

Projekt do předmětu Směřované a přepínané sítě

**Nalezení a ověření nástroje na testování multicasingu
(generátor a přijímač, na Linuxu, příp. i Windows)**

Vypracoval:

Jakub Slepánek, SLE074

Zadání:

Cílem tohoto projektu bylo nalezení a ověření nástroje na testování multicasingu, tedy generátoru a přijímače na Linuxu, případně i Windows.

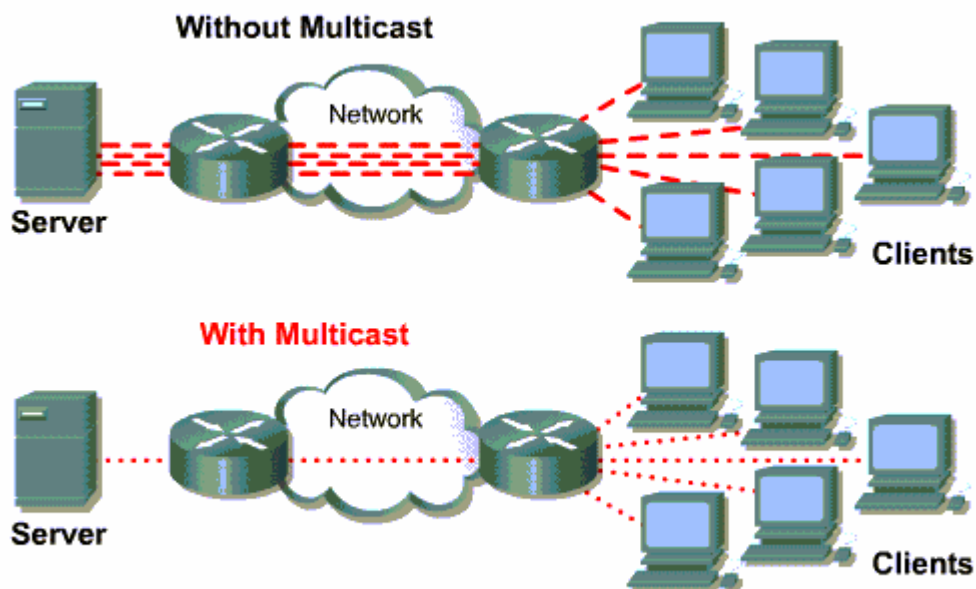
Úvod:

S rostoucími možnostmi dnešních počítačů, zvětšující se propustností Internetu a jeho pronikáním do běžného života vzrůstá také poptávka po službách jako jsou videokonference, internetové vysílání televize, rádia apod. Velká část těchto služeb se však dnes realizuje klasickým proudovým přenosem datagramů, což je velice nevýhodné. Zejména u zdroje může lehce nastat zahlcení linek, neboť tytéž datagramy se posílají po stejné lince několikrát. Možnost doručit jeden datagram z vysílajícího místa na různá přijímací místa v Internetu se tedy jeví jako důležitá věc.

Tento mechanismus doručování datagramu se nazývá **IP multicasting**.

Účelem multicastingu je tedy snížit zatížení linek při přenosu stejných dat mezi dvěma místy v Internetu. Místo toho aby se tytéž data přenášely v dané chvíli po lince několikrát, přenáší se pouze jednou a až při potřebě data rozestat do více míst kde již není možné data přenášet po jedné lince se data rozkopírují (například na jednotlivé porty Routeru ze kterých pak putují na jednotlivé příjemce multicastové skupiny).

Pro lepší představu si ukážeme fungování multicastingu na obrázku:



Obrázek1: Přenos dat na internetu bez a s použitím multicastingu

IP multicasting je jen abstrakcí multicasu na druhé vrstvě. Počítač (přesněji tedy síťové rozhraní) má možnost se začlenit do **multicastové skupiny** a poté bude dostávat datagramy určené pro tuto skupinu. IP multicasting má tyto vlastnosti:

- Skupinová adresa je třídy D. Několik multicastových adres je vyhrazeno pro konkrétní účely, další jsou volné k osobnímu použití. Adres je 2^{28} , není tedy třeba se obávat jejich nedostatku. Omezení by dříve nastalo ve směrovacích tabulkách účastníků se směrovačů.
- Hostitel se může libovolně do skupiny přihlašovat a odhlašovat, kdykoliv se mu zamane. Navíc může být současně členem více skupin.
- Pokud používaná druhá vrstva umožňuje multicasting, pak se používá. Jinak se simuluje IP multicasting pomocí broadcastu nebo unicastu.
- Členové multicastové skupiny mohou být na různých fyzických sítích, tudíž je potřeba, aby se na multicastingu podílely i směrovače.
- Multicastové datagramy se doručují stejně jako IP datagramy, mají stejné vlastnosti a podléhají stejným vlivům.
- Počítač může poslat multicastový datagram do libovolné skupiny, sám však nemusí být jejím členem. Členství v této skupině znamená pouze tyto datagramy přijímat.

Základní koncepty multicastingu

Pro efektivní implementaci IP multicastingu je potřeba dobře navrhnout tyto tři věci:

- Multicastové adresové schéma
- Efektivní notifikaci a doručování multicastových paketů
- Efektivní mezikřížové přeposílání multicastových paketů

Ne všechny problémy jsou na první pohled triviální. Adresovací schéma musí dovolit lokální autonomii při rozdělování adres, zatímco globálně musí adresy zůstat platné (a též unikátní nebo aspoň nekonfliktní). Je potřeba navrhnout notifikační mechanismus, kterým počítač řekne směrovačům, že má zájem na datagramech pro danou multicastovou skupinu. Taktéž je potřeba vyřešit doručovací mechanismus, jakým směrovače doručí pakety k cílům (HW multicastem, kde je to možné, jinak unicast nebo broadcast). Přeposílání datagramů je nejsložitější, je třeba zajistit, aby datagramy šly nejkratší cestou, aby se neposílaly do směrů, kde není žádný člen skupiny. Ovšem v okamžiku, kdy se v oblasti, kde provoz pro multicastovou skupinu neproudí, počítač do skupiny přihlásí, je potřeba tento provoz do takové oblasti nasměrovat.

Multicastové adresové schéma

Jak již bylo řečeno, IP adresy třídy D jsou vyhrazeny pro multicastové skupiny. Část adresového prostoru třídy D jsou tzv. dobře známé skupiny. Ty jsou permanentně přiřazeny k jistým účelům a existují, i když do nich není přihlášen zrovna žádný člen. Zbytek je volný k použití pro tzv. přechodné multicastové skupiny, které se vytvoří, když se do nich někdo přihlásí a zaniknou, když se odhlásí poslední člen.

Třída D IP multicastových adres, pro připomenutí, je v rozmezí 224.0.0.0 až 239.255.255.255. Permanentně přiřazené známé skupiny jsou 224.0.0.0/23 pro směrování a údržbu skupin, od 239.192.0.0 do 239.251.255.255 je privátní prostor adres pro jednu organizaci (viz. dále) a od 239.252.0.0 do 239.255.255.255 je privátní prostor adres pro jednu lokalitu. Speciální významné adresy jsou:

224.0.0.1 všechny systémy na lokální síti

224.0.0.2 všechny směrovače na lokální síti

Multicastová adresa je určena pouze pro cíl, nesmí se nikdy objevit ve zdrojové adrese datagramu. Navíc se nedoručují ICMP pakety při chybě směrování.

Multicastový obzor slouží k určení hranic skupiny. IP používá dvě techniky ke kontrole hranic. Jedna z nich je TTL bajt v hlavičce IP datagramu. Jestli např. nastavíme TTL bajt na jedničku, multicastová skupina se omezí pouze na lokální síť. Druhá metoda je známá jako **administrativní obzor**. Ta zakazuje internetovým směrovačům přeposílat ty pakety, které mají cílovou adresu organizace (239.192.0.0 až 239.251.255.255) nebo lokality (239.252.0.0 až 239.255.255.255).

