



ZÁPADOČESKÁ
UNIVERZITA
V PLZNI

Semestrální práce z předmětu KIV/PD

Simulace počítačové sítě v simulátoru

Jméno a příjmení: Pavel Bžoch

Osobní číslo: A08N0061P

Obor: ININ/SWI

Email: pbzoch@students.zcu.cz

Obsah

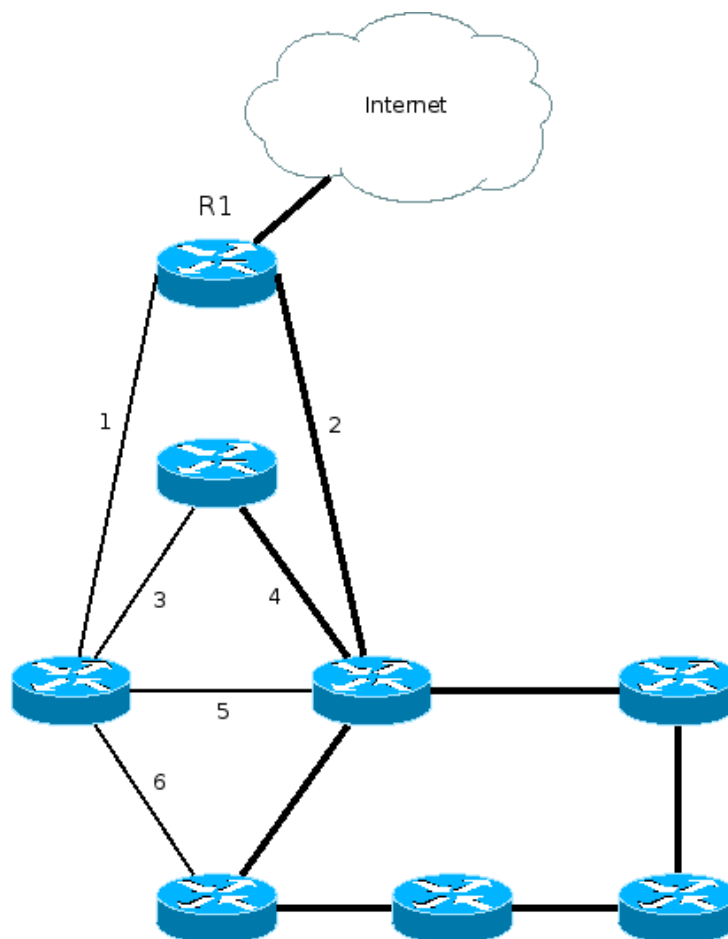
1. Zadání	3
2. Programátorská dokumentace	4
2.1 Výběr simulátoru	4
2.2 Realizace komunikace mezi uzly	4
2.3 Realizace komunikace z/do internetu	4
2.4 Realizace protokolu http	4
2.5 Realizace protokolu ssh	5
2.6 Realizace protokolu pop	5
2.7 Nalezené nedostatky simulátoru a jejich řešení.....	5
3. Uživatelská dokumentace	6
3.1 Spuštění simulátoru a otevření souboru s topologií.....	6
3.2 Editace topologie sítě.....	7
3.3 Simulace sítě.....	7
3.4 Přehrávání výsledků simulace	8
4. Výsledky měření zatížení linky 4	8
5. Závěr	9
6. Použité zdroje.....	10

1. Zadání

Následující topologii (obr. č.1) realizujte ve vybraném síťovém simulátoru. Routery na obrázku jsou propojeny linkami 1Gb/s (tlusté linky) včetně připojení do Internetu. Slabší linky značí rychlost 100Mb/s. Všude kromě routeru R1 je připojeno 24 stanic rychlostí 1Gb/s a 24 stanic rychlostí 100Mb/s.

Stanice mezi sebou komunikují (protokolem http každých 10 minut přenesou 10MB a protokolem ssh přenesou 10kB každé 2 minuty) a komunikují také směrem z/do Internetu (protokolem http přenesou 100MB z Internetu každé 3 minuty, protokolem pop přenesou 80kB z Internetu každou 1 minutu a protokolem ssh přenesou 10kB do Interneru každé 2 minuty).

