Slovenská technická univerzita

Fakulta infromatiky a infromačných technológií

Ilkovičova 3, 842 19 Bratislava 4

Zadanie č. 1 analyzátor sieťovej komunikácie

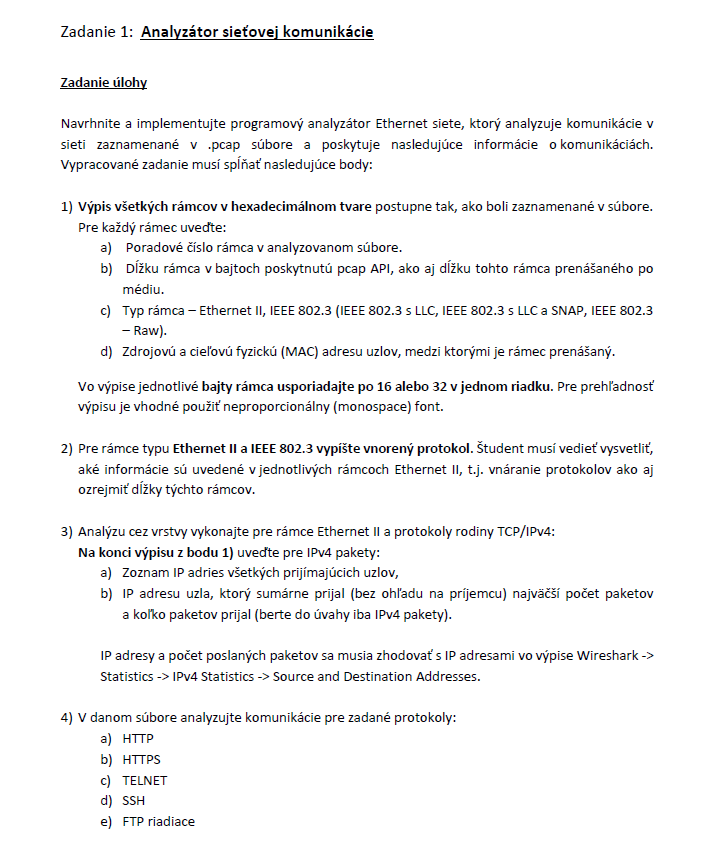
Autor: Michal Greguš

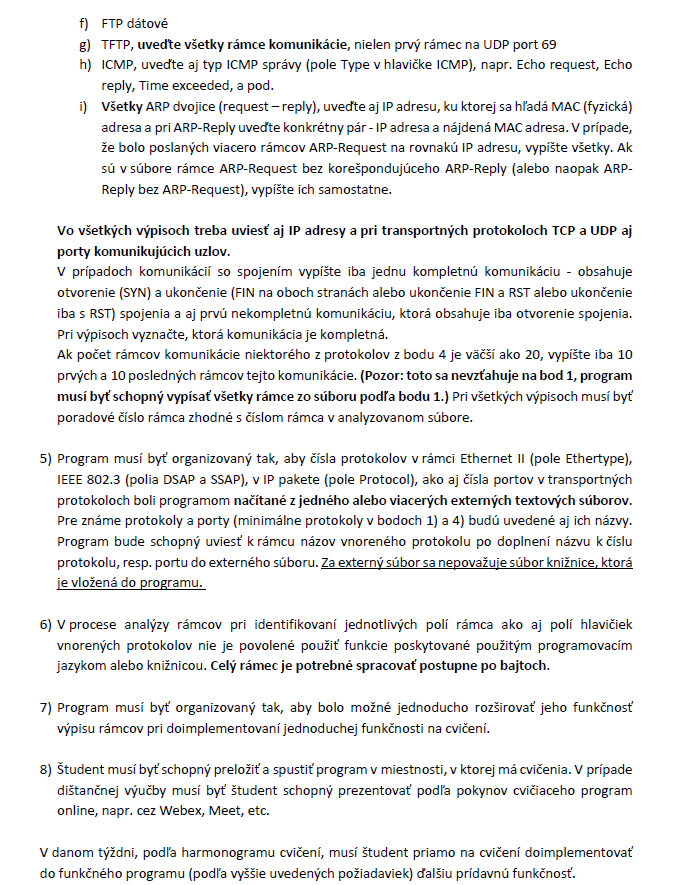
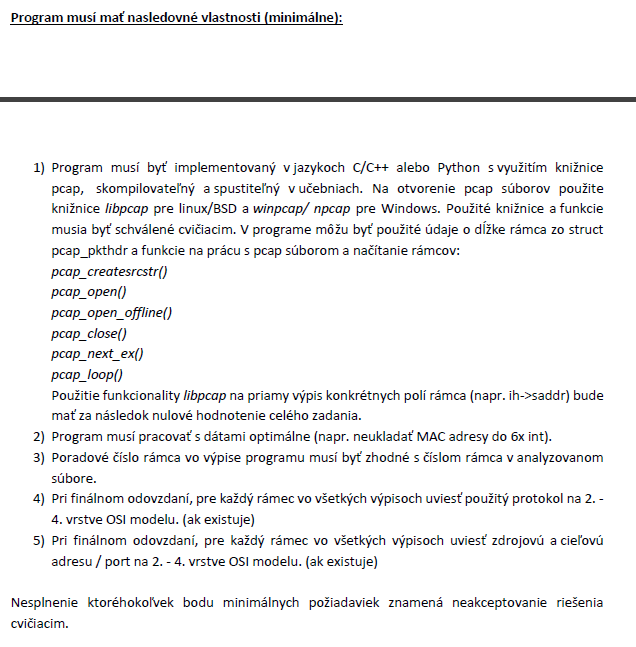
Predmet: Počítačové Komunikačné siete