Výsledky hledání nástroje na testování Multicastingu:

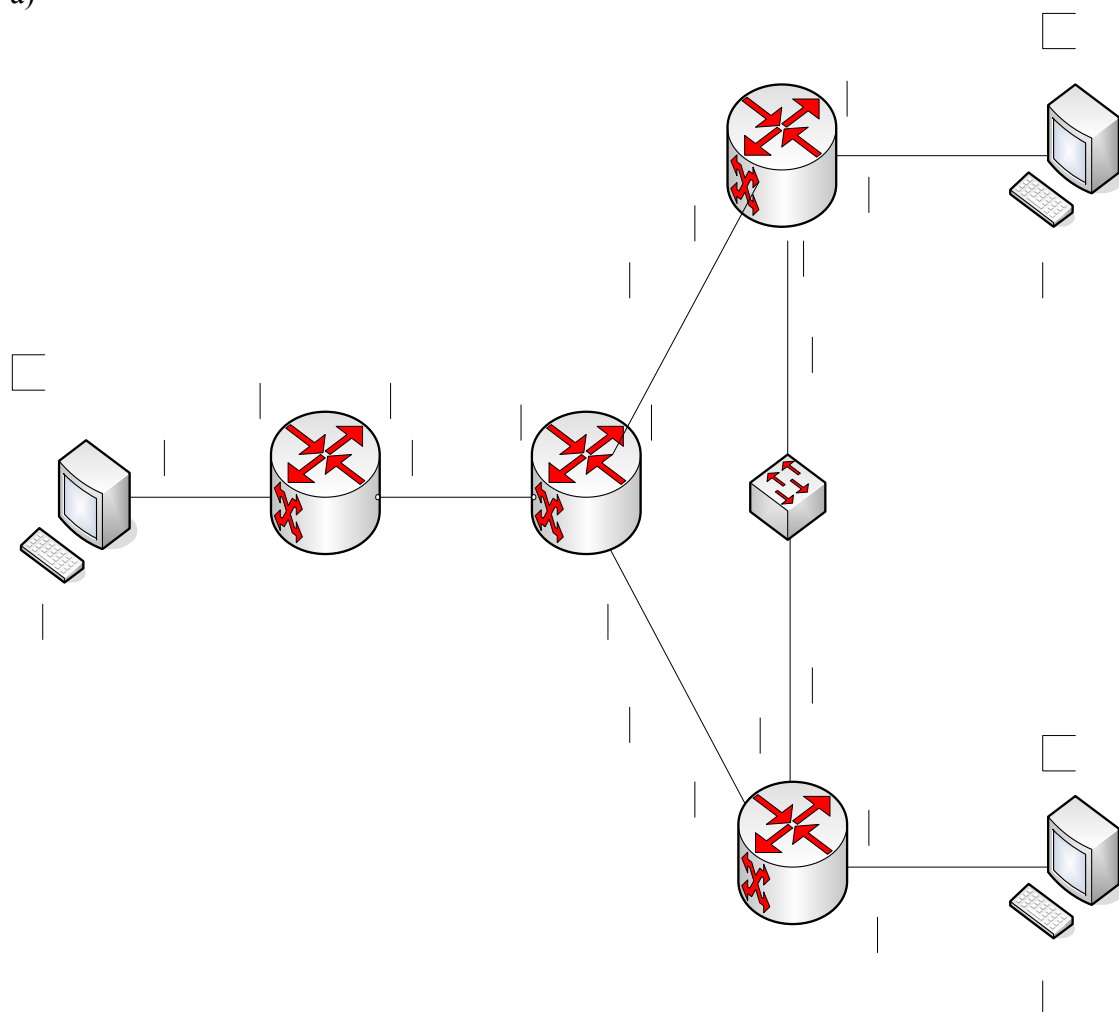
Při hledání na jsem použil internetový vyhledávač Google (www.google.com). Zkoušel jsem také různé P2P programy pro sdílení dat, ale nedošel jsem skoro k žádným výsledkům. Proto jsem raději strávil několik hodin hledáním v Googlu.

Výsledkem mého hledání bylo několik programů, ze kterých jsem vybral nástroj vytvořený v programovacím jazyku java, který jsem testoval pod Linuxem a druhý program pro testování pod Windows vytvořený v Delphi.