Sledujte zatížení linky 4.



Obr. č.1 – topologie sítě

2. Programátorská dokumentace

V následujících podkapitolách bude popsán výběr simulátoru, realizace komunikace mezi jednotlivými uzly v rámci topologie, realizace komunikace pomocí protokolů http, ssh a pop. V poslední podkapitole budou shrnuty nedostatky simulátoru.

2.1 Výběr simulátoru

Při výběru simulátoru jsem se snažil najít takový simulátor, který by měl grafické uživatelské rozhraní a ve kterém by síť byla snadno realizovatelná. Dalším kritériem bylo, aby simulátor fungoval pod MS Windows. Jediným takovým simulátorem byl OMNet++. Tento simulátor ovšem špatně fungoval v koexistenci s MS .net Studiem 2008. Dalším simulátorem, který přicházel v úvahu, byl simulátor NCTuns. Tento simulátor ovšem ke svému běhu vyžaduje operační systém Linux, distribuci Fedora. Výrobce uvádí, že by simulátor měl fungovat i pod jinými distribucemi Linuxu, preferovaná a vyzkoušená je ale jen již uvedená Fedora. Při instalaci simulátoru bylo ještě potřeba doinstalovat balíček „rsh server“, který není součástí distribuce.

2.2 Realizace komunikace mezi uzly

Jak již bylo zmíněno výše, simulátor obsahuje grafické uživatelské rozhraní, ve kterém lze intuitivně nakreslit celou topologii sítě, nastavit komunikaci mezi jednotlivými počítači a v neposlední řadě i nastavit propustnost jednotlivých linek. Simulátor automaticky nastavuje IP adresy počítačům.

Realizace komunikace mezi dvěma uzly je dle mého zjištění rychlejší editací vytvořeného xml dokumentu s topologií sítě. Nejprve je potřeba realizovat komunikaci mezi dvěma uzly pomocí GUI a topologii uložit, protože je potřeba zjistit strukturu příkazu v tomto dokumentu. Pro ostatní uzly lze provést editaci zmíněného souboru. Komunikace mezi dvěma uzly se realizuje pomocí příkazů „step“ a „rtcp“ (předpokládám, že uzly budou komunikovat pomocí protokolu TCP/IP). Příkaz „step“ vyžaduje ještě IP adresu druhého zařízení, se kterým bude komunikovat. Dále je potřeba u každého tohoto příkazu definovat, jak dlouho bude aktivní, tedy jak dlouho bude vysílat (step) nebo jak dlouho má přijímat (rtcp).

Z výše uvedeného plyne zásadní věc – v simulátoru nelze nastavit objem přenesených dat mezi uzly, nastavuje se pouze čas komunikace. V tomto bodě jsem tedy nemohl přesně dodržet zadání. Tento „nedostatek“ simulátoru jsem se snažil vynahradit tím, že vysílám na dané lince vždy jen takový čas, za který by se takový objem dat přibližně přenesl.

2.3 Realizace komunikace z/do internetu

Simulátor neobsahuje žádnou komponentu, jejíž pomocí by bylo možné realizovat komunikaci z/do internetu. Pro simulování jsem zvolil nejjednodušší možnost – celý internet bude představovat jedna stanice připojená k routeru R1.

2.4 Realizace protokolu http

Protokol http slouží k přenosu dat (dokumentů) mezi klientem a serverem. V naší topologii klient i server může představovat jakýkoli počítač. Při běžné komunikaci vyše

nejprve klient „GET *adresa protokol* \n HOST“. Tato zpráva je v mé implementaci reprezentována krátkým vysláním (1/100s). Server odpovídá (zjednodušeně budu předpokládat, že vše probíhá v pořádku) zprávou ve tvaru: „*protokol* 200 OK \n co obsahuje dokument \n datum \n dokument“. Délka této zprávy může být různě dlouhá, nicméně ze zadání předpokládám, že má konstantní délku 10MB při komunikaci mezi uzly resp. 100MB při komunikaci z internetu. Odpověď serveru je v mé implementaci realizována vysláním po dobu 1s pro 10MB přenos po lince s rychlostí 100Mb/s resp. 10s pro 100MB po téže lince.

