

Protokoly transportní vrstvy a jejich kategorizace, transportní protokol ARTP

Tomáš Rebok (xrebok@fi.muni.cz)

Fakulta informatiky
Masarykova univerzita v Brně
Botanická 68a, 602 00 Brno

CESNET, z.s.p.o.
Zikova 4, 160 00 Praha 6

Abstrakt

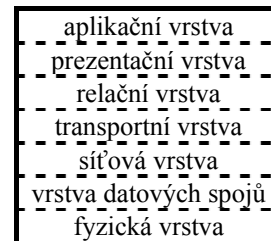
Transportní protokoly tvoří nedílnou součást dnešních počítačových sítí. Jak již jejich název napovídá, zajišťují pro aplikační vrstvu transport (neboli přenos) dat od vysílajícího uzlu sítě směrem k jejich příjemci. Jelikož mají aplikace odlišné požadavky na kvalitu tohoto přenosu (známo též pod pojmem *QoS – Quality of Service*), vznikla řada specializovaných transportních protokolů s rozdílnými vlastnostmi.

Jako součást vyvíjeného aktivního směrovače byl navržen nový spojově orientovaný transportní protokol – protokol ARTP (*Active Router Transport Protocol*). Jeho základní vlastnosti (které jsou v příspěvku porovnány s již existujícími transportními protokoly) se odvíjejí především od potřeb vyvíjeného směrovače; protokol je však navržen tak, aby byl použitelný i mimo prostředí aktivních sítí. ARTP zajišťuje spolehlivý duplexní přenos dat bez nutnosti zachování pořadí přijatých datagramů, detekci duplicit, řízení toku dat pro detekci a reakci na zahlcení sítě, *multiplexing aid*. V příspěvku jsou představeny důvody vedoucí k jeho návrhu a požadavky na něj kladené, filozofie návrhu a vytvořená prototypová implementace, která zahrnuje veškerou funkčnost navrženého protokolu.

Efektivita protokolu byla s využitím vytvořené prototypové implementace testována na 100 Mbit/s a 1 Gbit/s sítí. Výsledky testů jsou v příspěvku srovnány s běžně používanými existujícími transportními protokoly a možné použití navrženého protokolu je diskutováno například v souvislosti s multimediálními přenosy.

1 Úvod

Mezinárodní organizací ISO byl v roce 1979 představen jednotný standard pro vzájemné propojování systémů různých typů a koncepcí nazývaný ISO/OSI model (viz Obrázek 1, více viz [1]). Navržený model má sedm vrstev, které tvoří hierarchii začínající aplikacemi na vrcholu a končící fyzickými spojeními vespod. Přestože se tento model v praxi neuplatnil, nadále slouží jako metoda popisu komunikačních systémů.



Obrázek 1: ISO/OSI model

Transportní protokoly spadají do čtvrté vrstvy představeného referenčního modelu, což přeneseno do prostředí dnešních sítí (ve kterých přesný ISO/OSI model není implementován) znamená, že jsou umístěny přímo mezi vrstvou aplikační a vrstvou síťovou. Jak již jejich název napovídá, zajišťují pro aplikační vrstvu (s využitím služeb poskytovaných vrstvou síťovou) transport (neboli přenos) dat od vysílajícího uzlu sítě směrem k jejich příjemci. Jelikož síťová vrstva poskytuje pouze *best-effort* službu přenosu dat (tj. přenáší pakety, ale nezaručuje, že se cestou ke svému příjemci neztratí, neduplikují apod.), je hlavním úkolem transportních protokolů zajistit aplikační vrstvě spolehlivost a kvalitu přenosu v rámci *end-to-end* komunikace tak, jak ji daná aplikace požaduje (a během celého přenosu požadovanou kvalitu služby také udržovat).