K popisu těchto dvou nástrojů a výsledků jejich testování se dostanu za chvíli. Předtím Vás ale seznámím s testovanými zapojeními na kterých jsem oba nástroje testoval.

Testované zapojení:

a)



Multicast sender

Při testování na tomto zapojení, které jsme sestavili ve školní laboratoři, jsem spolupracoval s kolegy Zábojníkem a Škařupou, kteří na tomto zapojení sledovali provoz protokolů PIM pro směrování multicastingu.

Zapojení se skládá ze čtyř routerů, jednoho switche a třech počítačů na kterých byly rozmístěny dva přijímače a jeden vysílač multicastingu. Vysílač (na obrázku vlevo) je připojen na router RJ a přijímače jsou připojeny na routery RA (v pravo nahoře) a RI (v pravo dole).

Bližší informace o zapojení získáte z jejich projektu s názvem:

Konfigurace a sledování provozu protokolů PIM pro směrování multicastů.

RJ

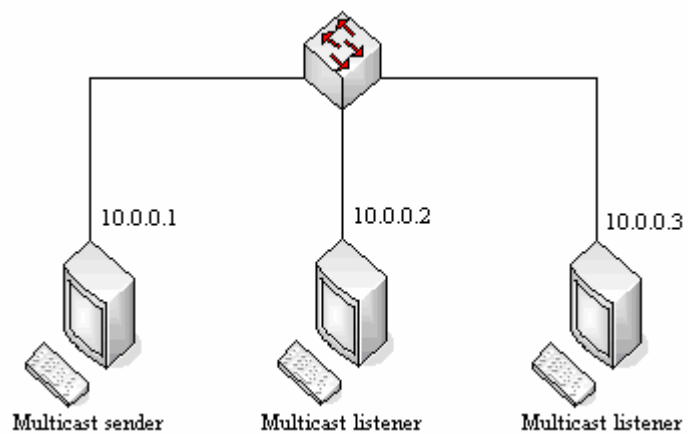
RG

100 1 100

2

100

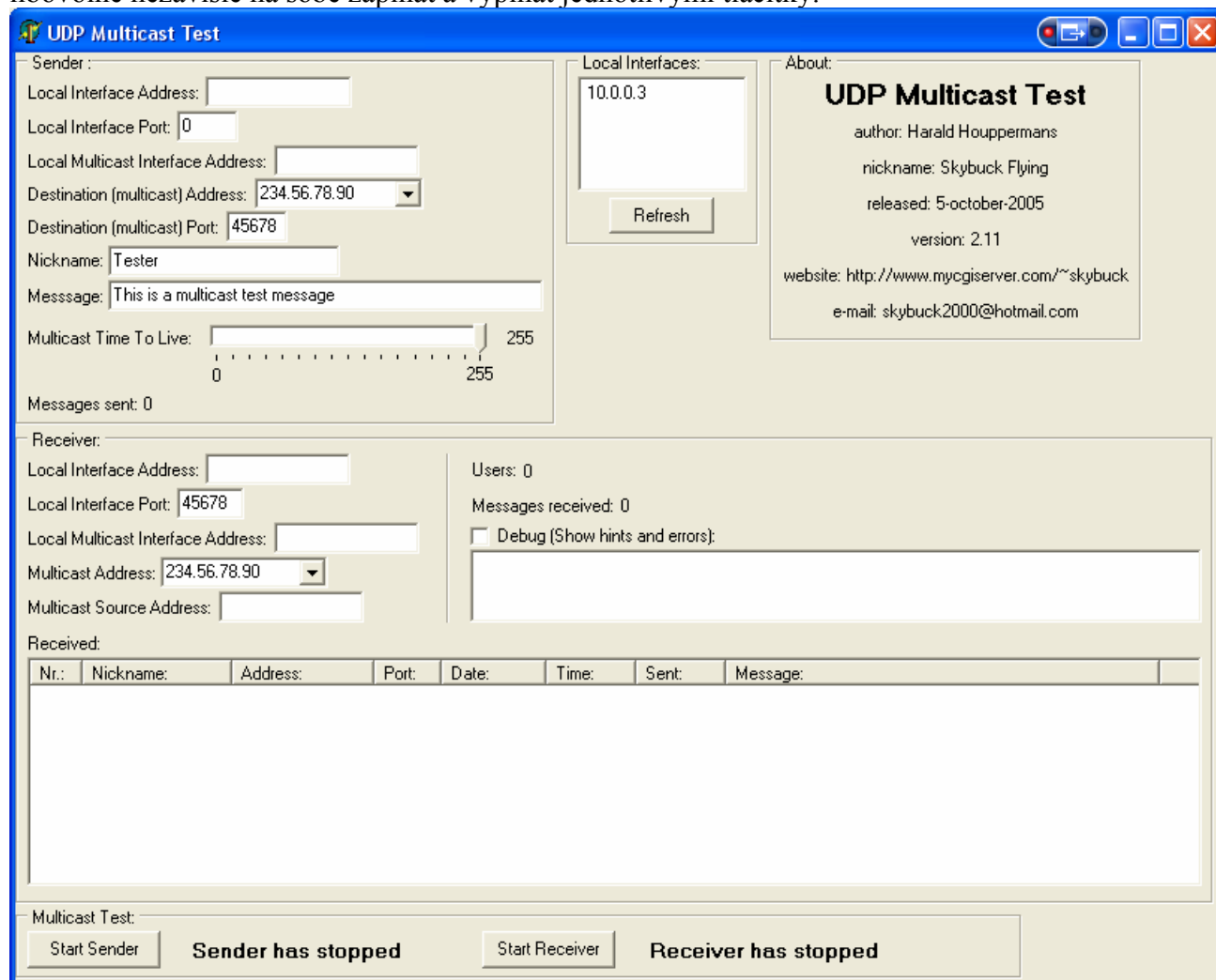
b)