2.5 Realizace protokolu ssh

Protokol ssh (Secure Shell) slouží obecně k práci na vzdáleném počítači, přihlášení se na vzdáleném počítači či ke spuštění skriptů na vzdáleném počítači. Ke komunikaci používá zabezpečený kanál. Klient se nejprve přihlásí na server, vymění si se serverem verze používaného protokolu, následně si vymění šifrovací klíče a poté již probíhá zabezpečená komunikace. Klient i server mohou být v mé implementaci libovolné počítače. Přihlášení, zjištění protokolu a výměnu klíčů jsem v simulaci realizoval opět krátkými vysíláními (1/100s). Následnou výměnu dat jsem opět realizovat pomocí krátkého vysílání (1/100s) vzhledem k malému množství přenášených dat.

2.6 Realizace protokolu pop

Pop je emailový protokol, který obecně slouží ke zjištění obsahu schránky, zjištění nové pošty, načtení celé zprávy nebo jen její části. Protokol se používá v offline modelu architektury, tj. klient si stáhne zprávu ze serveru (když je on-line) a následně ji může číst (může být off-line). Server je v mé implementaci pouze jeden (počítač, který reprezentuje internet), klientem může být kdokoli. Při používání pop protokolu klient se nejprve připojí na server, autorizuje se a následně probíhá požadovaná transakce. V rámci simulace budu předpokládat, že připojení na server a autorizace probíhá najednou, opět krátkým vysláním (1/100s). Transakce bude představována stažením e-mailu o délce 80kB (ze zadání). Realizace transakce bude opět simulována krátkým vysláním (1/100s).

2.7 Nalezené nedostatky simulátoru a jejich řešení

Laskavému čtenáři jistě neuniklo, že jsem pro malé množství přenášených dat vždy použil stejnou dobu přenosu, tj. 1/100s. Toto číslo sice neprezentuje přesně množství přenášených dat, nicméně bylo zvoleno úmyslně. Při různých pokusech v simulátoru jsem zjistil, že při použití menších časových úseků na ně simulátor nereaguje, tudíž se v simulaci tyto komunikace neobjevují.

Dalším nedostatkem, na který jsem přišel při realizaci semestrální práce, byla nestabilita při načítání topologie s více než cca 150 počítači (s nastavenou komunikací). Tato chyba se projevuje pádem celého programu. Chyba je pravděpodobně způsobena při parsování xml souboru s topologií sítě. Tento nedostatek způsobuje, že v simulátoru není možné simulovat celou síť. Zaměřil jsem se proto na simulaci těch uzlů, které při své komunikaci používají linku 4, tedy sledovanou linku. Z ostatních uzlů jsem náhodně zvolil uzly, které budou komunikovat. Počet uzlů jsem se snažil navýšit tak, aby aplikace fungovala.