Jednotlivé aplikace mají na kvalitu tohoto přenosu odlišné nároky. Některé požadují zcela spolehlivý přenos dat, jiné (jako například aplikace přenášející video či zvuk v reálném čase) se orientují především na rychlost přenosu a jakékoli zpoždění způsobené reakcí na ztrátu přenášeného paketu by pro ně mohlo být nepřijatelné. Transportní protokoly jsou tak nejčastěji děleny pouze podle kritéria spolehlivosti přenosu – v případě spolehlivých (anglicky *reliable*) protokolů se nejčastěji mluví o protokolu TCP (*Transmission Control Protocol*), v případě nespolehlivých (*unreliable*) protokolů je zmiňován především protokol UDP (*User Datagram Protocol*). Toto dělení je však silně zjednodušené, neboť existuje řada protokolů se specifickými vlastnostmi, díky nimž je nelze zařadit ani ke spolehlivým protokolům typu TCP, ani k nespolehlivým protokolům typu UDP (například zajišťují spolehlivý přenos dat bez zachování pořadí, nepoužívají fragmentaci a zpětnou defragmentaci přenášených dat atd.).

V další části příspěvku budou uvedeny charakteristické vlastnosti transportní vrstvy a následně budou uvedeny specifické vlastnosti několika existujících

transportních protokolů s cílem ukázat, že při psaní speciálních aplikací není potřeba omezovat se pouze na aktuálně nejrozšířenější protokoly TCP a UDP, ale lze s výhodou využít vlastnosti jiných transportních protokolů. V závěrečné části pak bude podrobněji představen transportní protokol ARTP, který byl navržen pro transportní vrstvu aktivního směrovače vyvíjeného ve spolupráci zájmového sdružení CESNET a Masarykovy univerzity v Brně.

2 Vlastnosti transportních protokolů

Nejjednodušší dělení protokolů transportní vrstvy je na dva základní typy:

- **transportní protokoly bez spojení** – tyto protokoly pouze zajišťují přenos datových bloků od jejich odesílatele k příjemci, nezaručují žádnou kvalitu služby,
- **transportní protokoly se spojením** – před přenosem dat samotných je mezi odesílatelem a příjemcem ustaveno spojení, které je po celou dobu komunikace udržováno. Data jsou přenášena až poté, co je spojení vyjednáno a mezi oběma komunikujícími stranami jsou dohodnuty jeho parametry. Na konci přenosu je pak ustavené spojení uvolněno.

***Poznámka:** S výše uvedeným dělením úzce souvisí ještě jeden aspekt, který je pro protokoly aplikační vrstvy velmi podstatný. V případě transportních protokolů bez spojení jsou jednotlivé bloky dat přenášeny samostatně a zcela nezávisle na jiných blocích, přičemž příjemce jednotlivé bloky dat rovněž chápe jako zcela samostatné celky a ne jako součást nějakého většího celku. Tomu pak odpovídá i skutečnost, že příjemce nemění svůj stav v závislosti na průběhu komunikace – tento způsob přenosu je tedy možno charakterizovat jako bezstavový (stateless). Naproti tomu má každá spojovaná služba nutně stavový charakter, neboť již pouhým navázáním spojení se příjemce dostává do jiného stavu, než v jakém byl předtím.*

Bezstavový způsob komunikace je tak vzhledem ke své podstatě dobře odolný vůči různým nestandardním situacím, především pak výpadkům a ztrátám dat. Naproti tomu, má-li být zachována konzistence právě probíhajících činností, mohou být v případě stavového způsobu komunikace určité nápravné akce zapotřebí.

Stejně jako v případě všech ostatních vrstev ISO/OSI modelu jsou i na transportní vrstvě definovány základní služby, které pak tato vrstva může poskytovat vrstvám sousedním. Patří k nim zejména:

- **Ustavení spojení a multiplexing** – odesílatel před začátkem přenosu dat kontaktuje příjemce a oba si (většinou s využitím třicetného „handshake“) ustaví nové spojení, které je pak využito pro odesílání dat. Obě strany si mohou v jeden okamžik ustavit více takovýchto spojení – pakety z těchto spojení jsou pak multiplexovány nad jednou fyzickou linkou (a při příchodu zpětně demultiplexovány).