Toto zapojení jsem si sestavil doma s mými dostupnými prostředky. Jako switch jsem použil nekonfigurovatelný osmiportový 10/100BaseT Ethernet mini switch Gembird.

Nástroj pod Windows

Pod operační systém Windows jsem na stránce <http://www.mycgiserver.com/~skybuck> našel jednoře pochopitelný program pro vysílání i příjem multicastu. Tento program má jen jeden soubor s názvem: *UDPMulticastTest.exe* a slouží zároveň pro vysílání i pro příjem. Vysílač i přijímač lze libovolně nezávisle na sobě zapínat a vypínat jednotlivými tlačítky.



Nyní si program lehce popíšeme:

Veškerá testování programu jsem prováděl pod Windows XP Professional Service Pack 2. Při spuštění automaticky načte IP adresy všech síťových karet (rozhraní) v položce Local Interfaces, kde si můžeme vybrat přes jaké rozhraní chceme multicasting přijímat nebo vysílat. Vysílači i přijímači můžeme přidělit jakoukoliv z načtených adres přetáhnutím myši. Pro funkčnost vysílače i přijímače stačí zadat IP adresu rozhraní do položek Local Interface Address. Možnosti kde všude lze IP adresu přidělit se zvýrazní (viz. obrázek 2).

Pokud chceme použít program jen pro vysílání, tak přesuneme IP adresu jen do části "Sender:". Pokud jen pro příjem, tak do části "Receiver:". Program může sloužit zároveň jako vysílač pro jednu multicastovou skupinu a přijímač pro jinou multicastovou skupinu.

The screenshot shows a network configuration window with two main sections: 'Sender' and 'Receiver'.

Sender Section:

- Local Interface Address: [Redacted]
- Local Interface Port: 0
- Local Multicast Interface Address: 10.0.0.3
- Destination (multicast) Address: 234.56.78.90
- Destination (multicast) Port: 45678
- Nickname: Tester
- Message: This is a multicast test message
- Multicast Time To Live: A slider set to 255 (range 0 to 255)
- Messages sent: 0

Receiver Section:

- Local Interface Address: [Redacted]
- Local Interface Port: 45678
- Local Multicast Interface Address: [Redacted]
- Multicast Address: 234.56.78.90
- Multicast Source Address: [Redacted]
- Users: 0
- Messages received: 0
- ☐ Debug (Show hints and errors):

Local Interfaces:

- 10.0.0.3
- Refresh button

Obrázek 2

Část Sender:

Local Interface Address – tady se zadává adresa síťového rozhraní přes které se bude vysílat.