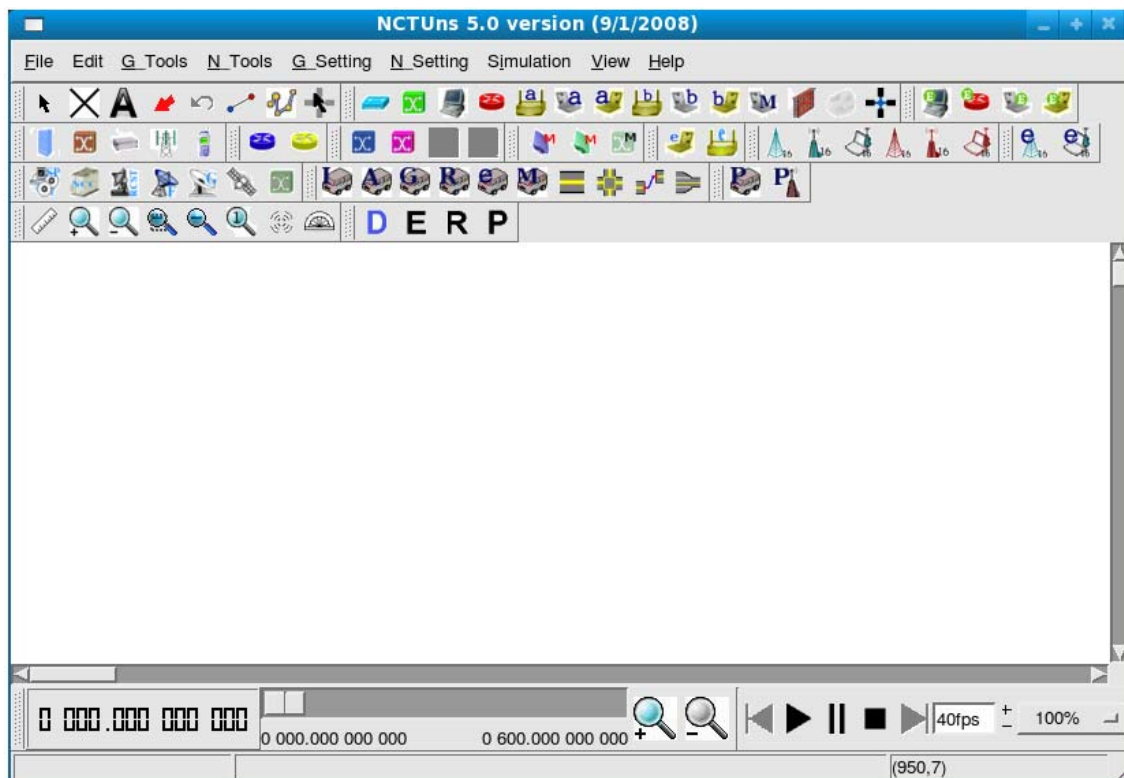
3. Uživatelská dokumentace

V následujících podkapitolách bude popsáno, jak se pracuje se simulátorem. Součástí odevzdané práce je i soubor s uloženou topologií sítě. Tento soubor stačí v simulátoru otevřít a spustit simulaci. Podrobný postup je popsán níže.

3.1 Spuštění simulátoru a otevření souboru s topologií

Před samotným spuštěním simulátoru je nejprve potřeba vykonat následující kroky. Nejprve je doporučeno vypnout službu iptables pomocí příkazu „service iptables stop“ spuštěným v terminálu. Vypnutí se doporučuje, aby nedocházelo k přesměrování paketů na jiné porty, nežli používá simulátor. Dále je potřeba spustit služby simulátoru – „coordinator“ a „dispatcher“. Tyto služby jsou nutné pro správný běh simulace sítě.

Při spuštění simulátoru se zobrazí okno, které je zobrazeno na následujícím obrázku (obr. č. 2). Pomocí položky „Open“ v menu „File“ lze otevřít již navrženou topologii sítě (soubory s příponou tpl).



Obr. č.2 – Okno simulátoru

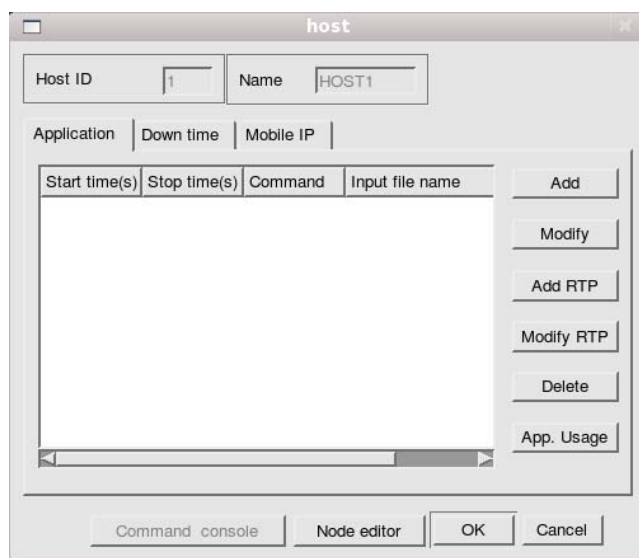
3.2 Editace topologie sítě

Nejprve bych se rád zmínil o tom, že simulátor má 4 režimy funkcionality. První režim je režim vytváření topologie sítě (označený písmenem D), druhým režimem je režim editace (označený písmenem E), třetím režimem je režim simulace (označený písmenem R) a posledním režimem je režim přehrávání simulace (označený písmenem P). Při navrhování a simulaci sítě bych doporučil postupovat přesně v tomto pořadí režimů. V návrhu je sice možné vrátit se zpět, ale při tomto lze snadno přijít např. o již nasimulovaná data. Mezi jednotlivými režimy se lze přepínat pomocí lišty umístěné v toolbaru (obr. č. 3).