- **Pomalý start a kontrola zahlcení** – aby se předešlo náhlému zahlcení sítě, posílá odesílatel na začátku komunikace data pomaleji. V případě, že síť není zahlcena, je rychlost odesílání navýšena, aby byla síť co nejefektivněji využívána. Jakmile se síť začne zahlcovat (detekováno například zvýšenými ztrátami paketů), odesílatel opět sníží rychlost odesílání, přičemž po určité době se jí snaží opět navýšit, aby byla síť stále efektivně využívána.
- **Mechanismy řízení toku dat** – zatímco při kontrole zahlcení je zkoumáno zahlcení sítě, mechanismy řízení toku dat pomáhají zabránit zahlcení příjemce příliš velkým objemem dat, které by nestíhal zpracovat – odesílatel reaguje snížením množství odesílaných dat.
- **Služby pro zajištění důvěryhodnosti přenosu** – tyto služby jsou využívány pro nápravu defektů sítě, které mohou mít za následek narušení přenášeného datového toku (zničené, ztracené, přeskládané či zahozené pakety). Odesílané pakety jsou číslovány, aby mohly být na straně příjemce zmíněné defekty detekovány (a v případě jejich zjištění je pak s přispěním odesílatele sjednána náprava).

3 Transportní protokoly

3.1 UDP (User Datagram Protocol)

Jeden ze dvou neznámějších transportních protokolů si lze zjednodušeně představit jako jednoduchou „obálku“ nad síťovým protokolem IP, která nijak nemění povahu ani kvalitu jeho přenosových služeb, ale pouze zprostředkovává své služby vrstvě vyšší. V podstatě jediné, co UDP zajišťuje navíc, je multiplexování a demultiplexování datagramů (podle čísla portu). Aplikační programy využívající UDP k přenosu dat pak na sebe přebírají veškerou zodpovědnost za zajištění takové úrovně kvality přenosů, jakou samy potřebují.

Vhodné použití: jelikož UDP poskytuje nespojované a nespolehlivé přenosové služby, je vhodné jej použít v prostředí, ve kterém je zapotřebí rychlý a efektivní transportní protokol a případné výpadky či jiné defekty sítě nejsou pro přenos kritické (případně se s nimi vyrovná sama aplikace). Příkladem takového prostředí je například přenos audio/video v reálném čase.

Další informace: RFC 768

3.2 RDP (Reliable Data Protocol)

RDP je spojově orientovaný transportní protokol určený pro spolehlivý přenos dat na paketově orientovaných sítích. Kvůli své efektivnosti a jednoduché implementovatelnosti byla před jeho návrhem vytyčena minimální množina funkcí, které by měl implementovat (ve srovnání s protokolem TCP se tak nesnaží zajišťovat rozsáhlou a složitou funkčnost, jeho ideou je

podporovat pouze nezbytnou množinu funkcí, avšak co nejjednodušeji a nejeftivněji). Komunikuje přímo se síťovým protokolem IP a poskytuje spolehlivý přenos dat, přičemž zaručení pořadí přijatých datagramů si může aplikace při vytváření spojení sama vyžádat (či odmítnout). Samozřejmě implementuje i mechanismy řízení toku dat a zábrany zahlcení sítě (oproti TCP však velmi zjednodušené).

Na základě tohoto protokolu byl navržen a implementován transportní protokol RUDP (*Reliable User Datagram Protocol*), který však není umístěn přímo nad síťový protokol IP, ale až nad transportní protokol UDP (jedná se o rozšíření protokolu UDP o sadu funkcí zajišťující spolehlivý přenos dat). Navíc neimplementuje veškerou funkčnost protokolu RDP (například možnost volby, zda mají být data aplikaci předávána ve správném pořadí či nikoli).

Vhodné použití: RDP je možno využít jako alternativní protokol k protokolu TCP v mnoha aplikacích, ve kterých je zapotřebí spolehlivý, jednoduchý a efektivní komunikační protokol.

Další informace:

- RFC 908, RFC 1151
- <http://www.ietf.org/proceedings/99mar/I-D/draft-ietf-sigtran-reliable-udp-00.txt>

3.3 XTP (*Xpress Transport Protocol*)

Protokol XTP je vhodný komunikační protokol pro Internet, intranet, aplikace pracující v reálném čase či multimediální síťové aplikace. Může pracovat nad libovolnou síťovou vrstvou (IP, CLNP), libovolnou linkovou vrstvou (LLC, MAC) či přímo nad vrstvou AAL v ATM sítích. V tomto jediném protokolu je zahrnuta funkcionalita transportních protokolů TCP, UDP a TP4, spolu s dalšími službami, jako například přenos multicastu, správa multicastových skupin, prioritizace transportní vrstvy, popis síťového provozu pro možnosti vyjednávání *Quality of Service*, kontrola rychlosti přenosu, zasílání dat mimo pořadí či možnost volby mechanismů pro řízení toku a korekce chyb (plně spolehlivý režim přenosu pracující podobně jako TCP, nespolehlivý režim podobně jako v případě UDP či spolehlivý přenos s rychlým negativním potvrzováním pro chybějící pakety). Jelikož se opět jedná o spojově orientovaný protokol, implementuje také možnosti správy vytvořených spojení (spojení se však dopředu nevyjednává, data mohou být přenášena již prvním zasílaným paketem).

