55d4ee4768a0f"  
Accept-Ranges: bytes  
Content-Length: 84  
Content-Type: text/html
```

Wniosek:

W polu Last-Modified widnieje data ostatniej zmiany w pliku index.html. Jest to Mon, 6 Nov 2017 11:38:00 GMT. Wpisując powyższe komendy serwer zwrócił kod 200 o prawidłowym zapytaniu. W przypadku umieszczenia w if-modified-since daty przyszłej od pola last-modified serwer zwróciłby kod 304.

Wpisując komendę pobrania pliku a1.gif serwer zwróci nam go w postaci tekstu. Kopiując zwrócony tekst do notatnika i zapisując go z rozszerzeniem .gif niestety nadal nie odczytamy go. Należy dokonać zmian w konfiguracji programu Putty.

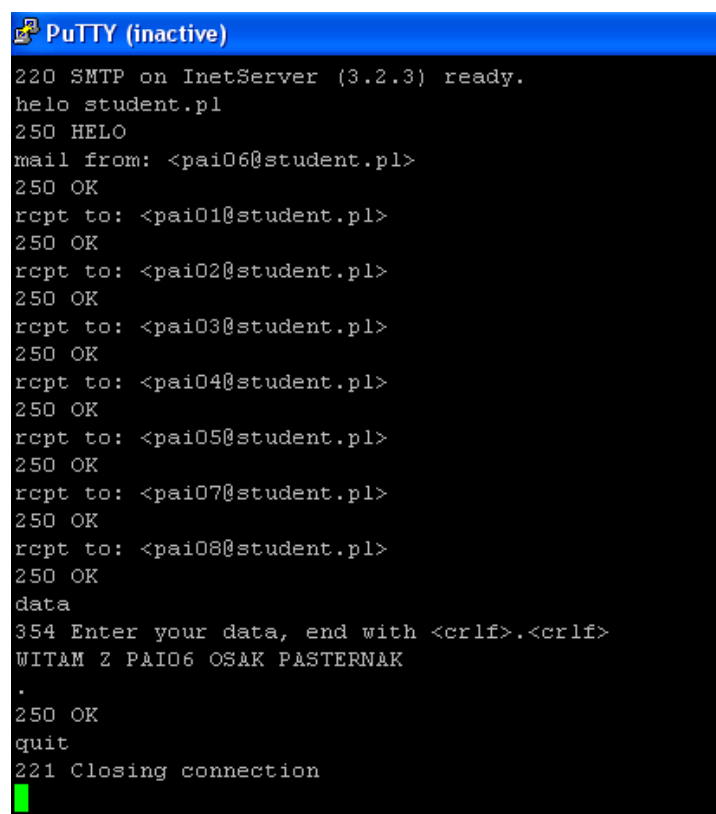
2. Połączyć się telnetem z serwerem pocztowym.

a) Rozesłać listy do wszystkich uczestników zajęć.

Wpisywane komendy:

```
HELO student.pl
mail from: <pai06@student.pl>
rcpt to: <pai01@student.pl>
rcpt to: <pai02@student.pl>
rcpt to: <pai03@student.pl>
rcpt to: <pai04@student.pl>
rcpt to: <pai05@student.pl>
rcpt to: <pai07@student.pl>
rcpt to: <pai08@student.pl>
data
WITAM Z PAI06 OSAK PASTERNAK
.
quit
```

Wynik zadziałania:



```
PuTTY (inactive)
220 SMTP on InetServer (3.2.3) ready.
helo student.pl
250 HELO
mail from: <pai06@student.pl>
250 OK
rcpt to: <pai01@student.pl>
250 OK
rcpt to: <pai02@student.pl>
250 OK
rcpt to: <pai03@student.pl>
250 OK
rcpt to: <pai04@student.pl>
250 OK
rcpt to: <pai05@student.pl>
250 OK
rcpt to: <pai07@student.pl>
250 OK
rcpt to: <pai08@student.pl>
250 OK
data
354 Enter your data, end with <crLf>.<crLf>
WITAM Z PAI06 OSAK PASTERNAK
.
250 OK
quit
221 Closing connection
```

Wniosek:

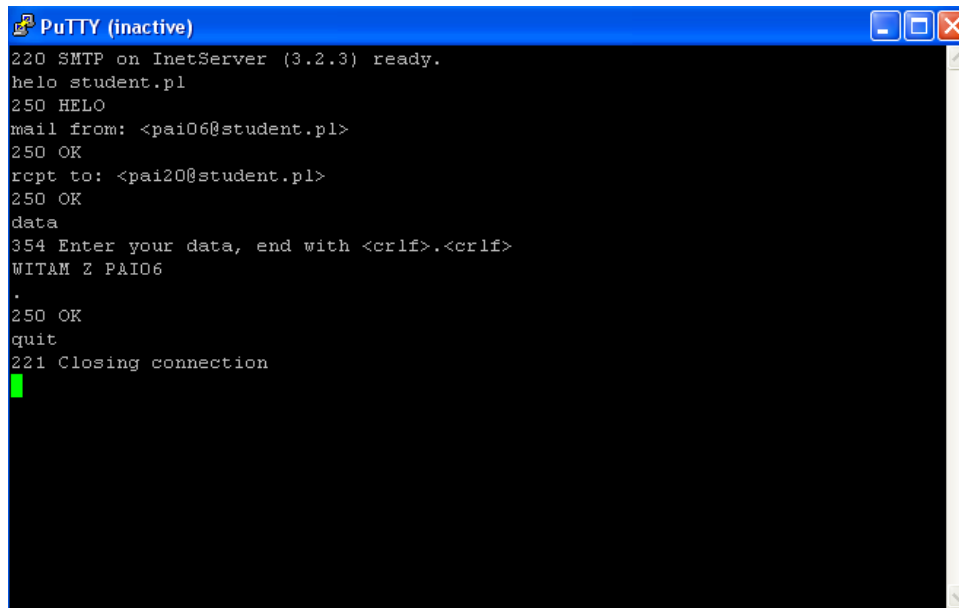
Celem połączenia się z SMTP należało zmienić port dla 10.0.0.4 na wartość 25. Następnie po wpisywaniu kolejnych komend tj. helo (przywitanie się z hostem), mail from (określenie nadawcy), rcpt to (określenie odbiorcy), data (komenda umożliwiająca wpisywanie wiadomości) – serwer zwraca kod 250 z informacją OK, czyli proces potoczył się prawidłowo.

b) Wysłać list do nieistniejącego użytkownika.

Wpisywane komendy:

```
HELO student.pl
mail from: <pai06@student.pl>
rcpt to: <pai20@student.pl>
data
WITAM Z PAI06 OSAK PASTERNAK
.
quit
```

Wynik zadziałania:



```
PuTTY (inactive)
220 SMTP on InetServer (3.2.3) ready.
helo student.pl
250 HELO
mail from: <pai06@student.pl>
250 OK
rcpt to: <pai20@student.pl>
250 OK
data
354 Enter your data, end with <crLf>.<crLf>
WITAM Z PAI06
.
250 OK
quit
221 Closing connection
```

Wniosek:

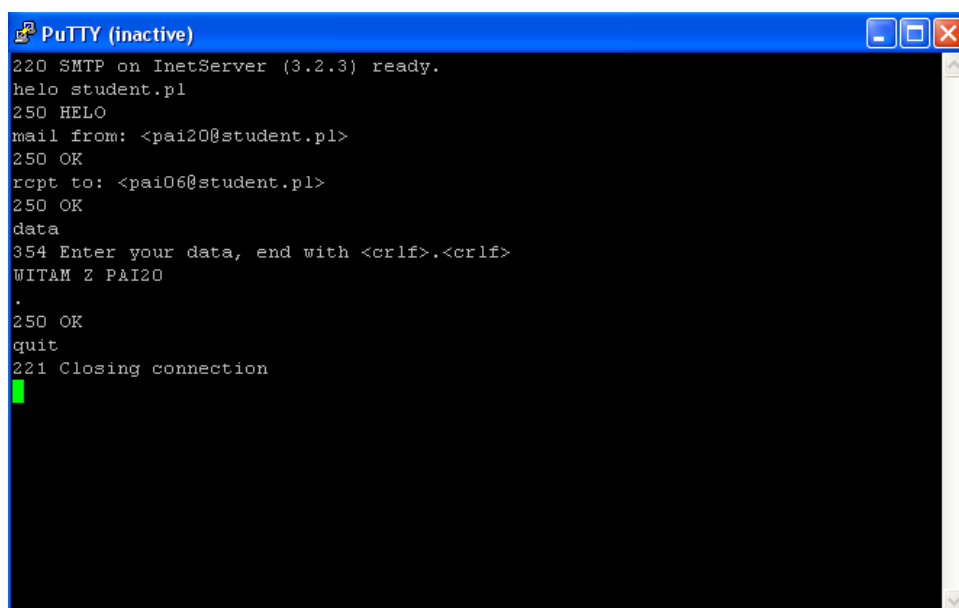
Po wpisaniu komendy helo oraz określenia nadawcy, w polu rcpt to wpisaliśmy nieistniejący adres mailowy. Następnie po wpisaniu polecenia data napisaliśmy wiadomość. Serwer pomimo nieistniejącego odbiorcy zwraca komunikat o numerze 250 z informacją OK.

c) Wysłać list od nieistniejącego użytkownika.

Wpisywane komendy:

```
HELO student.pl
mail from: <pai20@student.pl>
rcpt to: <pai06@student.pl>
data
WITAM Z PAI20
.
quit
```

Wynik zadziałania:



```
PuTTY (inactive)
220 SMTP on InetServer (3.2.3) ready.
helo student.pl
250 HELO
mail from: <pai20@student.pl>
250 OK
rcpt to: <pai06@student.pl>
250 OK
data
354 Enter your data, end with <crLf>.<crLf>
WITAM Z PAI20
.
250 OK
quit
221 Closing connection
```


Wniosek:

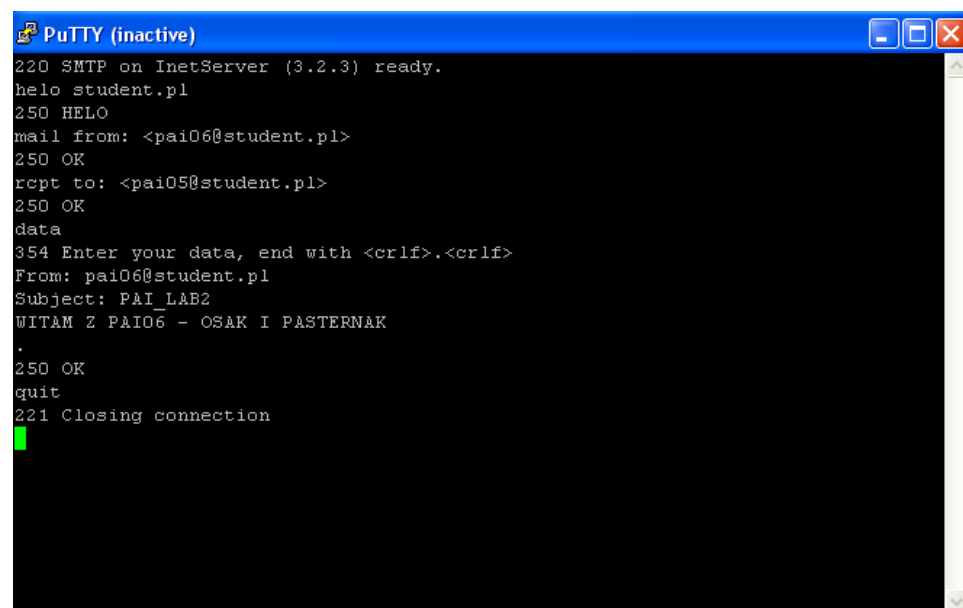
Po wpisaniu komendy helo oraz określenia nadawcy w polu mail from wpisaliśmy nieistniejący adres mailowy. W polu rcpt to wpisaliśmy własny adres -pai06@student.pl. Następnie po wpisaniu polecenia data napisaliśmy wiadomość. Serwer pomimo nieistniejącego nadawcy zwraca komunikat o numerze 250 z informacją OK.

d) Wysłać list z polami nagłówka.

Wpisywane komendy:

```
HELO student.pl
mail from: <pai06@student.pl>
rcpt to: <pai05@student.pl>
data
From: pai06@student.pl
Subject: PAI_LAB2
WITAM Z PAI06 - OSAK PASTERNAK
.
quit
```

Wynik zadziałania:



```
PuTTY (inactive)
220 SMTP on InetServer (3.2.3) ready.
helo student.pl
250 HELO
mail from: <pai06@student.pl>
250 OK
rcpt to: <pai05@student.pl>
250 OK
data
354 Enter your data, end with <crLf>.<crLf>
From: pai06@student.pl
Subject: PAI_LAB2
WITAM Z PAI06 - OSAK I PASTERNAK
.
250 OK
quit
221 Closing connection
```

Wniosek:

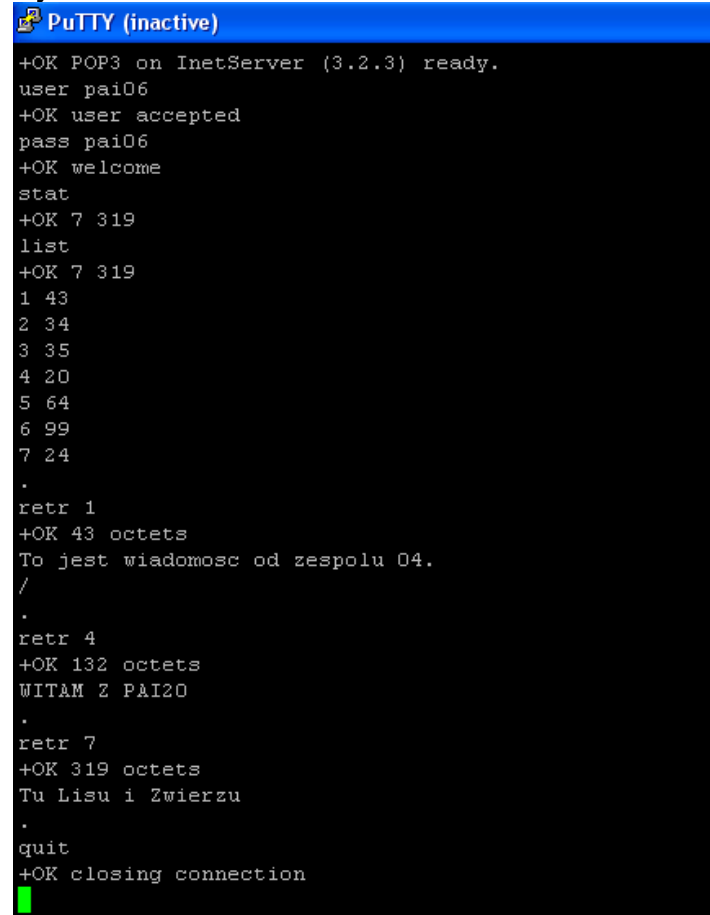
Po przywitaniu się komenda helo wysłaliśmy mail do sąsiedniego zespołu z numerem 05. Po wpisaniu komendy data użyliśmy dwóch pól nagłówka – From (określenie skąd dotarł mail) oraz Subject (określenie tematu wiadomości). Po zakończeniu kropką serwer zwraca komunikat 250 o treści OK.

e) Sprawdzić zawartość swojej skrzynki pocztowej.

Wpisywane komendy:

```
user pai06
pass pai06
stat
list
retr 1
retr 4
retr 7
quit
```

Wynik zadziałania:

A screenshot of a PuTTY terminal window titled "PuTTY (inactive)". The terminal shows a POP3 session with the following commands and responses:

```
+OK POP3 on InetServer (3.2.3) ready.  
user pai06  
+OK user accepted  
pass pai06  
+OK welcome  
stat  
+OK 7 319  
list  
+OK 7 319  
1 43  
2 34  
3 35  
4 20  
5 64  
6 99  
7 24  
.  
retr 1  
+OK 43 octets  
To jest wiadomosc od zespolu 04.  
/  
.  
retr 4  
+OK 132 octets  
WITAM Z PAI20  
.  
retr 7  
+OK 319 octets  
Tu Lisu i Zwierzu  
.  
quit  
+OK closing connection
```

Wniosek:

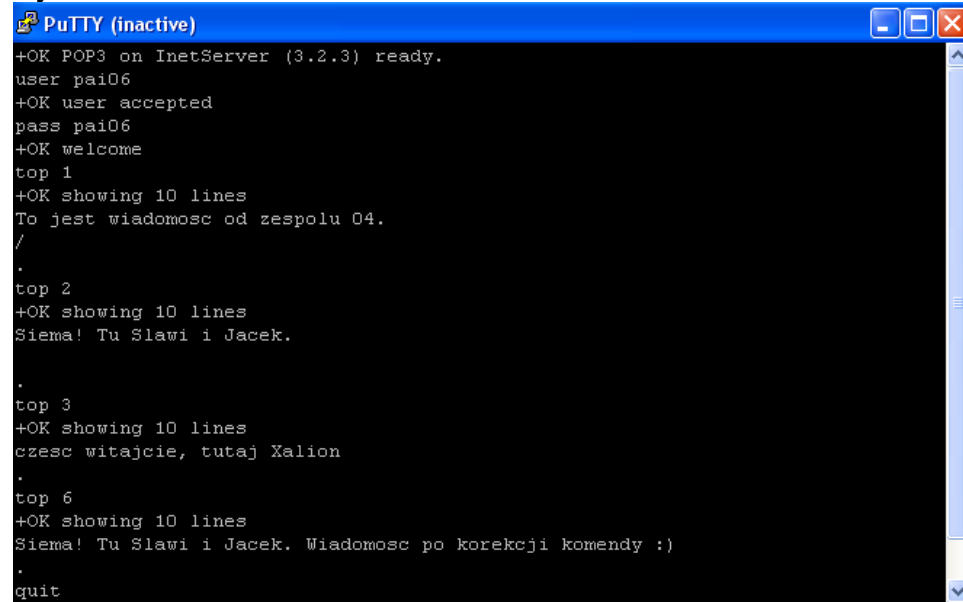
Celem odczytania skrzynki pocztowej należało w programie putty ustawić port dla 10.0.0.4 na wartość 110. Następnie należało się zalogować za pomocą dyrektyw user oraz pass. Następnie polecenie stat zwróciło nam ogólną liczbę listów w naszej skrzynce wraz z całym rozmiarem naszej skrzynki odbiorczej.. Potem wpisaliśmy polecenie list, które zwróciło nam listę wszystkich mail wraz z rozmiarem poszczególnego maila. Odczytaliśmy trzy maile (1,4,7) poleceniem retr. Po odczytaniu zakończyliśmy działanie poleceniem quit.

f) Odczytać nagłówki listów.

Wpisywane komendy:

```
user pai06  
pass pai06  
top 1  
top 2  
top 3  
top 6  
quit
```

Wynik zadziałania:



```
PuTTY (inactive)
+OK POP3 on InetServer (3.2.3) ready.
user pai06
+OK user accepted
pass pai06
+OK welcome
top 1
+OK showing 10 lines
To jest wiadomosc od zespolu 04.
/
.
top 2
+OK showing 10 lines
Siema! Tu Slawi i Jacek.
.
top 3
+OK showing 10 lines
czesc witajcie, tutaj Xalion
.
top 6
+OK showing 10 lines
Siema! Tu Slawi i Jacek. Wiadomosc po korekcji komendy :)
.
quit
```

Wniosek:

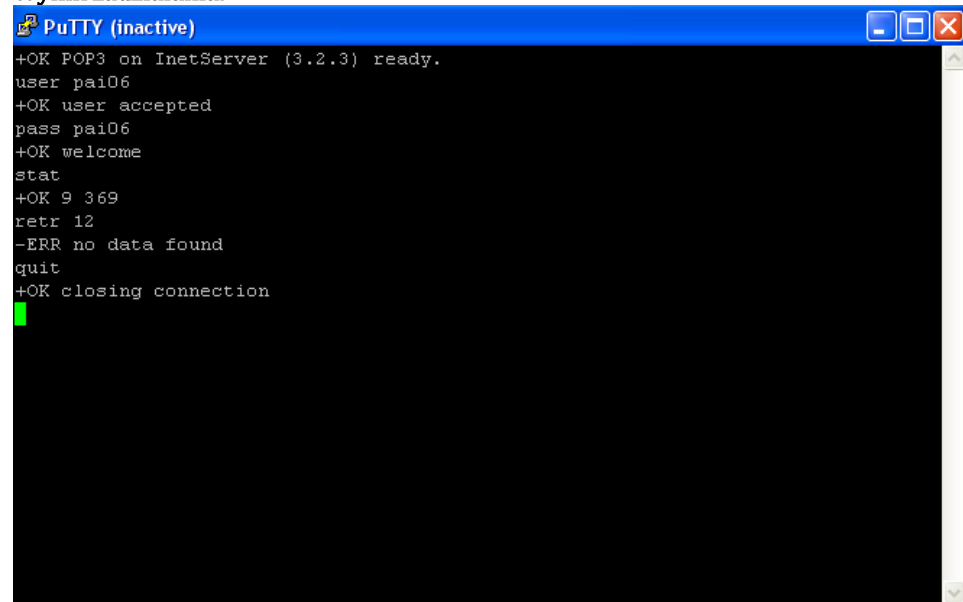
Odczyt nagłówek wykonujemy poleceniem top oraz wskazaniem numeru maila w naszej skrzynce odbiorczej. Odczytaliśmy 1,2,3 oraz 6 mail. Po wpisaniu np. top 1 zwracany jest nagłówek listu.

g) Spróbować przeczytać nieistniejący list.

Wpisywane komendy:

```
user pai06
pass pai06
stat
retr 12
quit
```

Wynik zadziałania:



```
PuTTY (inactive)
+OK POP3 on InetServer (3.2.3) ready.
user pai06
+OK user accepted
pass pai06
+OK welcome
stat
+OK 9 369
retr 12
-ERR no data found
quit
+OK closing connection
```

Wniosek:

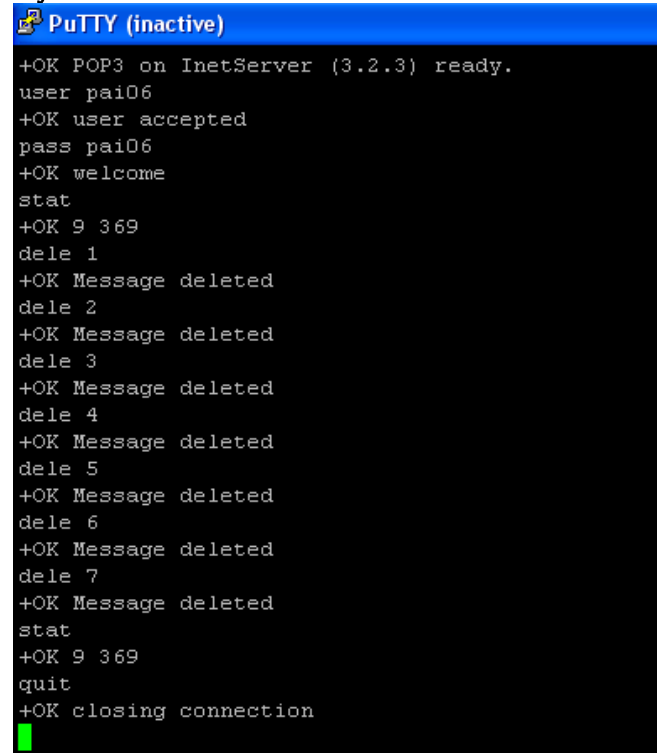
Chcąc przeczytać nieistniejący mail sprawdziliśmy na początku poleceniem stat, ile listów znajduje się w naszej skrzynce odbiorczej. Następnie poleceniem retr 12 chcieliśmy odczytać nieistniejący list (listów było mniej). Serwer w takim przypadku zwraca błąd: „—ERR no data found”.

h) Usunąć listy.

Wpisywane komendy:

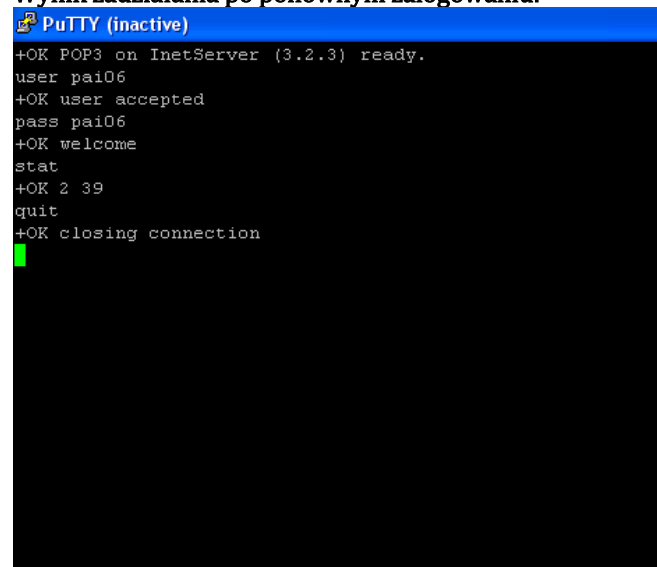
```
user pai06
pass pai06
stat
dele 1
dele 2
dele 3
dele 4
dele 5
dele 6
dele 7
stat
quit
```

Wynik zadziałania:



```
PuTTY (inactive)
+OK POP3 on InetServer (3.2.3) ready.
user pai06
+OK user accepted
pass pai06
+OK welcome
stat
+OK 9 369
dele 1
+OK Message deleted
dele 2
+OK Message deleted
dele 3
+OK Message deleted
dele 4
+OK Message deleted
dele 5
+OK Message deleted
dele 6
+OK Message deleted
dele 7
+OK Message deleted
stat
+OK 9 369
quit
+OK closing connection
```

Wynik zadziałania po ponownym zalogowaniu:



```
PuTTY (inactive)
+OK POP3 on InetServer (3.2.3) ready.
user pai06
+OK user accepted
pass pai06
+OK welcome
stat
+OK 2 39
quit
+OK closing connection
```

Wniosek:

Po zalogowaniu i wpisaniu komendy stat sprawdziliśmy, ile listów zalega w naszej skrzynce odbiorczej. Następnie usunęliśmy poleceniem dele siedem wiadomości, dwie wiadomości pozostały nieusunięte. Po ponownym wpisaniu stat serwer nadal zwracał, iż w skrzynce jest 9 wiadomości. Dopiero po ponownym zalogowaniu się do skrzynki i wpisaniu polecenia stat serwer zwrócił informację, że znajdują się tylko dwie wiadomości.

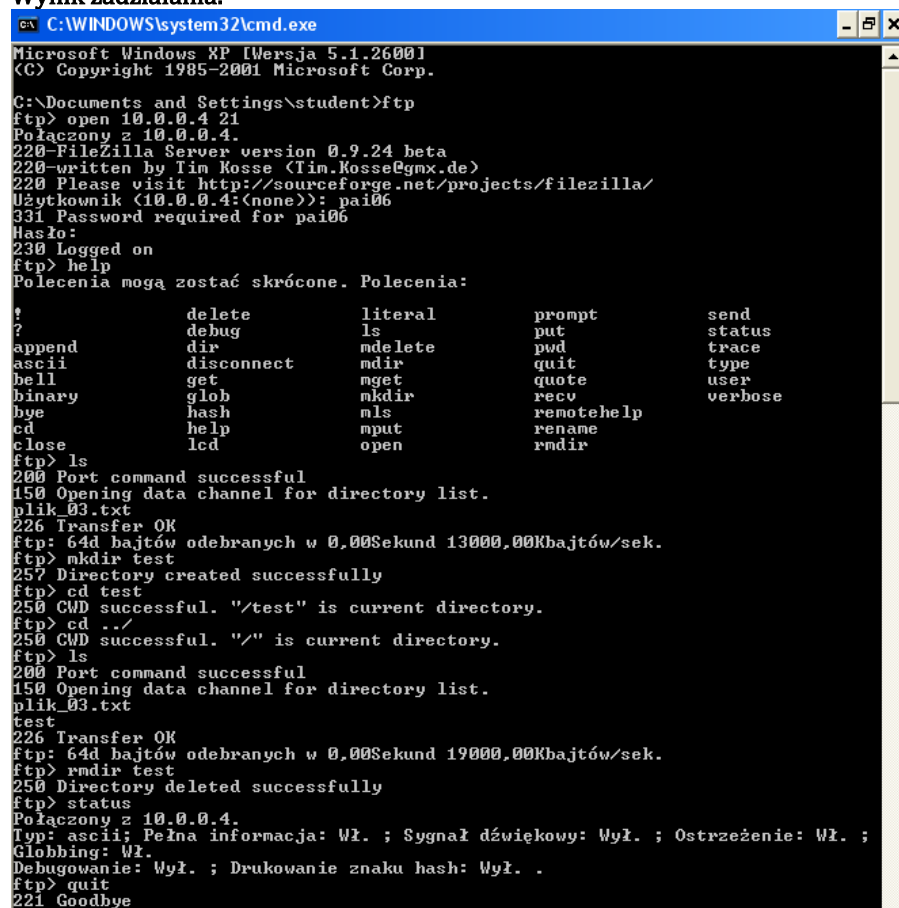
3. Połączyć się z protokołem FTP z serwerem ftp (Klient usługi).

a) Przetestować działanie podstawowych komend protokołu ftp.

Wpisywane komendy:

```
open 10.0.0.4 21
pai06
pai06
help
ls
mkdir test
cd test
cd ../
ls
rmdir test
status
quit
```

Wynik zadziałania:



```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows XP [Wersja 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\Documents and Settings\student>ftp
ftp> open 10.0.0.4 21
Połączony z 10.0.0.4.
220-FileZilla Server version 0.9.24 beta
220-written by Tim Kosse <Tim.Kosse@gmx.de>
220 Please visit http://sourceforge.net/projects/filezilla/
Użytkownik (10.0.0.4:(none)): pai06
331 Password required for pai06
Hasło:
230 Logged on
ftp> help
Polecenia mogą zostać skrócone. Polecenia:
?      delete      literal      prompt      send
?      debug       ls           put         status
append dir           mdelete     pwd         trace
ascii disconnect mdir        quit        type
bell   get          mget        quote       user
binary glob         mkdir       recv        verbose
bye    hash         mls         remotehelp
cd     help         mput        rename
close  lcd          open        rmdir
ftp> ls
200 Port command successful
150 Opening data channel for directory list.
plik_03.txt
226 Transfer OK
ftp: 64d bajtów odebranych w 0.00Sekund 13000.00Kbajtów/sek.
ftp> mkdir test
257 Directory created successfully
ftp> cd test
250 CWD successful. "/test" is current directory.
ftp> cd ../
250 CWD successful. "/" is current directory.
ftp> ls
200 Port command successful
150 Opening data channel for directory list.
plik_03.txt
test
226 Transfer OK
ftp: 64d bajtów odebranych w 0.00Sekund 19000.00Kbajtów/sek.
ftp> rmdir test
250 Directory deleted successfully
ftp> status
Połączony z 10.0.0.4.
Typ: ascii; Pełna informacja: Wł. ; Sygnał dźwiękowy: Wł. ; Ostrzeżenie: Wł. ;
Globbing: Wł.
Debugowanie: Wł. ; Drukowanie znaku hash: Wł. .
ftp> quit
221 Goodbye
```

Wniosek:

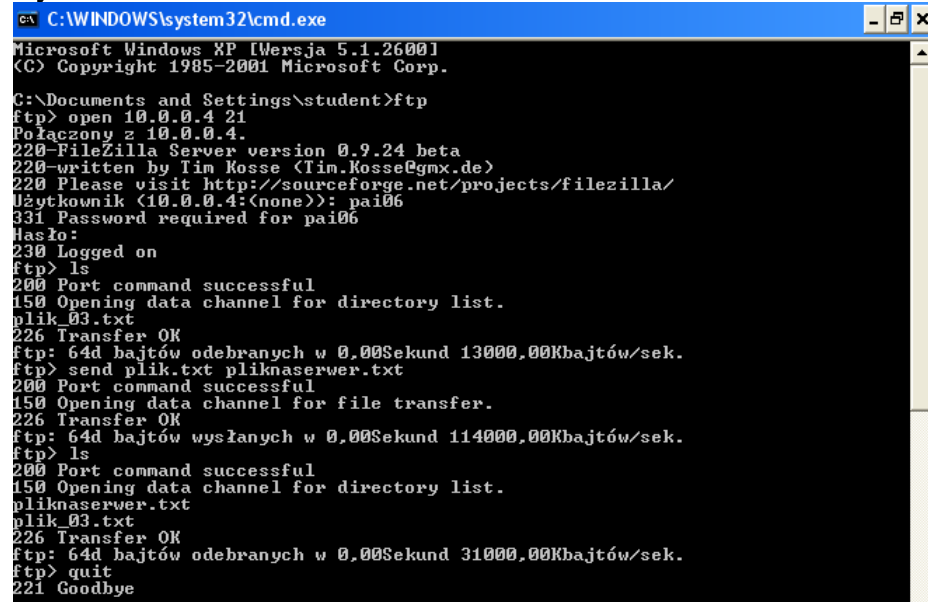
Po połączeniu się z 10.0.0.4 na porcie ftp 21 wpisaliśmy login oraz hasło. Następnie wpisując polecenie help sprawdziliśmy podstawowe dostępne komendy dla protokołu ftp. Następnie sprawdziliśmy działanie komendy ls (zwraca zawartość bieżącego katalogu), mkdir (zakładanie nowego katalogu), cd (przejdź do danej lokalizacji), rmdir (usuwanie katalogu), help (wyświetlanie pomocy). Po zakończeniu wychodzimy poleceniem quit.

b) Przesłać plik na serwer (PLIK TEKSTOWY Z DANYMI ZESPOŁU).

Wpisywane komendy:

```
open 10.0.0.4 21
pai06
pai06
ls
send plik.txt pliknaserwer.txt
ls
quit
```

Wynik zadziałania:



```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows XP [Wersja 5.1.26001
(C) Copyright 1985-2001 Microsoft Corp.