Obr. č.3 – Lišta pro přepínání režimů

Topologie celé sítě je již navržena, můžeme se tedy podívat na editaci jednotlivých částí sítě (přepneme se do režimu editace). Zde stačí poklepat na jakoukoli komponentu sítě a nastavit jí potřebná data pro simulaci. Pokud poklepeme na uzel, zobrazí se nám okno pro přidávání příkazů komunikace (obr. č. 4).



Obr. č. 4 – Okno pro přidávání příkazů komunikace

Pro přidání nového příkazu stačí stisknout tlačítka Add, pro modifikaci již existujícího tlačítko Modify.

3.3 Simulace sítě

Pro simulaci sítě je potřeba se přepnout do režimu simulace. Pro spuštění simulace stačí vybrat položku Run v menu Simulation. Pokud jsme zapomněli na spuštění služeb „coordinator“ a „dispatcher“ popsaných výše, nepůjde simulace spustit. V rámci simulace je zvoleno logování vytíženosti linky 4. Výběr linky, která se má logovat, již není jednoduše popsatelná. Čtenáře bych proto rád odkázal na instruktážní videa ze stránek výrobce.

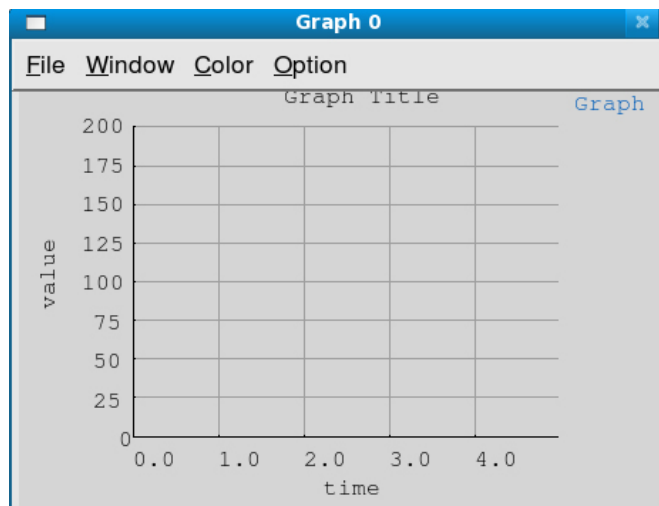
3.4 Přehrávání výsledků simulace

Pro přehrávání výsledků simulace se musíme přepnout do režimu přehrávání. Přehrávání lze spustit pomocí tlačítek umístěných v dolní liště okna (obr. č. 5). Přehrávat lze zpomaleně nebo zrychleně. Rychlost přehrávání lze nastavit také v dolní liště (standardně je nastaveno 100%). Přehrávání simulace se uskutečňuje nad obrázkem topologie, kde se zobrazují šipky nad jednotlivými linkami, které reprezentují směr toku dat.