Vhodné použití: Přestože lze protokol XTP použít pro podporu velkého množství aplikací, je především určen pro náročné aplikace sloužící ke speciálním účelům (telekonference, digitální multimediální servery a distribuce multimédií, distribuované simulace, algoritmy pro správu velkých multicastových skupin či podporu dalších protokolů (například RTP, RSVP, NTP a dalších). Jedním z hlavních cílů jeho návrhu byla jeho vysoká výkonnost (aby dokázal přená-

šet velké toky dat s nízkou latencí), proto jej lze s výhodou použít i v náročných aplikacích.

Další informace:

- <http://www.cs.virginia.edu/~acw/netx/>
- <http://www.mentat.com/xtp/XTP40b.pdf>

3.4 NETBLT (*NETwork BLock Transfer*)

NETBLT je transportní protokol určený pro rychlý a spolehlivý přenos velkých bloků dat. Pracuje přímo nad síťovým protokolem IP (avšak měl by být schopen spolupracovat i s jiným síťovým protokolem), umožňuje řízení toku přenášených dat a snaží se na dané síti poskytnout maximální možnou přenosovou rychlost a být tak rychlejší než ostatní transportní protokoly.

Před vlastním přenosem je vytvořeno spojení, přes které jsou pak jednotlivé (ideálně velké) bloky dat přenášeny v blokujícím režimu přenosu (další blok dat nelze odeslat před tím, než od příjemce dojde potvrzení o přijetí předchozího přenášeného bloku). Opět podporuje i mechanismy řízení toku dat a zábrany zahlcení sítě.

Vhodné použití: Protokol NETBLT je možno použít v aplikacích přenášející velké bloky dat (například přenosy multimediálních dat). Jeho rysy umožňují výhodné použití i na vysokolatenčních kanálech, aniž by tím byla oslabena výkonnost na vysokorychlostních sítích.

Další informace: RFC 998

3.5 SCTP (*Stream Control Transmission Protocol*)

SCTP je spolehlivý transportní protokol pracující nad nespolehlivou IP vrstvou. Oproti TCP je orientován na přenos zpráv namísto přenosu proudu jednotlivých bytů. Po navázání spojení (v terminologii SCTP nazýváno asociace) dokáže přenášet řadu navzájem nezávislých proudů (*streams*), v rámci každého z nich pak SCTP garantuje doručení všech dat ve správném pořadí. Případný výpadek (a pozdější opakování, čili zdržení) v některém z proudů se však nijak netýká proudů ostatních, jejich komunikace tak pokračuje bez přerušení. Schopnosti SCTP asociace by se tedy daly přirovnat ke svazku souběžných TCP spojení, avšak s menší režii. Aby byla zachována možnost jeho koexistence s protokolem TCP, implementuje podobné mechanismy na řízení toku a kontrolu zahlcení.

Zajímavostí SCTP je také to, že podporuje tzv. *multihoming* – tedy situaci, kdy má komunikující uzel několik IP adres. Dostupné adresy si partneři vyměňují při vytvoření asociace a může se jednat o libovolnou směs IPv4 a IPv6 adres. Během komunikace je jedna z nich brána jako primární a na ni jsou odesílána data. Pro opakování (retransmisi) nepotvrzeného paketu však vybírá jinou a pokud má primární adresa častější

problémy se svou dostupností, přejde odesílatel na alternativní (je-li k dispozici).

Vhodné použití: vhodným prostředím pro nasazení protokolu SCTP je to, ve kterém přenášíme více na sobě nezávislých proudů dat (například při webové komunikaci prostřednictvím protokolu HTTP). V takovéto situaci totiž potřebujeme spolehlivý přenos jednotlivých proudů dat při zachování pořadí přijatých paketů pouze v rámci jednotlivých proudů, nikoli v rámci veškeré komunikace (jako to zajišťuje protokol TCP), čímž lze dosáhnout výrazného zefektivnění její rychlosti.

