

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ  
FAKULTA INFORMAČNÍCH TECHNOLOGIÍ

Klient-server pro získání  
informace o uživateli

IPK - Počítačové komunikace a sítě

## Obsah

<b>1</b>	<b>Zadání</b>	<b>3</b>
<b>2</b>	<b>Návrh protokolu</b>	<b>4</b>
<b>3</b>	<b>Implementace protokolu</b>	<b>5</b>
3.1	Popis práce . . . . .	5
3.2	Převod dat . . . . .	5
<b>4</b>	<b>Implementace projektu</b>	<b>6</b>
4.1	Prostředky implementace . . . . .	6
<b>5</b>	<b>Funkčnost</b>	<b>6</b>

# 1 Zadání

Úkolem bylo vytvořit komunikaci typu server-klient, s návrhem vlastního aplikačního protokolu. Pro řešení projektu byl použit protokol TCP/IP.

Projekt obsahuje dva spustitelné soubory *ipk-server* a *ipk-client* s následující konvecí volání:

*./ipk-client -h host -p port [-n -f -l] login*

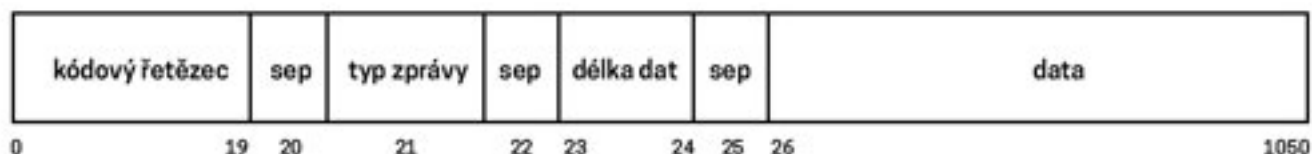
- host (IP adresa nebo fully-qualified DNS name) identifikace serveru jakožto koncového bodu komunikace klienta;
- port (číslo) cílové číslo portu;
- n značí, že bude vráceno plné jméno uživatele včetně případných dalších informací pro uvedený login (User ID Info);
- -f značí, že bude vrácena informace o domácím adresáři uživatele pro uvedený login (Home directory);
- -l značí, že bude vrácen seznam všech uživatelů, tento bude vypsán tak, že každé uživatelské jméno bude na zvláštním řádku; v tomto případě je login nepovinný. Je-li však uveden bude použit jako prefix pro výběr uživatelů.
- login určuje přihlašovací jméno uživatele pro výše uvedené operace.

*./ipk-client -h host -p port [-n -f -l] login*

- *./ipk-server -p port*

## 2 Návrh protokolu

Po vyslechnutí přednášek a samostudiu v tištěné i elektronické formě jsem dospěl k této formě protokolu. Samotný protokol je reprezentován jako pole znaků typů *char*, které má fixní velikost, a to *1050B*. Níže je uvedena schématická reprezentace protokolu:



pozn. uvedené hodnoty v grafu značí počet bajtů určených pro jednotlivé části protokolu.

- **kódový řetězec** *ipk18-proj1\_xmitas02* slouží k identifikaci připojeného klienta k našemu serveru.
- **sep** Oddělovač mezi užitnou informací (ASCII hodnota 31)
- **typ zprávy** hodnota (1 — 2 — 3) pro přepínače klienta (jméno, adresář, list) a (4 — 5 — 6) pro server, kde 4 je správná odpověď, 5 je též s správně, ale s pokračováním a 6 je obecná chyba. Tato hodnota je reprezentována datovým typem *uint8<sub>t</sub>*, tedy je přímo převeditelná na *char*.
- **sep** Oddělovač mezi užitnou informací (ASCII hodnota 31)
- **délka dat** označuje délku dat ve zprávě. Jedná se o hodnotu typu *uint16<sub>t</sub>*, jelikož užitný blok má až *1024B*, tedy tuto hodnotu není možno reprezentovat na 8b, je tedy nutno použít větší číslo. Jak je uloženo do pole *char* je vysvětleno níže.
- **sep** Oddělovač mezi užitnou informací (ASCII hodnota 30)
- **data** samotný balík dat, maximální délka *1024B*.