Obr. č.5 – Tlačítka přehrávače

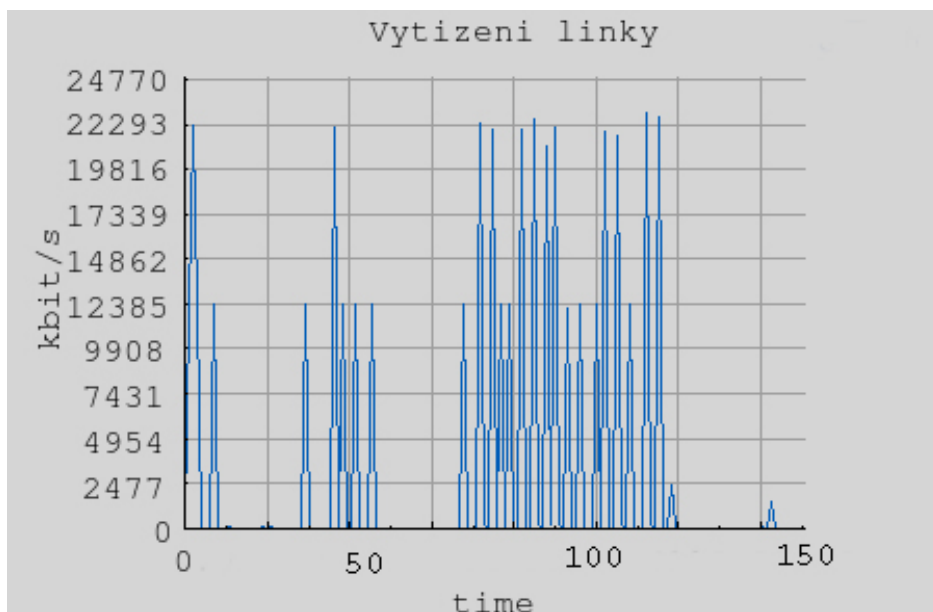
Pokud byla komunikace na nějaké lince logována, lze zakreslit graf této komunikace. Graf se vykresluje za výběru možnosti „Plot graf“ z menu „G_Tools“. Pro vykreslování grafu je pouze nutné v okně grafu (obr. č.6) vybrat soubor s logy komunikace položkou „Open“ z menu „File“. Samotné vykreslování grafu se děje společně s přehráváním výsledků simulace.



Obr. č.6 – Okno grafu

4. Výsledky měření zatížení linky 4

Výsledkem semestrální práce je graf (obr. č.7), který zobrazuje vytížení linky 4. Simulace probíhala na mém počítači (Intel Core 2 Duo 1,6Ghz; 2GB RAM) celkem 3 hodiny. Simuloval jsem od času 0 do času 2,5min (150s), protože v této době dochází na lince k největším přenosům dat. V pozdější době uzly komunikují „jen“ pomocí protokolu ssh, který výrazně neovlivňuje traffic na lince. Pozn. Veškerý přenos dat z/do internetu se odehrával v rámci linky 3, tudíž nijak neovlivňoval traffic na lince 4.



Obr. č.7 – Vytížení linky 4

5. Závěr

Prvním problémem, který jsem v projektu musel řešit, byla instalace samotného simulátoru. Mé zkušenosti s prací (instalací programů) pod operačním systémem linux byly téměř nulové. Nejprve jsem se pokusil nainstalovat simulátor metodou „pokus omyl“. Došlo bohužel k variantě omyl – simulátor nefungoval. Musel jsem tedy prostudovat materiály na stránkách výrobce, pomocí nichž již instalace takovým problémem nebyla.

Dalším problémem bylo samotné nastavování komunikace mezi uzly. Zde si myslím, že výrobce simulátoru dobře použil místo psaných návodů video návody. Pomocí těchto návodů se mi podařilo uskutečnit první komunikaci mezi uzly.

Posledním problémem, se kterým jsem se potýkal až v konečné fázi projektu, je rychlost simulace. Jak jsem již uvedl, samotná simulace trvala cca 3 hodiny. Musel jsem tedy celou síť připravit bez chyb v komunikaci, aby se vše povedlo hned při první simulaci. V tomto bodě bych ještě rád uvedl, že soubory, které popisují celou simulaci (včetně logu linky 4) zabírají necelý 1GB. Simulátor je tedy i dosti náročný na volné místo na HDD.

Výsledkem semestrální práce je graf, který prezentuje vytížení linky 4 v zadané topologii. Přes úskalí, kterým jsem musel čelit v rámci celé semestrální práce, a přes vynucené úpravy zadání, si myslím, že jsem zadání semestrální práce splnil.

6. Použité zdroje

Instruktažní videa výrobce - <http://nsl10.csie.nctu.edu.tw/support/documentation/DemoVideos/>

GUI User Manual - <http://nsl10.csie.nctu.edu.tw/support/documentation/GUIManual.pdf>

Protokol http - <http://www.jakpsatweb.cz/server/http-protokol.html>

Protokol ssh - http://cs.wikipedia.org/wiki/Secure_Shell

Protokol pop - <http://www.cs.vsb.cz/grygarek/kotasek/pop02.htm>