Další informace:

- RFC 2960, RFC 3286 (a dodatky)
- referenční implementace na http://people.cs.uchicago.edu/~piggy/sctp_chicago_il_us/

3.6 TCP (Transmission Control Protocol)

Spojově orientovaný transportní protokol, jenž nabízí zcela spolehlivý přenos dat, řízení toku, kontrolu zahlcení sítě atp. Jelikož se jedná o nejrozšířenější transportní protokol současných sítí, není nutno jej zde dále popisovat.

Vhodné použití: Protokol TCP je v dnešních sítích hojně využíván v prostředí aplikací požadujících zcela spolehlivou komunikaci.

Další informace: RFC 793 (a dodatky)

4 ARTP (Active Router Transport Protocol)

Protokol ARTP je primárně určen pro použití ve vyvijeném aktivním směrovači (viz [5]), a proto se jeho vlastnosti a chování odvíjí především od jeho potřeb. Všechny zásadní požadavky na něj kladené lze shrnout do následujících bodů:

- **Základní přenos dat** – navržený komunikační protokol musí být mezi dvěma komunikujícími uzly schopen v libovolném směru přenášet souvislé bloky dat. Přenášena data mohou být dvou druhů – řídicí data, jejichž úkolem je přenos spravovat, a aplikační data, což jsou vlastní přenášena data. U aplikačních dat by měl protokol také pamatovat na možnost jejich šifrování a podepisování.
- **Spolehlivost přenosu** – protokol musí zaručit spolehlivý přenos dat (všechna odesílacím uzlem vyslaná data musí být přijímacím uzlem řádně přijata). Musí se vypořádat se ztrátami, zpožděními či duplicitami dat a všechny tyto defekty sítě řádně opravit. Přijaté bloky dat nemusí být přijímací aplikaci předány v pořadí, v jakém byly odeslány (tj. není zapotřebí zachovávat pořadí přijatých datagramů).
- **Řízení toku dat** – vysílající strana protokolu je povinna sledovat a řídit množství do sítě odesí-

laných dat (*congestion control*). V případě, že komunikace probíhá bez problémů, může toto množství zvyšovat. Jakmile se začne přijímací strana nebo síť odesílanými daty zahlcovat, musí protokol podniknout nutné kroky k odstranění tohoto stavu.

- **Multiplexing** – protokol by měl umožnit vytvoření více spojení mezi dvěma komunikujícími stranami určenými svými IP adresami a porty. Tato spojení musí být vzájemně rozlišitelná a jednoznačně identifikovatelná.

4.1 Filozofie protokolu ARTP

Protokol ARTP je spojově orientovaný transportní protokol umožňující spolehlivý přenos dat bez nutnosti zachovávat jejich pořadí. Klientská aplikace, využívající protokol ARTP pro přenos dat na serverovou aplikaci, je před vlastním přenosem nucena s přijímací stranou vytvořit spojení (v protokolu je označováno jako relace), které je po ukončení přenosu potřeba uzavřít.

Vzhledem k požadavku multiplexity protokolu (možnost více spojení mezi dvěma komunikujícími stranami) poskytuje ARTP identifikační číslo relace – každé vytvořené spojení je tak jednoznačně určeno IP adresou a portem obou komunikujících stran a svým identifikačním číslem. Tento způsob návrhu umožňuje vytvářet ve stejném časovém okamžiku mezi dvěma aplikacemi více spojení – tato spojení jsou pak vzájemně odlišitelná přiděleným identifikátorem. Vytvoření relace je klientem iniciováno odesláním požadavku na server, který tento požadavek vyhodnotí a klientovi odešle své rozhodnutí. V případě kladné odpovědi se po oboustranném vytvoření potřebných struktur pro zajištění správného chodu přenosu stává relace ustavenou.

Ustavená relace může být využita pro obousměrný přenos dat, která mohou být v protokolu ARTP dvou druhů – řídicí a aplikační. Řídicí data jsou určena k řízení chování komunikujících aplikací a jejich dorozumívání se. Každá řídicí informace může obsahovat svůj identifikátor, typ (zda se jedná o požadavek nebo o odpověď) a svou hodnotu. Aplikační data jsou vlastní přenášena data; jsou členěna do datagramů a protokolu jsou předávána ve dvou částech (protokol umožňuje rozdělení přenášných dat na vlastní data a jejich podpisovou informaci). Každý předávaný datový datagram je označen svým identifikátorem (datovým sekvenčním číslem), který si určuje samotná vysílající aplikace. Velikost přenášných datagramů není protokolem omezena.