### 3 Implementace protokolu

Aplikační protokol je implementován ve vlastním souboru *ipk-protocol.c* a jeho hlavičkovém souboru. Jádro implementace se točí vedle dvou funkcí:

```
void ipk18_protocol_encode(  
char *buffer ,  
uint8_t type ,  
uint16_t data_len ,  
char *data  
);  
  
char * ipk18_protocol_decode(  
char *buffer  
);
```

#### 3.1 Popis práce

První jmenovaná poskytuje základní funkčnost serializace. Jelikož je v programu poněkud nevhodné pracovat s jednou **homogenní strukturou**, která obsahuje větší počet odlišných informací a naopak je nevhodné mít v protokolu heterogenní strukturu, vznikla potřeba převádět mezi těmito formáty. V předchozí sekci je popsáno informační rozdělení protokolu, které je již **serializované**. Proto byl implementován pár funkcí, které jsou navzájem symetrické a realizují (de)serializaci dat.

#### 3.2 Převod dat

*ipk18\_protocol\_encode()* slouží k uchování všech potřebných dat do protokolu, jednotlivé parametry jsou popsány v výše. Všimněme si hlavně parametru *data\_len*, který je realizován datovým typem *uint16\_t*, proto je zapotřebí speciální funkce, která ho převede na dva 8b znaky.

```
int ipk18_protocol_sep(char *buffer , int type , int i) {  
    buffer[i+1] = type;  
    return i+2;  
}
```

Protokol je tedy nezávislý na samotném programu a jeho implementaci. Je možno ho použít i v jiném kontextu. Dále by stálo za to považovat nad řešením zabezpečení (ačkoliv toto není náplní tohoto projektu).

## 4 Implementace projektu

Jelikož byl k implementaci projektu použit programovací jazyk *C*, bylo nutno přistoupit k určitým kompromisům co se týče programátorova pohodlí, nicméně troufám si tvrdit, že výsledný projekt splňuje body dostatečně. Pro určitá implementační omezení je v archívu přiložen soubor *Readme*.

### 4.1 Prostředky implementace

Řešení projektů leží v použití neblokující verze BSD socketů. Jejich použití je vidno v *ipk-client.c*

```
socket ();  
bind ();  
connect ();  
write ();  
  
while (1) {  
    read ();  
}
```

Základní princip je vidět výše v pseudokódu. Vytvoříme socket, připevníme k němu adresu, připojíme se na adresu serveru, ozveme se (tedy nás server registruje a umožňuje nám spolupráce, pokud splníme před dané podmínky) a po stvrzení dostaneme námi požadovaná data.

## 5 Funkčnost

Funkčnost byla otestována na mém osobním stroji (macOS), a také na serverech **merlin.fit.vutbr.cz** a **eva.fit.vutbr.cz**. Projekt byl napsán v souladu se standardem *C99* a přeložen lokální verzí *GCC7.1*. Projekt je zatížen určitými nesrovnalostmi, jak je zmíněno v *Readme*.

## Reference

- AUTHORS, L. *Linux manual pages* [online]. 2018. [cit. 12. 3. 2018]. Dostupné z: <http://man7.org/linux/man-pages/index.html>.
- FROM WIKIPEDIA, t. f. e. *Berkeley sockets* [online]. 2018. [cit. 12. 3. 2018]. Dostupné z: [https://en.wikipedia.org/wiki/Berkeley\\_sockets](https://en.wikipedia.org/wiki/Berkeley_sockets).
- GRYGAREK, P. *Softwarova rozhrani systemu UNIX pro pristup k sitovym sluzbam* [online]. [cit. 12. 3. 2018]. Dostupné z: <http://www.cs.vsb.cz/grygarek/LAN/sockets.html>.
- NIXCRAFT. *Understanding /etc/passwd File Format* [online]. 2017. [cit. 12. 3. 2018]. Dostupné z: <https://www.cyberciti.biz/faq/understanding-etcpasswd-file-format/>.
- RYSAVY, O. – RAB, J. *IPK - BSD schranky - 3. prednaska* [online]. 2018. [cit. 12. 3. 2018]. Dostupné z: <https://wis.fit.vutbr.cz/FIT/st/course-files-st.php.cs?file=%2Fcourse%2FIPK-IT%2Flectures%2FIPK2017L-02-APLIKACE.pdf&cid=11963>.