

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ  
VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

Počítačové komunikácie a siete  
Projekt č. 1  
Varianta 1

# Obsah

<b>1</b>	<b>Úvod</b>	<b>2</b>
1.1	Cieľ projektu . . . . .	2
<b>2</b>	<b>Popis protokolu IPK 1.0</b>	<b>2</b>
<b>3</b>	<b>Programová realizácia</b>	<b>3</b>
3.1	Kontrola parametrov . . . . .	3
3.2	Server . . . . .	3
3.3	Klient . . . . .	3
<b>4</b>	<b>Demonštrácia funkčnosti</b>	<b>4</b>
<b>5</b>	<b>Použité zdroje</b>	<b>5</b>

# 1 Úvod

Tento dokument reprezentuje dokumentáciu k projektu č. 1 z predmetu Počítačové komunikácie a siete (IPK). Vytvorené programy v rámci projektu sú implementované v jazyku C++<sup>1</sup>. Riešená bola varianta č. 1 - Získavanie informácií o užívateľovi. Navrhnutý protokol sa nazýva IPK 1.0.

## 1.1 Cieľ projektu

Cieľom projektu bolo vytvoriť funkčnú komunikáciu typu Klient-Server medzi aplikáciami **ipk-client** a **ipk-server**, kde klient vyžiada od serveru na základe vhodných prepínačov informáciu o užívateľoch. Server túto informáciu získa zo súboru `/etc/passwd`.

Typy požiadavkov zo strany klienta	prepínač
Zistenie mena a bližších informácií o užívateľovi	-n login
Zistenie domovského adresára užívateľa	-f login
Výpis loginov všetkých užívateľov v <code>/etc/passwd</code>	-l
Výpis loginov užívateľov začínajúcich zadaným prefixom	-l prefix

Výsledný tvar spustenia jednotlivých programov potom vyzerá nasledovne

<b>Klient</b>	<code>./ipk-client -h host -p port [-nl-fl-l] login</code>
<b>Server</b>	<code>./ipk-server -p port</code>

## 2 Popis protokolu IPK 1.0

Navrhnutý protokol IPK 1.0 vychádza z protokolu TCP. Po navedení spojenia medzi klientskou aplikáciou a serverom klient pošle serveru login a prepínač pomocou príkazov `LOGIN login` a `OPTION opt`, na základe ktorého bude server ďalej pracovať. V prípade prepínaču `-l`, ak nebol login zadaný, posiela sa správa `LOGIN no_login`. Ak boli obe dáta správne doručené a server ich doručenie potvrdil, pošle klient serveru žiadosť o dáta príkazom `SEND_INFO`. Server následne pošle klientovi dáta. V prípade, že zadaný login je neplatný alebo sa informácie o užívateľovi s súborom nenachádzajú, server pošle klientovi správu `NONE`. V takom prípade klient nevypisuje na výstup nič. Ak sa jedná o viac informácií (viz. prepínač `-l`), server ukončí po odoslaní všetkých dát komunikáciu pomocou príkazu `END`.

**Ukážka komunikácie:**

Odosielateľ	Tvar správy	Prijímateľ
Klient	<code>LOGIN login</code>	Server
Server	<code>LOGIN_OK</code>	Klient
Klient	<code>OPTION option</code>	Server
Server	<code>OPTION_OK</code>	Klient
Klient	<code>SEND_INFO</code>	Server
Server	<code>DATA</code>	Klient
Klient	<code>SEND_INFO</code>	Server
Server	<code>END</code>	Klient

Klient nemá možnosť ukončiť server, ukončenie serveru je vyvolané signálom `SIGKILL`. V prípade prepínačov `-n` a `-f` server neposiela signál `END`, pretože sa posiela iba jedna správa obsahujúca dáta. Po potvrdení dát sa spojenie aj klient ukončia.

<sup>1</sup>Norma C++11, viz. <https://isocpp.org/wiki/faq/cpp11>

### 3 Programová realizácia

Klient aj server boli naprogramované v jazyku C++ objektovo orientovaným prístupom. Implementácia vychádza z dema TCP komunikácie v archíve Demo\_C.zip<sup>2</sup>

#### 3.1 Kontrola parametrov

Kontrola parametrov prebieha pomocou funkcie `getopt`. Nie je povolené opakované zadanie parametru, čo zabezpečuje premenná typu `bool` (flag) pre každý parameter, ktorá značí, či bol už parameter zadáný. Rovnakým spôsobom sa rieši nepovolená kombinácia parametrov `-n`, `-f` a `-l`. Či bola daná hodnota pri určitom parametre sa zistí pomocou premennej `optarg` vychádzajúca z funkcie `getopt`. Zadanie loginu prípadne príliš mnoho parametrov sa vypočíta vzt'ahom `argc-optind`, kde `argc` je celkový počet zadáných parametrov a `optind` je počet nespracovaných parametrov po spracovaní funkciou `getopt`. Pri parametri `-p` kontroluje validita prevodu na číslo funkciou `strtol`.