Fyzický přenos probíhá formou paketů, do kterých jsou přenášena data vkládána. Pakety jsou složeny z hlavičky, která obsahuje nutné informace o přenášném paketu, a datové části, v níž jsou zakódována přenášena data. Každý paket může navíc obsahovat řídicí informace protokolu, které lze využít pro komunikaci vysílající a přijímací části protokolu (například řízení maximální velikosti přenášných paketů, vzájemné prioritizaci ustavených spojení, atd.) a nejsou předává-

ny aplikacím. Konkrétní přenášené informace si může každá implementace protokolu ARTP stanovit zcela libovolně.

Hlavička ARTP paketu dále obsahuje informaci o času jeho odeslání (*timestamp*) a době, po jejímž uplynutí přestává být paket platný (doba expirace). Tyto informace jsou nezbytné pro detekci zpožděných paketů¹.

Spolehlivost a důvěryhodnost přenosu je zajištěna mechanismem potvrzování přijatých paketů a detekcí jejich duplicit. Každý paket má ve své hlavičce uvedeno identifikační (sekvenční) číslo, které jej jednoznačně identifikuje. Přijímající strana protokolu odesílateli periodicky zasílá seznam nově přijatých sekvenčních čísel (těch, která byla přijata od doby odeslání posledního potvrzení), čímž mu potvrzuje jejich příjem. Jakmile je nějaký paket vyhodnocen jako ztracený (po definovanou dobu na něj nepřišlo potvrzení), je příjemci znovu odeslán (retransmise).

Protokol ARTP řídí množství do sítě vysílaných dat, aby nedocházelo k zahlcení příjemce nebo sítě a tím velké degradaci rychlosti a kvality přenosu. Pro tento účel je zvolen mechanismus okna zahlcení (*congestion window*), které vyjadřuje maximální množství dat, které lze v daném časovém okamžiku do sítě odeslat. Velikost okna se mění v závislosti na stavu přenosu – pokud probíhá přenos bez problémů (přicházejí potvrzení na všechny odeslané pakety), velikost okna se zvětšuje, poněvadž se předpokládá, že ještě nebyla naplněna dostupná kapacita linky. Zahlcení je detekováno zvýšenými ztrátami paketů, na které vysílající strana reaguje snížením velikosti okna a tím omezením množství vysílaných dat.

Vhodné použití: Protokol ARTP je vhodný především pro ty aplikace, které vyžadují spolehlivý přenos dat, avšak nevyžadují zachování jejich pořadí (buď pozdě přijaté datagramy ignorují nebo se s nimi vypořádají po svém, například bufferováním). Data jsou pak odesílána v samostatných (libovolně velkých) celcích, které jsou na straně příjemce zpracovávány nezávisle na ostatních. Jednou z konkrétních aplikací může být například přenos multimediálních dat ke kódování na straně serveru a přesun výsledku zpět na klienta.

Další informace:

- Podrobná funkční specifikace je součástí technické zprávy č. 11/2004 na <http://www.cesnet.cz/doc/techzpravy/>
- <http://www.fi.muni.cz/~xrebok/arp/>

¹ Identifikační čísla paketů mají sice velký, ale konečný rozsah. V případě, že by pakety neobsahovaly informaci o času své expirace, mohl by být za jistých okolností příliš zpožděný paket se starým identifikačním číslem pokládán za aktuální (kdyby přišel v okamžiku příjmu paketů s identifikačním čísly „nového kola“, která jsou blízká inkriminovanému, nebyl by „odchycen“ ani detekcí duplicit).

4.2 Vzorová implementace

Vzorová implementace protokolu ARTP je naprogramována formou knihovny v programovacím jazyce C (zdrojové kódy uvolněné pod BSD licenci lze nalézt na [4]). Aktuálně tak protokol ARTP spolupracuje s protokolem UDP, na pozdější spolupráci přímo se síťovým protokolem IP však jeho specifikace také pamatuje (tato funkčnost bude implementována v následujících verzích).