Local Interface Port je číslo portu přes který se bude vysílat. Můžeme si zvolit jakýkoliv volný port. V případě, že nezadáme žádný nebo necháme standardně nastavenou nulu, tak se automaticky vybere nějaký volný port.

Do položky Destination (multicast) Address se zadává adresa multicastové skupiny. Lze zde zvolit jednu ze dvou adres 234.56.78.90 nebo 232.34.56.78. Případně lze zadat i jinou IP adresu třídy D vyhrazenou pro multicastové skupiny.

Destination (multicast) port je číslo portu na adrese multicastové skupiny a toto číslo musí být stejné i v položce Local Interface Address v části pro příjem.

Nickname – zde si můžeme zadat přezdívku nebo název pro vysílače.

Message – do tohoto pole zadáváme text který se bude šířit jako obsah multicastu.

Multicast Time To Live – tuto položku lze nastavit na hodnotu 0-255. Hodnota TTL se snižuje o 1 při každém průchodu přes router. Takže například při průchodu multicastu přes dva routery musí být hodnota TTL nastavena minimálně na hodnotu 3.

Messages sent: - zde se zobrazuje celkový počet multicastů odeslaných vysílačem.

Část Receiver:

Local Interface Address – tady se zadává adresa síťového rozhraní přes které se bude přijímat.

Local Interface Port je číslo portu na adrese multicastové skupiny a toto číslo musí být shodné s číslem portu v položce Destination (multicast) port.

Multicast Address je adresa multicastové skupiny a musí být stejná jako adresa u vysílače od kterého chceme přijímat multicasty.

Messages received – zobrazuje počet multicastů přijatých přijímačem.

Received:							
Nr.:	Nickname:	Address:	Port:	Date:	Time:	Sent:	Message:
1	Tester	10.0.0.3	1718	28-06-2006	23:50:45	17	This is a multicast test message
2	Tester	10.0.0.3	1773	29-06-2006	00:52:19	33	This is a multicast test message
3	Tester	10.0.0.3	1774	29-06-2006	00:52:50	60	This is a multicast test message
4	Vysilac	10.0.0.3	1775	29-06-2006	00:57:06	66	Zprava vyslana vysilacem multicastu

Obrázek 3

Na obrázku 3 vidíme tabulku přijatých multicastů, kde je uveden název vysílače (Nickname), adresa, port, datum, čas, nejvyšší číslo přijatého multicastu a text obsažený v daném multicastu.

Nástroj pod Linux

Na adrese <http://www.nmsl.cs.ucsb.edu/MulticastSocketsBook/> jsem našel vysílač a přijímač multicastu napsaný v programovacím jazyku java, C a C#. Tyto programy jsou zde volně ke stažení jako doplňky ke knize: *Multicast Sockets: Practical Guide for Programmers*. Vybral jsem si program v jazyce java a použil jej pro testování pod operačním systémem Linux.

Program se skládá ze dvou souborů s názvy *mcsend.java* a *mcreceive.java*. Mcsend slouží jako vysílač a mcreceive jako přijímač multicastu. Každý z těchto dvou souborů se musí před použitím zkompileovat a poté spustit s příslušnými parametry kterými jsou IP adresa multicastové skupiny a číslo portu.

kompilace vypadá takto:

```
javac mcsend.java
javac mcreceive.java
```

spuštění provedeme takto:

- vysílač:

```
java mcsend <MULTICAST_IP> <PORT>
```

Po spuštění lze na obrazovce napsat text, který se odešle stisknutím klávesy *Enter*. Můžeme odesílat libovolný počet zpráv. Pokud chceme vysílač vypnout provedeme to pomocí stisku *Ctrl+c*.

- přijímač:

```
java mcreceive <MULTICAST_IP> <PORT>
```

Po spuštění se nám na obrazovce objevují přijaté zprávy z vysílače které obsahují velikost v bytech, adresu vysílače a text zprávy. Přijatá zpráva může vypadat například takto:

Received 16 bytes from /10.0.0.3: Zpráva pro test.