C:\Documents and Settings\student>ftp
ftp> open 10.0.0.4 21
Połączony z 10.0.0.4.
220-FileZilla Server version 0.9.24 beta
220-written by Tim Kosse (tim.kosse@gmx.de)
220 Please visit http://sourceforge.net/projects/filezilla/
Użytkownik (10.0.0.4:(none)): pai06
331 Password required for pai06
Hasło:
230 Logged on
ftp> ls
200 Port command successful
150 Opening data channel for directory list.
plik_03.txt
226 Transfer OK
ftp: 64d bajtów odebranych w 0.00Sekund 13000.00Kbajtów/sek.
ftp> send plik.txt pliknaserwer.txt
200 Port command successful
150 Opening data channel for file transfer.
226 Transfer OK
ftp: 64d bajtów wysłanych w 0.00Sekund 114000.00Kbajtów/sek.
ftp> ls
200 Port command successful
150 Opening data channel for directory list.
pliknaserwer.txt
plik_03.txt
226 Transfer OK
ftp: 64d bajtów odebranych w 0.00Sekund 31000.00Kbajtów/sek.
ftp> quit
221 Goodbye
```

Wniosek:

Po zalogowaniu się do serwera ftp wpisaliśmy polecenie ls sprawdzając, jakie pliki znajdują się na nim. Następnie wcześniej przygotowany plik tekstowy z danymi zespołu o nazwie plik.txt wysłaliśmy na serwer pod nazwą pliknaserwer.txt poleceniem send. Następnie sprawdziliśmy poleceniem ls, czy plik znajduje się już na ftp. Polecenie zwróciło, iż pliknaserwer.txt jest na serwerze.

c) Pobrać plik z serwera.

Wpisywane komendy:

```
open 10.0.0.4 21
pai06
pai06
ls
recv pliknaserwer.txt pliksciagniety.txt
quit
```

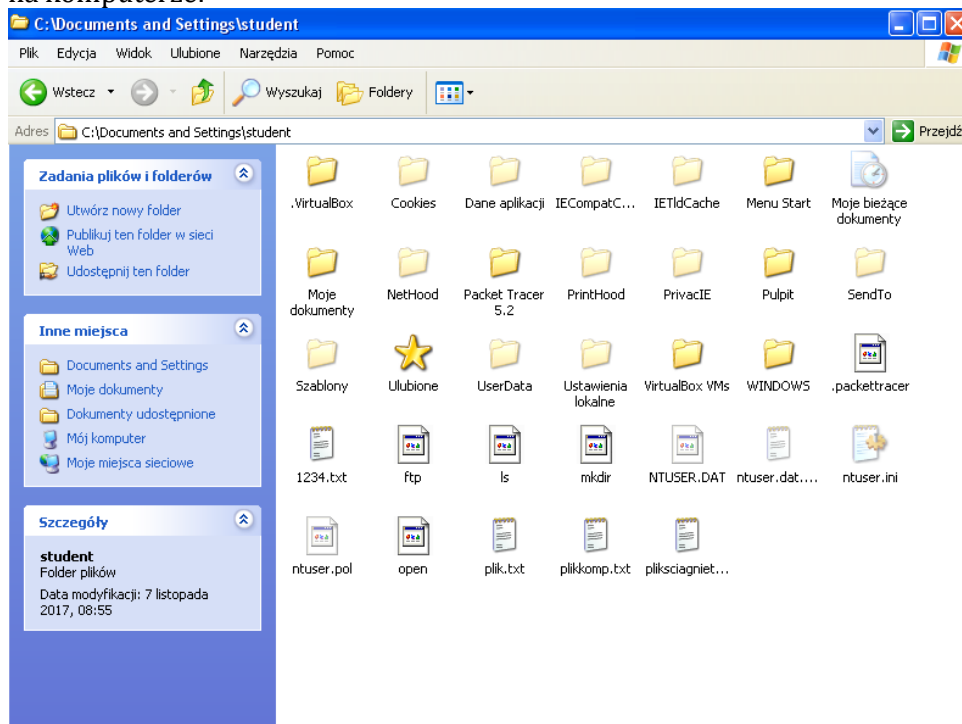
Wynik zadziałania:

```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows XP [Wersja 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\Documents and Settings\student>ftp
ftp> open 10.0.0.4 21
Połączony z 10.0.0.4.
220-FileZilla Server version 0.9.24 beta
220-written by Tim Kosse (Tim.Kosse@gmx.de)
220 Please visit http://sourceforge.net/projects/filezilla/
Użytkownik (10.0.0.4:(none)): pai06
331 Password required for pai06
Hasło:
230 Logged on
ftp> ls
200 Port command successful
150 Opening data channel for directory list.
pliknaserwer.txt
plik_03.txt
226 Transfer OK
ftp> 64d bajtów odebranych w 0,00Sekund 31000,00Kbajtów/sek.
ftp> recv pliknaserwer.txt pliksciagniety.txt
200 Port command successful
150 Opening data channel for file transfer.
226 Transfer OK
ftp> 64d bajtów odebranych w 0,00Sekund 114000,00Kbajtów/sek.
ftp> quit
221 Goodbye
```

Wniosek:

Po zalogowaniu się do serwera ftp wpisaliśmy polecenie ls sprawdzając, jakie pliki znajdują się na nim. Następnie poleceniem recv ściągnęliśmy pliknaserwer.txt na komputer pod nazwą pliksciagniety.txt. Serwer zwrócił informację, iż pobieranie powiodło się. Plik faktycznie znajduje się na komputerze:

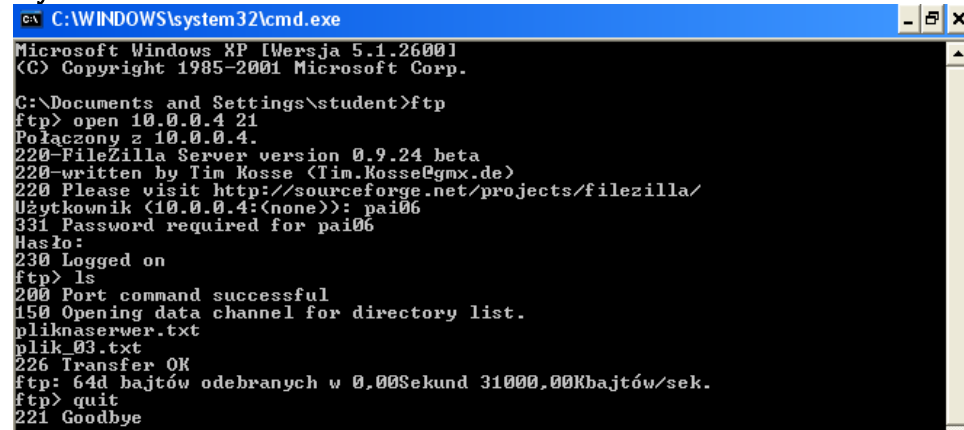


d) Pobrać listę plików dostępnych na serwerze.

Wpisywane komendy:

```
open 10.0.0.4 21
pai06
pai06
ls
quit
```

Wynik zadziałania:



```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows XP [Wersja 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\Documents and Settings\student>ftp
ftp> open 10.0.0.4 21
Połączony z 10.0.0.4.
220-FileZilla Server version 0.9.24 beta
220-written by Tim Kosse (Tim.Kosse@gmx.de)
220 Please visit http://sourceforge.net/projects/filezilla/
Użytkownik (10.0.0.4:(none)): pai06
331 Password required for pai06
Hasło:
230 Logged on
ftp> ls
200 Port command successful
150 Opening data channel for directory list.
pliknaserwer.txt
plik_03.txt
226 Transfer OK
ftp> quit
221 Goodbye
```

Wniosek:

Celem pobrania listy dostępnych plików na serwerze posłużyliśmy się poleceniem ls. Można również zastosować polecenie ls . plik.txt, gdzie lista plików zostanie zapisana do pliku tekstowego plik.txt celem np. archiwizacji, jakie pliki znajdowały się o danej porze na serwerze. Po wpisaniu ls i wyświetleniu plików serwer zwrócił informację o kodzie 226 „Transfer OK”.

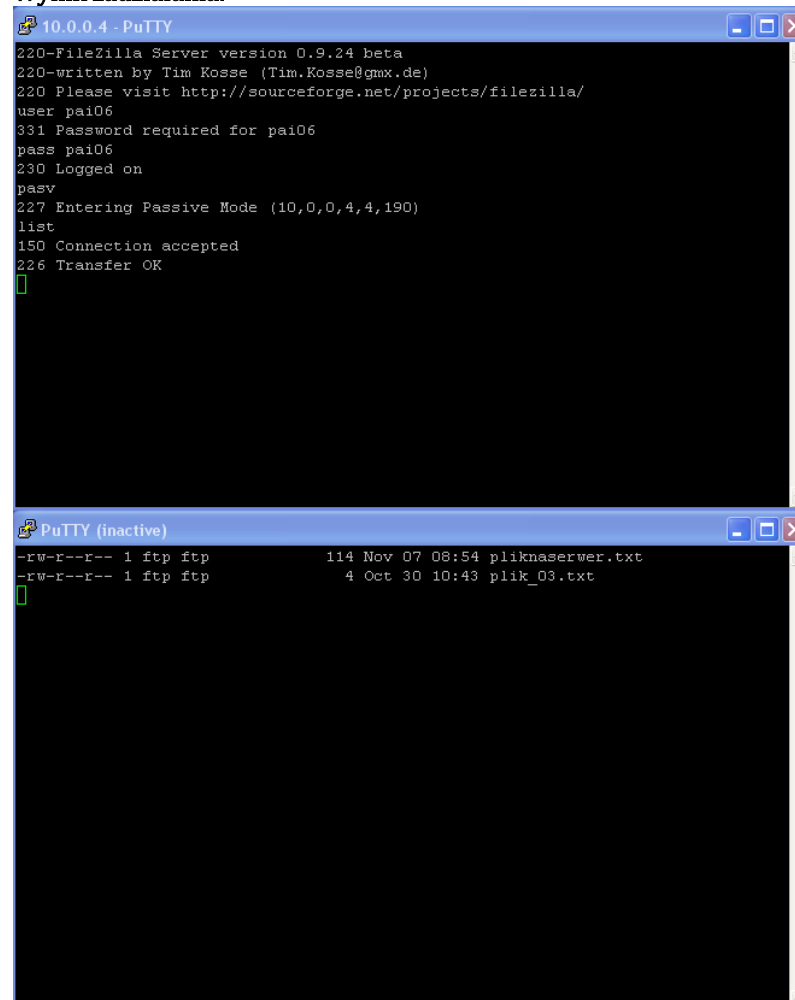
4. Połączyć się telnetem z serwerem ftp.

a) Pobrać listę plików.

Wpisywane komendy:

```
user pai06
pass pai06
pasv
list
```

Wynik zadziałania:



```
10.0.0.4 - PuTTY
220-FileZilla Server version 0.9.24 beta
220-written by Tim Kosse (Tim.Kosse@gmx.de)
220 Please visit http://sourceforge.net/projects/filezilla/
user pai06
331 Password required for pai06
pass pai06
230 Logged on
pasv
227 Entering Passive Mode (10,0,0,4,4,190)
list
150 Connection accepted
226 Transfer OK
█

PuTTY (inactive)
-rw-r--r-- 1 ftp ftp      114 Nov 07 08:54 pliknaserwer.txt
-rw-r--r-- 1 ftp ftp       4 Oct 30 10:43 plik_03.txt
█
```


Wniosek:

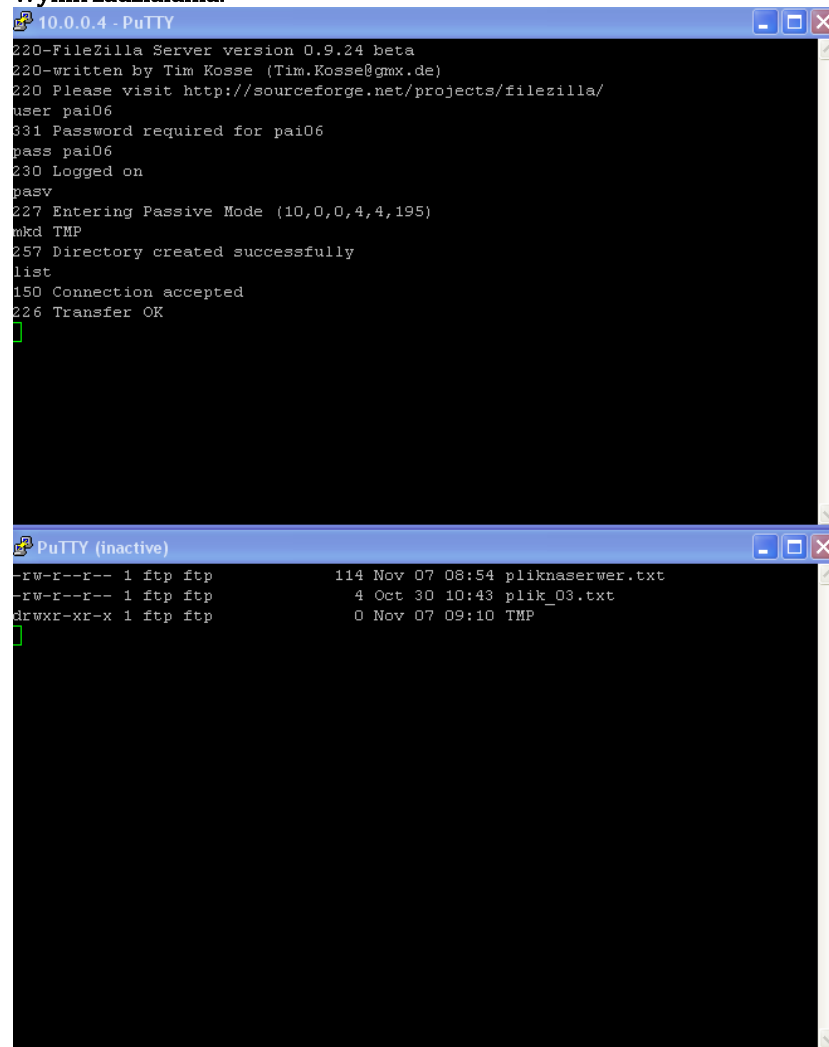
W programie Putty celem połączenia się telnetem do ftp trzeba było ustawić dla 10.0.0.4 port o numerze 21. Po wyświetleniu się konsoli należało zalogować się używając polecenia user oraz podać hasło poleceniem pass. Następnie wpisaliśmy polecenie pasv, aby serwer słuchał na porcie, który nie jest standardowym data portem i czekał na połączenie, a nie sam je inicjował. Serwer zwrócił nam wartość (10,0,0,4,190). Następnie należało zostawić w tle uruchomiony ftp na porcie 21 oraz uruchomić ponownie program Putty i połączyć się z 10.0.0.4 na porcie wyliczonym na podstawie danych zwróconych przez polecenie pasv. W naszym przypadku było to: $256*4+190 = 1214$. Zatem należało się połączyć na 10.0.0.4 na porcie 1214. Następnie na pierwszej konsoli wpisując polecenie list, wszystkie pliki na serwerze wraz z prawami dostępu i datą modyfikacji wyświetliły się na drugiej konsoli.

b) Utworzyć katalog TMP.

Wpisywane komendy:

```
user pai06
pass pai06
pasv
mkd TMP
list
```

Wynik zadziałania:



```
10.0.0.4 - PuTTY
220-FileZilla Server version 0.9.24 beta
220-written by Tim Kosse (Tim.Kosse@gmx.de)
220 Please visit http://sourceforge.net/projects/filezilla/
user pai06
331 Password required for pai06
pass pai06
230 Logged on
pasv
227 Entering Passive Mode (10,0,0,4,195)
mkd TMP
257 Directory created successfully
list
150 Connection accepted
226 Transfer OK

PuTTY (inactive)
-rw-r--r-- 1 ftp ftp      114 Nov 07 08:54 pliknaserwer.txt
-rw-r--r-- 1 ftp ftp         4 Oct 30 10:43 plik_03.txt
drwxr-xr-x 1 ftp ftp         0 Nov 07 09:10 TMP
```

Wniosek:

W programie Putty celem połączenia się telnetem do ftp trzeba było ustawić dla 10.0.0.4 port o numerze 21. Po wyświetleniu się konsoli należało zalogować się używając polecenia user oraz podać hasło poleceniem pass. Następnie wpisaliśmy polecenie pasv, aby serwer słuchał na porcie, który nie jest standardowym data portem i czekał na połączenie, a nie sam je inicjował. Serwer zwrócił nam wartość (10,0,0,4,195). Następnie należało zostawić w tle uruchomiony ftp na porcie 21 oraz uruchomić ponownie program Putty i połączyć się z 10.0.0.4 na porcie wyliczonym na podstawie danych zwróconych przez polecenie pasv. W naszym przypadku było to: $256*4+195 = 1219$. Zatem należało się połączyć na 10.0.0.4 na porcie 1219. Następnie na pierwszej konsoli

wpisując polecenie `mkd TMP` tworzy się katalog na serwerze ftp. Celem sprawdzenia wpisaliśmy polecenie `list` i katalog `TMP` pojawił się wraz z prawami dostępu (`d` na początku – `directory`) na drugiej konsoli.

c) Pobrać plik.

Wpisywane komendy:

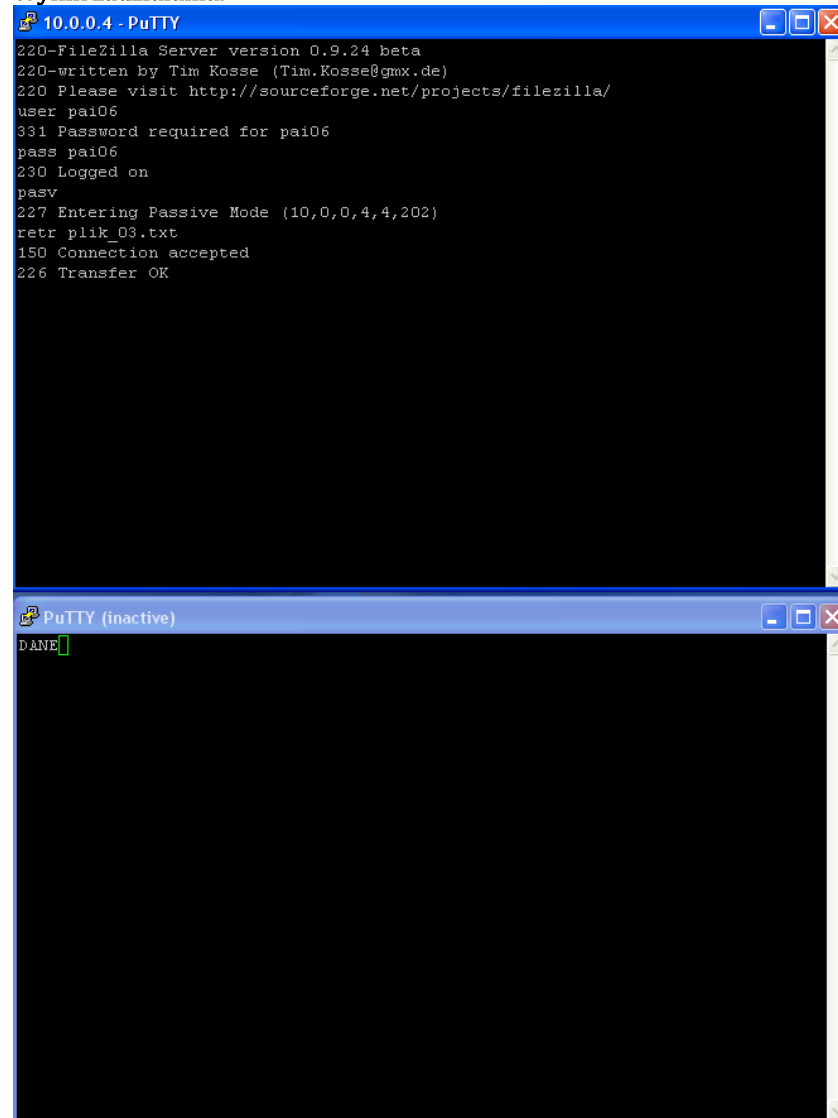
`user pai06`

`pass pai06`

`pasv`

`retr plik_03.txt`

Wynik zadziałania:



```
10.0.0.4 - PuTTY
220-FileZilla Server version 0.9.24 beta
220-written by Tim Kosse (Tim.Kosse@gmx.de)
220 Please visit http://sourceforge.net/projects/filezilla/
user pai06
331 Password required for pai06
pass pai06
230 Logged on
pasv
227 Entering Passive Mode (10,0,0,4,4,202)
retr plik_03.txt
150 Connection accepted
226 Transfer OK

PuTTY (inactive)
DANE
```

Wniosek:

W programie Putty celem połączenia się telnetem do ftp trzeba było ustawić dla 10.0.0.4 port o numerze 21. Po wyświetleniu się konsoli należało zalogować się używając polecenia `user` oraz podać hasło poleceniem `pass`. Następnie wpisaliśmy polecenie `pasv`, aby serwer słuchał na porcie, który nie jest standardowym data portem i czekał na połączenie, a nie sam je inicjował. Serwer zwrócił nam wartość (10,0,0,4,4,202). Następnie należało zostawić w tle uruchomiony ftp na porcie 21 oraz uruchomić ponownie program Putty i połączyć się z 10.0.0.4 na porcie wyliczonym na podstawie danych zwróconych przez polecenie `pasv`. W naszym przypadku było to: $256 \cdot 4 + 202 = 1226$. Zatem należało się połączyć na 10.0.0.4 na porcie 1226. Następnie na pierwszej konsoli wpisując polecenie `retr plik_03.txt` (dowolny plik znajdujący się na serwerze) otwiera się jego zawartość na drugiej konsoli. Na drugiej konsoli została wyświetlona jego zawartość, dość krótka, bo tylko o treści „DANE”.