Vzorová implementace byla vytvořena a testována na operačním systému Linux (konkrétně distribuce Debian s jádrem 2.4.26 a Mandrake s jádrem 2.4.23). Protože je v maximální možné míře použito volání standardů POSIX a ISO C, je možno knihovnu s minimálními změnami provozovat i na jiných unixových operačních systémech (např. *BSD, IRIX, Solaris a dalších). Komentáře k funkcím jsou psány ve formátu pro *doxygen*, pomocí kterého je také vygenerována dokumentace (viz [4]).

Knihovna protokolu ARTP implementuje veškerou funkčnost navrženého protokolu. Splňuje klíčové požadavky, které na ni byly vzhledem k předpokladu, že se bude protokol ARTP neustále vyvíjet, kladeny – jednoduchost, strukturovanost a čitelnost zdrojových kódů (díky těmto vlastnostem jsou usnadněny její další případné úpravy). Z těchto důvodů byla v případech, kdy bylo nutno vybírat mezi čitelností a rychlostí kódu, zvolena právě čitelnost.

4.3 Testy

Efektivita vytvořené implementace protokolu ARTP byla testována nad pseudonáhodnými daty v zapojení *point-to-point* jak na síti o rychlosti 1 Gbit/s, tak na síti o rychlosti 100 Mbit/s. Použité počítače byly následující konfigurace:

- DELL PowerEdge 1600 SC
- 2 × Intel Xeon CPU 2.8 GHz
- paměť 1024 MB
- síťové karty Intel PRO/1000 32 bit/66 MHz,
- operační systém Linux Debian Woody 2.4.26.

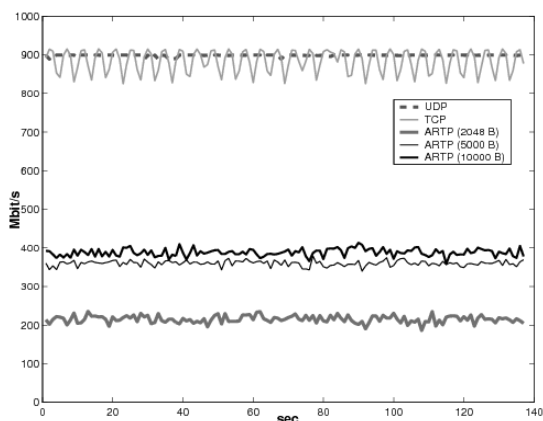
Test byl zaměřen na zjištění maximální přenosové rychlosti vytvořené implementace a jejím srovnáním s existujícím transportním protokolem (jako srovnávací protokol byl vybrán protokol TCP). V obou měřených konfiguracích (tj. 100Mbit síť a 1Gbit síť) byla také s využitím protokolu UDP změřena maximální propustnost linky².

Oba protokoly (tj. ARTP a TCP) byly testovány na třech velikostech přenášených datagramů – 2000 B, 5000 B a 10000 B. V případě protokolu TCP je však v následujících grafech zobrazen pouze naměřený průtok pro datagramy o velikosti 2000 B, poněvadž všechny tři testy dopadly téměř shodně a z důvodu přehlednosti nebyly do grafu zaneseny.

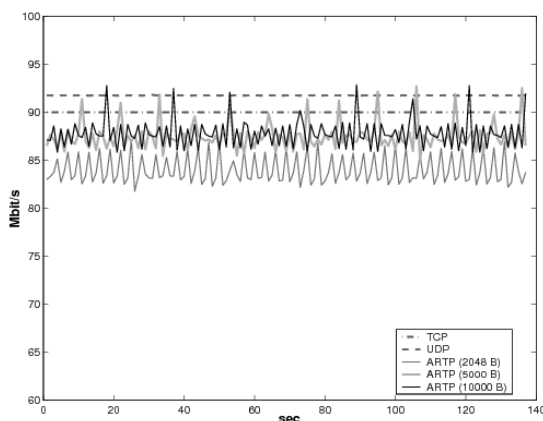
² Ve všech měřeních se přenosová rychlost počítala pouze z přenesených dat – do jejich objemu nebyly zahrnuty velikosti hlaviček protokolů nižších vrstev (UDP, IP, atp.) ani hlavičky testovaných protokolů.