#### 3.2 Server

Server je reprezentovaný triedou `Server` v súbore `ipk_proj1_server.cpp`.

Atribúty			Funkcie		
typ	meno	popis	typ	meno	popis
int	port	port	string	find_user	vyhl'adá užívateľa v súbore
int	server_socket	socket serveru	vector	find_users	vyhl'adá užívateľov v súbore
sockaddr_in	server_address	adresa serveru	void	find_substr	extrahuje atribút užívateľa
sockaddr_in	client_address	adresa klienta	void	establish_connection	zabezpečí celú komunikáciu

Funkcia `establish_connection` na strane serveru funguje interne **konečným automatom**, ktorý zabezpečuje správne poradie správ. V cykle sa prijímajú správy od klienta a stav sa mení po úspešnom dokončení predchádzajúceho stavu. Stav 0 reprezentuje načítanie loginu, stav 1 načítanie parametru a stav 2 opakované posielanie dát.

Aby sa zabezpečila schopnosť serveru obsluhovať viacerých klientov súčasne (**paralelne**) a nemuseli čakať v rade na obsluhu, pre každého pripojeného klienta je na strane serveru vytvorený **podproces**, ktorý ho obsluhuje a hlavný proces tak dostáva možnosť obsluhovať nových klientov.

#### 3.3 Klient

Klient je reprezentovaný triedou `Klient` v súbore `ipk_proj1_client.cpp`.

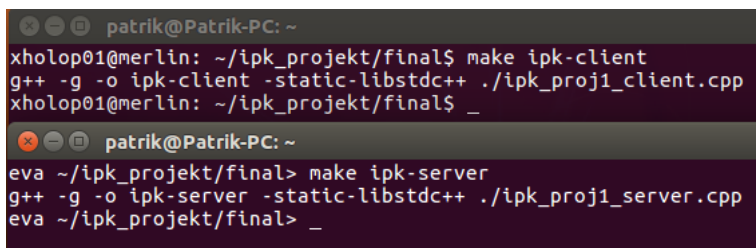
Atribúty			Funkcie		
typ	meno	popis	typ	meno	popis
int	port	port	void	send_login	pošle login a prijme potvrdenie
string	server_name	adresa/dns serveru	void	send_option	pošle option a prijme potvrdenie
string	login	login / no_login	void	request_data	pošle SEND_INFO a prijme dáta
char	opt	prepínač -l, -n, -f	void	establish_connection	zabezpečí celú komunikáciu
int	client_socket	klientsky socket			
sockaddr_in	server_adress	adresa servera			

V prípade, že klient od serveru dostane dátovú správu v tvare `NONE`, znamená to, že pri prepínačoch `-f` a `-n` nebol užívateľ nájdený. V takom prípade je na `stderr` vypísaný chybný login.

<sup>2</sup>Dostupné z [https://wis.fit.vutbr.cz/FIT/st/course-files-st.php?file=%2Fcourse%2FIPK-IT%2Fother%2FDemo\\_C.zip&cid=11963](https://wis.fit.vutbr.cz/FIT/st/course-files-st.php?file=%2Fcourse%2FIPK-IT%2Fother%2FDemo_C.zip&cid=11963)

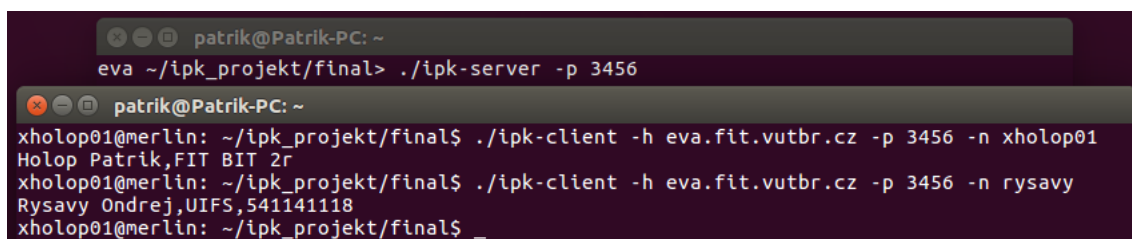
## 4 Demonštrácia funkčnosti

Funkčnosť celej komunikácie budeme demonštrovať na komunikácii medzi servermi Eva a Merlin, kde Eva bude v roli serveru a Merlin klienta. Ako login bude použitý môj vlastný, xholop01, a referenčný zo zadania rysavy.