Cvičiaci: Ing. Rastislav Bencel, PhD.

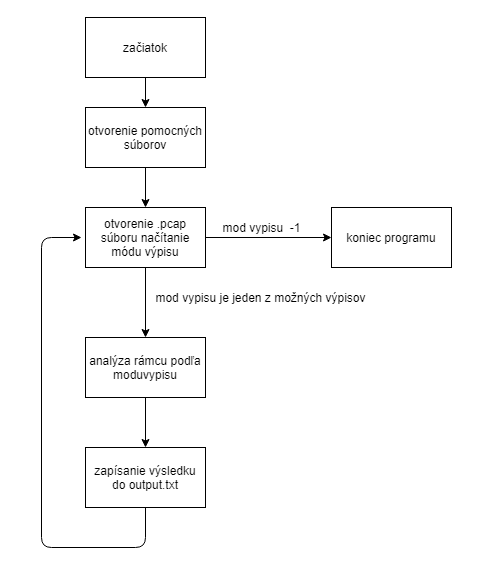
Cvičenie: Štvrtok 8:00

# Zadanie úlohy

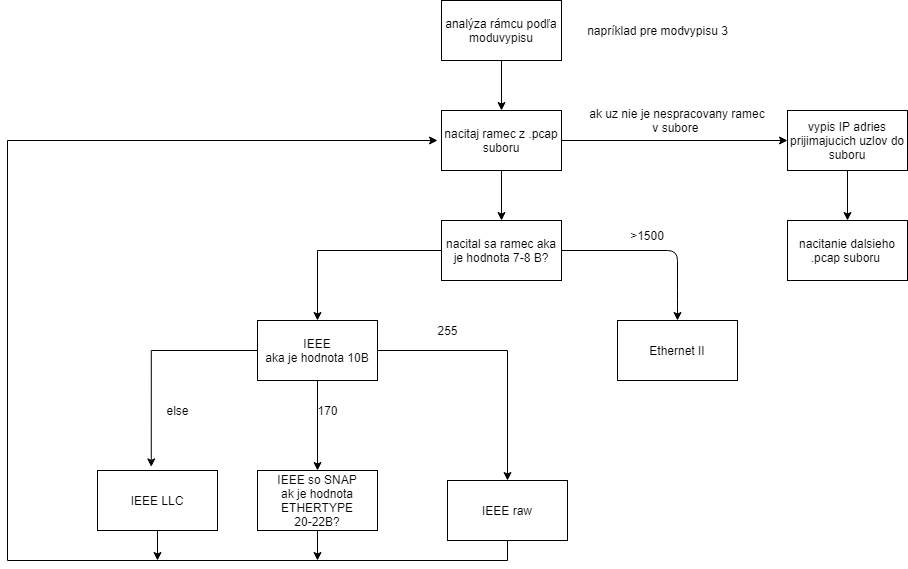


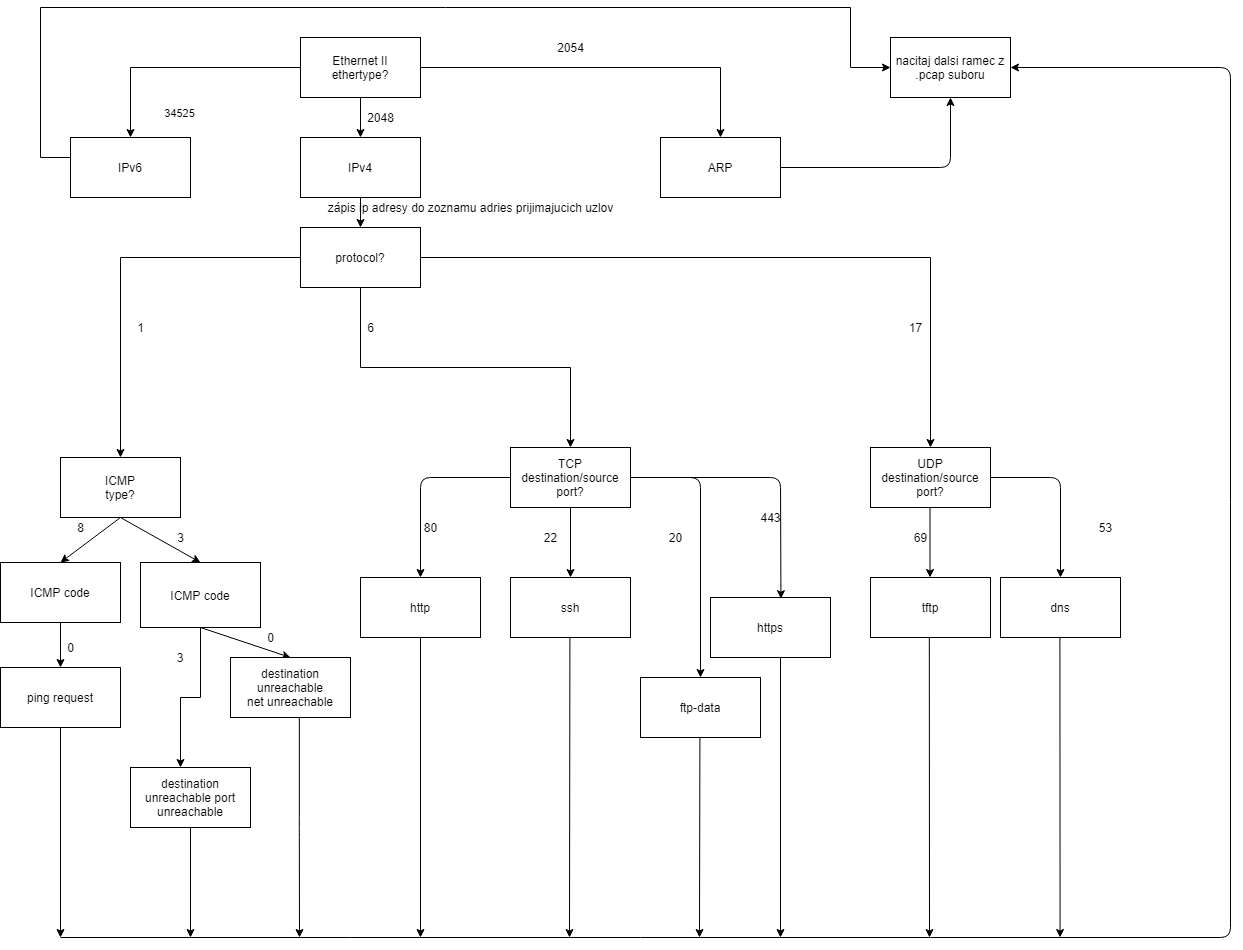
# Blokový návrh programu a koncepcia fungovanie riešenia



Rozhodovanie o type na 2 vrstve



Analýza ethernet rámca cez 3-4 vrstvu



# Mechanizmus analyzovanie rámcov

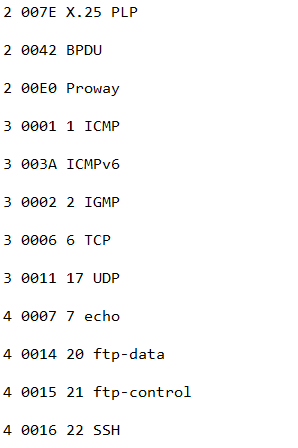
# Príklad štruktúry externých súborov, pre určenie protokolov a portov

Štruktúra súboru messages.txt

# 

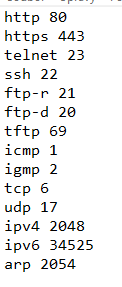
Príklad štruktúry súbory obsahujúceho icmp správy prvé čislo v riadku definuje ICMP type a druhé ICMP code nasledovaný samotným obsahom správy. Obe čísla sú v decimálnom tvare.

štruktúra súboru protokoly1.txt



Príklad štruktúry súboru pre čísla protokolov a portov kde prvé číslo(decimálne) určuje vrstvu v hlavičky protokolu v ktorej sa nachádza nasledujúce hexadecimálne číslo a to kód samotného protokolu nasledovaný názvom daného protokolu/portu. Priradí k číslu portu/protokolu názov.