Dgram.	1 Gbit/s (v Mbit)			100 Mbit/s (v Mbit)		
	UDP	TCP	ARTP	UDP	TCP	ARTP
2048B	898,46 ± 2,49	882,94 ± 30,15	215,24 ± 9,45	91,75 ± 0,00	90,00 ± 0,00	84,01 ± 1,47
5000B	-	881,63 ± 27,67	359,82 ± 7,18	-	89,53 ± 0,00	87,50 ± 1,43
10000B	-	881,27 ± 32,42	387,98 ± 9,72	-	89,17 ± 0,00	87,80 ± 1,48

Tabulka 1: Průměrné rychlosti testovaných protokolů se směrodatnou odchylkou naměřených hodnot



Obrázek 2: Srovnání rychlostí testovaných protokolů na 1Gbit/s síti



Obrázek 3: Srovnání rychlostí testovaných protokolů na 100Mbit/s síti

Z naměřených výsledků lze vyčíst, že vytvořená implementace protokolu ARTP je na 100Mbit síti konkurenceschopná, avšak na 1Gbit síti dosti zaostává. To je způsobeno řadou faktorů – především tím, že vypracovaná implementace je zaměřena na přehlednost a strukturovanost kódu, nikoli na efektivitu.

Z výsledků dosažených na 1Gbit síti pak vyplývá ještě jedna skutečnost – rychlost protokolu ARTP se mírně zvyšuje s velikostí přenášeného datagramu. Tento jev je způsoben menší manipulací s pamětí při alokaci a dealokaci paměťových bloků na přijímající straně protokolu (místo pěti sérií kroků nutných ke zpracování datagramu o velikosti 2000 B je v případě datagramu velikosti 10000 B provedena pouze jedna taková série).

5 Závěr

Po shrnutí charakteristických vlastností transportní vrstvy byly v příspěvku představeny některé existující transportní protokoly s různými vlastnostmi, které mohou být s výhodou využity v síťových aplikacích, jimž nejnámější a nejrozšířenější transportní protokoly TCP a UDP díky svým vlastnostem nevyhovují. V další části pak byl představen nově navržený transportní protokol ARTP určený pro použití ve vyvíjeném aktivním směrovači. K navrženému protokolu byla vytvořena i jeho vzorová implementace, která byla testována vůči nejnámějším transportním protokolům TCP a UDP. Ačkoli lze její výkon z hlediska vysokorychlostních sítí hodnotit jako průměrný, je dosažená rychlost pro širokou třídu aplikací zcela dostatečná. Pro použití ve vyvíjeném směrovači je totiž v první fázi vývoje hlavním požadavkem kladeným na všechny jeho vrstvy jejich plná funkčnost a čistota kódu, což implementovaná komunikační vrstva splňuje.

Další práce na protokolu a vytvořené implementaci se tak může ubírat především směrem k dosažení jeho většího výkonu na vysokorychlostních sítích. Toho by mohlo být dosaženo především následujícími úpravami:

- zefektivněním práce s pamětí a paměťových struktur použitých ve vytvořené implementaci,
- navržením a důkladným otestováním lepších způsobů manipulace s oknem zahlcení, (*congestion window*),
- zavedením protokolu do vrstvy jádra operačních systémů Linux/Unix.

6 Poděkování

Protokol ARTP byl vytvořen jako diplomová práce za podpory výzkumného záměru Optická síť národního výzkumu a její nové aplikace (MŠM 6383917201) zájmového sdružení Cesnet. Autor by chtěl dále poděkovat RNDr. Evě Hladké, Ph.D. a Mgr. Zdeňku Salvetovi za mnoho cenných rad a připomínek, a všem kolegům z Laboratoře pokročilých síťových technologií Fakulty informatiky Masarykovy univerzity v Brně.

7 Odkazy a literatura

- [1] <http://cbdd.wsu.edu/kewlcontent/cdoutput/TR502/page82.htm> – ISO OSI referenční model
- [2] <http://www.zvon.org/> – RFC dokumenty
- [3] <http://www.earchive.cz/> – archiv článků a přednášek Jiřího Peterky
- [4] <http://www.fi.muni.cz/~xrebok/artp/> – informace o protokolu ARTP
- [5] Hladká Eva, Salvét Zdeněk: An Active Network Architecture: Distributed Computer or Transport Medium, Heidelberg : Springer-Verlag Berlin, 2001.