U přijímače a vysílače musí být zadán port v rozmezí 1024 – 65535 a IP adresa musí být ve správném tvaru jako například: 234.56.78.90

Další možností u vysílače je nastavení hodnoty TTL, které se musí provést přepsáním přímo ve zdrojovém kódu souboru před kompilací a následným spuštěním.

Při testu obou zapojení pod Windows vše fungovalo bez problémů. Pro zapojení s routery (zapojení *a*) jsem zkoušel testovat nastavování hodnoty TTL. Mezi vysílačem a přijímačem jsou v zapojení 3 routery přes které se multicast šířit. Na každém routeru se sníží hodnota TTL o 1. Zkoušel jsem postupně zvyšovat hodnotu TTL od 1 a teprve až při nastavení TTL na hodnotu 4 se na přijímači začaly objevovat přijaté multicasty. Tedy nastavení TTL u tohoto programu funguje jak má. Při testování zapojení *b* nebylo třeba nastavovat hodnotu TTL. Multicast se šířil od vysílače na všechny přijímače bez jakýchkoliv problémů.

Pokoušel jsem se procházet zdrojový kód, ale nepřišel jsem na konkrétní důvod proč vysílání přes routery neprochází.

Received	79 bytes from /10.0.0.3:	OOUDPMLTICASTTESTVysilac	oooooooooooooKoooZprava	vyslana	vysilacem	multicastuO		
Received	79 bytes from /10.0.0.3:	OOUDPMLTICASTTESTVysilac	ooooooooooooooooo1oooZprava	vyslana	vysilacem	multicastuO		
Received	79 bytes from /10.0.0.3:	OOUDPMLTICASTTESTVysilac	ooooooooooooooooomoooZprava	vyslana	vysilacem	multicastuO		
Received	79 bytes from /10.0.0.3:	OOUDPMLTICASTTESTVysilac	ooooooooooooooooooKoooZprava	vyslana	vysilacem	multicastuO		
Received	79 bytes from /10.0.0.3:	OOUDPMLTICASTTESTVysilac	ooooooooooooooooooKoooZprava	vyslana	vysilacem	multicastuO		

Po tomto testu jsem se zamyslel nad tím jestli je vysílač *mcsend.java* správně naprogramován. Ovšem kdyby byl vytvořen špatně, tak by nefungovalo vysílání ani pod Linuxem. Takže jediným problémem asi bude špatné naprogramování TTL.

Zhodnocení:

Výsledkem této práce mělo být nalezení a ověření nástroje pro testování multicastingu pod Linuxem případně Windowsem. Nástrojů pod Linux jsem sice našel několik, ale ani jeden nedokázal vysílat přes routery. Sám bych se neodvážil vytvořit (například v javě) opravdu fungující nástroj, protože programování není mojí silnou stránkou a tak mi zbylo jen hledat všude možně nějaký program který funguje, což se mi ovšem nepovedlo. Důvodem může být mé špatné hledání, nebo nedostupnost funkčního nástroje volně na internetu.

Pozitivním výsledkem bylo ovšem nalezení opravdu funkčního nástroje pod Windows. Tento program je nejen velice přehledný a snadno pochopitelný, ale lze v něm měnit hodnoty i při běhu programu a tím urychlit průběh testování. Například hodnotu TTL lze takto nastavovat jednoduše jakýmsi "posuvníkem", oproti tomu v programu pod Linuxem se musí po každé změně hodnoty TTL zdrojový kód překompilovat a znovu spustit.

Odkazy na zdroje ze kterých jsem čerpal:

UDPMulticastTest:

<http://www.mycgiserver.com/~skybuck>

e-mail autora skybuck2000@hotmail.com

Multicast Sender and Receiver (java):

<http://www.nmsl.cs.ucsb.edu/MulticastSocketsBook/>

Zdroje informací k problematice multicastingu:

<http://www.cs.vsb.cz/grygarek/SPS/lect/multicast/multicast.html>

<http://www.isdn.cz/>