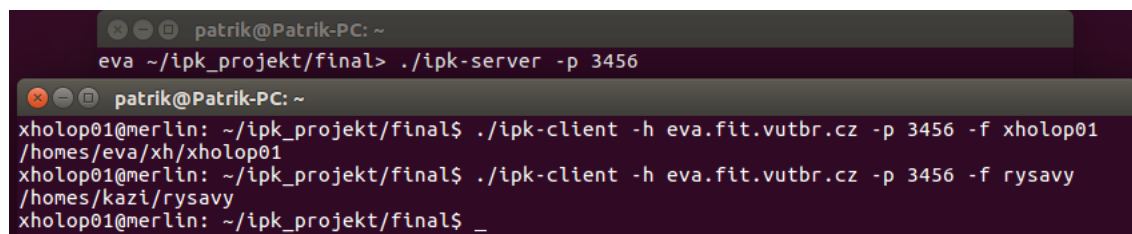
```
patrik@Patrik-PC: ~  
xholop01@merlin: ~/ipk_projekt/final$ make ipk-client  
g++ -g -o ipk-client -static-libstdc++ ./ipk_proj1_client.cpp  
xholop01@merlin: ~/ipk_projekt/final$ _  
patrik@Patrik-PC: ~  
eva ~/ipk_projekt/final> make ipk-server  
g++ -g -o ipk-server -static-libstdc++ ./ipk_proj1_server.cpp  
eva ~/ipk_projekt/final> _
```

Obrázek 1: Ukážka prekladu



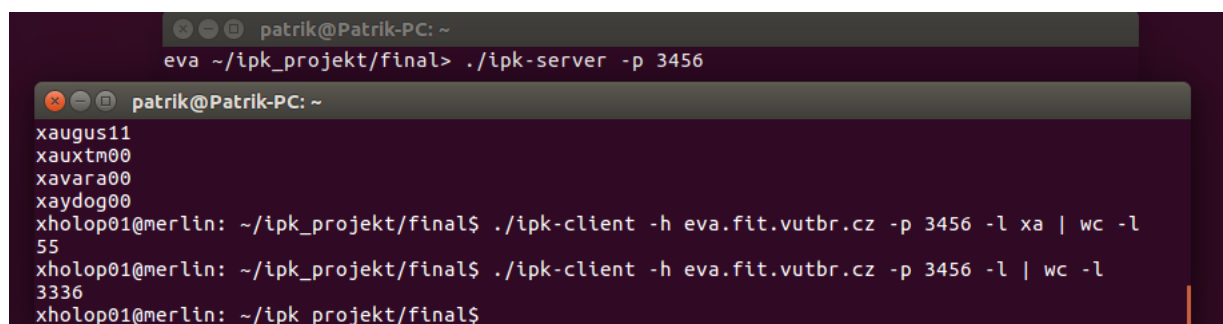
```
patrik@Patrik-PC: ~  
eva ~/ipk_projekt/final> ./ipk-server -p 3456  
patrik@Patrik-PC: ~  
xholop01@merlin: ~/ipk_projekt/final$ ./ipk-client -h eva.fit.vutbr.cz -p 3456 -n xholop01  
Holop Patrik,FIT BIT 2r  
xholop01@merlin: ~/ipk_projekt/final$ ./ipk-client -h eva.fit.vutbr.cz -p 3456 -n rysavy  
Rysavy Ondrej,UIFS,54114118  
xholop01@merlin: ~/ipk_projekt/final$ _
```

Obrázek 2: Ukážka parametru -n



```
patrik@Patrik-PC: ~  
eva ~/ipk_projekt/final> ./ipk-server -p 3456  
patrik@Patrik-PC: ~  
xholop01@merlin: ~/ipk_projekt/final$ ./ipk-client -h eva.fit.vutbr.cz -p 3456 -f xholop01  
/homes/eva/xh/xholop01  
xholop01@merlin: ~/ipk_projekt/final$ ./ipk-client -h eva.fit.vutbr.cz -p 3456 -f rysavy  
/homes/kazi/rysavy  
xholop01@merlin: ~/ipk_projekt/final$ _
```

Obrázek 3: Ukážka parametru -f



```
patrik@Patrik-PC: ~  
eva ~/ipk_projekt/final> ./ipk-server -p 3456  
patrik@Patrik-PC: ~  
xaugus11  
xauxt000  
xavara00  
xaydog00  
xholop01@merlin: ~/ipk_projekt/final$ ./ipk-client -h eva.fit.vutbr.cz -p 3456 -l xa | wc -l  
55  
xholop01@merlin: ~/ipk_projekt/final$ ./ipk-client -h eva.fit.vutbr.cz -p 3456 -l | wc -l  
3336  
xholop01@merlin: ~/ipk_projekt/final$ _
```

Obrázek 4: Ukážka parametru -l (navrchu sú posledné 4 loginy pri zadanom prefixe 'xa')

## 5 Použité zdroje

DONAHOO M., CALVERT K., 2001. *TCP/IP Sockets in C - Practical guide for programmers*. San Francisco: Morgan Kaufmann Publishers. ISBN: 978-1-55860-826-9

*The /etc/passwd File*, 2006 [online]. The Linux Information Project. Dostupné z: [http://www.linfo.org/etc\\_passwd.html](http://www.linfo.org/etc_passwd.html)

RYŠAVÝ O., RÁB J., *IPK - BSD schránky - 3. přednáška*, 2017 [online]. Dostupné z: <https://wis.fit.vutbr.cz/FIT/st/course-files-st.php?file=%2Fcourse%2FIPK-IT%2Flectures...>