

Trendy v monitorování vysokorychlostních počítačových sítí

Úkolem monitorování počítačových sítí je poskytnout provozovateli síte a náročným uživatelům se specifickými požadavky informace o operačním stavu síte, o výkonnostních charakteristikách a o možných bezpečnostních problémech. V tomto článku se zaměříme na současné trendy v monitorování vysokorychlostních sítí zejména s ohledem na sledování výkonnostních charakteristik pro náročné aplikace.

Aktivní a pasivní monitorování

Většinu monitorovacích metod lze rozdělit mezi aktivní a pasivní. Při *aktivním monitorování* posíláme do síte testovací pakety, které opět přijímáme v jiném místě síte. Tímto způsobem můžeme měřit například zpoždění při průchodu sítí, ztrátovost nebo dosažitelnou propustnost. Nevýhodou aktivního monitorování je přidaná zátěž do síte (zejména při měření propustnosti "hrubou silou" intenzivním datovým tokem), možné ovlivnění provozu uživatele a to, že měříme charakteristiky našich testovacích paketů, nikoliv charakteristiky provozu uživatele, které mohou být velmi odlišné. Je například obtížné měřit aktivně ztrátovost paketu v síti, protože ta velmi závisí na objemu a dynamice provozu, které jsou u skutečného provozu uživatele velmi odlišné od testovacích paketů, které si můžeme dovolit do síte posílat.

Při *pasivním monitorování* neposíláme do síte testovací pakety, ale vyhodnocujeme časové a objemové charakteristiky uživatelského provozu. Pasivní monitorování neovlivňuje uživatelský provoz a může sledovat charakteristiky, které jsou aktivním monitorováním nezjistitelné. Například jaký je objem a dynamika volné kapacity v síti, které aplikace uživatele mají největší nároky na kapacitu síte nebo zda v síti dochází k bezpečnostním útokům. Aktivní monitorování si lze tedy představit jako testovací sondu poslanou jednorázově nebo opakovaně do síte, zatímco pasivní monitorování je zpravidla trvale běžící pozorovatel dení na síti.

Kromě čisté aktivního nebo pasivního monitorování jsou i metody využívající kombinace obou přístupů (vhodné například pro měření ztrátovosti), metody zpracovávající data získaná z komponentů síťové infrastruktury (např. pomocí SNMP nebo protokolu Netflow) a měření sledující stav koncové stanice (např. pomocí rozhraní PAPI). V tomto článku se soustředíme na možnosti pasivního monitorování, které se pro radu aplikací jeví jako velmi atraktivní.

Hardwarová akcelerace

Problémem pasivního monitorování je potřeba zpracování velkého objemu dat v reálném case. Páteří linky současných sítí mají standardně kapacitu 10 Gb/s. Monitorování tak velkého objemu dat je potřeba rozdělit mezi hardware a software. Specializované *hardwarové monitorovací adaptéry* provádějí operace na nižších vrstvách komunikace v síti, jako je sledování časových a objemových charakteristik paketu a jejich klasifikace podle protokolu nebo jiných údajů. Potom následuje zpracování již menšího objemu dat na vyšších úrovních komunikace. Tím může být výpočet dlouhodobých statistik nebo rozpoznávání aplikací nebo možných bezpečnostních útoků podle obsahu vybraných filtrovaných paketů.

Potřebné monitorovací adaptéry lze dnes získat od několika výrobců. Několik typů programovatelných monitorovacích adaptéru bylo také vyvinuto v projektu Liberouter sdružení CESNET a v mezinárodním projektu SCAMPI. Adaptéry jsou standardně osazeny programovatelným hradlovým polem (FPGA), ve kterém jsou implementovány monitorovací

funkce adaptéru. Adaptéry se liší zejména v typu síťového rozhraní (Ethernet od 10 Mb/s do 10 Gb/s, ATM, PoS) a v monitorovacích funkcích implementovaných v hardware. Většina adaptéru umí rozdelovat pakety do tříd podle obsahu jejich hlaviček, přidělovat paketům přesné časové značky z vnějšího zdroje času a počítat statistiky pro jednotlivé třídy. Základní srovnání dostupných adaptéru je v Tabulce 1. Obecně lze říci, že karty od firem Endace a Force10 kladou důraz na 100% propustnost na dané linkové technologii při všech délkách paketu včetně hardwarové klasifikace paketu. Karty Napatech a COMBO naproti tomu kladou důraz na větší rozsah monitorovacích funkcí implementovaných přímo na karte s možností jejich programování.