štruktúra súboru cisla.txt



Príklad štruktúry súboru na priradenie čísla protokolu k názvu. Použité najmé pri modvypisi(4 - 9)

Prvý údaj je názov protokolu/portu nasledovaný decimálnym číslom daného portu.

# Opis používateľského rozhrania

Pred samotným spustením programu je potrebné vytvoriť 3 pomocné textové súbory. Prípadne skontrolovať správnosť ich názvu v programe.



Ďalej treba nastaviť do premennej filepath cestu k .pcap súborom, ktoré chceme analyzovať. Momentálne je nastavená tak, aby program hľadal názov .pcap súboru zo vstupu v priečinku pcap nachádzajúcom sa v aktuálnom adresári.



# 

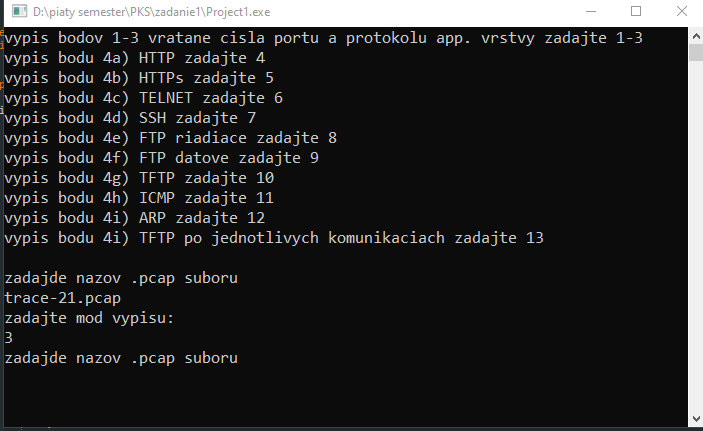
Po spustení programu možno zadať názov súboru, ktorí chceme analyzovať a následne aj jeho mód výpisu pre príslušný výpis ako vidieť na obrázku. Pre zvolenie módu 1,2,3 program vypíše vždy všetky požiadavky zo zadanie v bodoch 1 až 3. Teda vypíše všetky rámce analyzované cez 2-4 vrstvu a na konci sú uvedené ip adresy prijímajúcich uzlov. Následne pre jednotlivé konkrétne módy vypíše len konkrétnu komunikáciu.

Pri výpisoch 4-9 je najskôr vypísaná kompletná a následne nekompletná komunikácia.(Ak sa také nachádzajú v analyzovanom súbore)

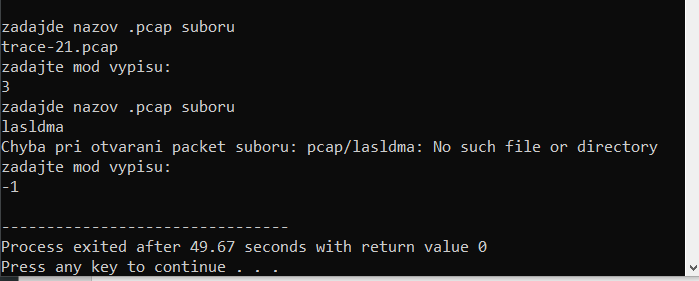
TFTP 10 vypíše len jednu TFTP komunikáciu

TFTP 13 vypíše všetky TFTP komunikácie po jednotlivých komunikáciach.

Pri ARP výpise je pre každý request reply uvedené do ktorej komunikácie patria.



Po zadaní módu výpisu program spracuje súbor a výpis zapíše do súboru output x(kde x je číslo súboru). Z každým ďalším načítaním a výpisom sa číslo x o jedna zväčší teda je zápis vykonaný vždy do ďaľšieho súboru.



Ak chceme program ukončiť možno zadať ľubovoľný názov súboru a následne ako mód výpisu zvoliť

-1. V prípade ak sme omylom zadali nesprávny názov súboru treba zadať nejaký mód výpisu(okrem -1). Program sa následne automaticky opäť dopytuje na názov súboru na spracovanie.

# Zvolené implementačné prostredie

Program som vytvoril v prostredí Dev-Cpp 5.11 s použitím 32-bit verzie TGM-GCC compileru. Predtým som si na windowse musel nainštalovať knižnicu pcap(automaticky spolu s wiresharkom). Program wireshark som používal pri kontrole správností výpisu môjho programu.

Následne som podľa návodu vytvoril projekt, s príslušnými parametrami, ktorí umožňuje využitie funkcionalít z knižnice pcap.h potrebných na otvorenie .pcap súborov. Program možno následne štandardne spustiť na (windows) zariadení ak ho otvoríme v Dev-Cpp prostrí ako Dev-C++ project.

návod: <https://www.csie.nuk.edu.tw/~wuch/course/csc521/lab/ex1-winpcap/>