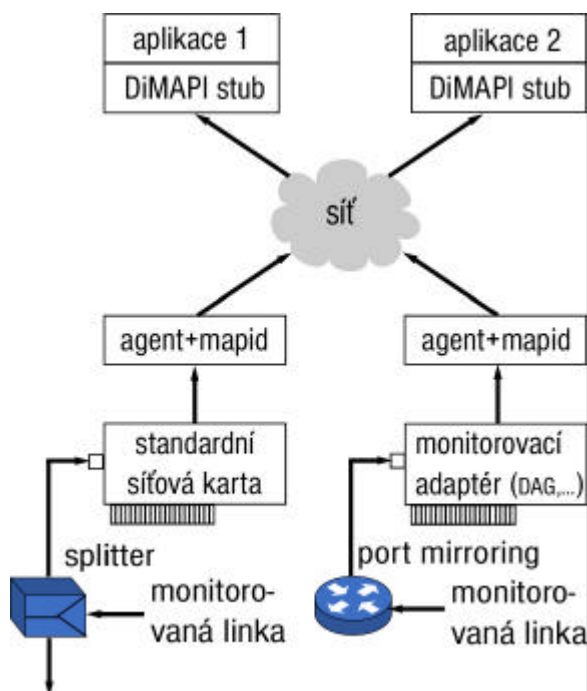
	Endace (karty DAG)	Force10 (P-Series)	Napatech	COMBO
Rozhraní	2x 1Gb, 1x 10Gb, 1x OC-48 a další	2x 1Gb, 2x 10Gb	4x 1Gb, 8x 1Gb, 1x 10Gb	4x 1Gb (počet využitých portů závisí na typu firmware)
Transceivery	výmenné SFP a pevné TX pro 1Gb, pevné 10GBaseLX pro 10Gb	výmenné SFP pro 1Gb, výmenné XPAK pro 10Gb	výmenné SFP pro 1Gb, pevné 10GBaseLX pro 10Gb	výmenné SFP a pevné TX pro 1Gb
Sběrnice	PCI-X 64-bit/133MHz	Dodáváno jako celé zařízení pro montáž do racku	PCI-X 64-bit/133MHz	PCI-X 64-bit/64MHz
Funkce v hardware	Filtrace a klasifikace paketu, podrobná u 1Gb karet s koprocесorem, jednodušší u 10Gb karet	Filtrace a klasifikace paketu, prohledávání payloadu	K dispozici jsou dva druhy karet - "protocol and traffic analysis adapter" a "programmable adapter", přesný popis monitorovacích funkcí není k dispozici	Filtrace a klasifikace paketu, smplování, statistiky a generování Netflow záznamu
Informace	www.endace.com	www.force10networks.com	www.napatech.com	www.liberouter.org

Tabulka 1: Srovnání monitorovacích adaptéru

Monitorovací adaptér je možné připojit k monitorované lince buď pomocí *optického rozbočovacího* (splitteru) nebo pomocí portu na paketovém prepínací nebo smerovací, který je nastaven pro kopírování všech paketů (*port mirroring*) z monitorované linky. Optický rozbočovač je levnější a zároveň zachová původní časové charakteristiky provozu. Vyžaduje ale rozpojení monitorované linky, vkládá do ní další útlum a musíme použít monitorovací adaptér s rozhraním odpovídajícím fyzické vrstvě dané linky. Monitorovací port umožňuje použít jiný typ rozhraní monitorovacího adaptéru (například Gigabit Ethernet pro méně zatíženou linku PoS 2.5 Gb/s), ale smerovace mají obvykle omezení na počet a typ portu pro které lze kopírování paketů nastavit.

Integrace infrastruktury - architektura LOBSTER

V síti obvykle potřebujeme nasadit několik různých monitorovacích aplikací pracujících s různými typy monitorovacích adaptéru zahrnujícími různý stupeň hardwarové podpory monitorování. Aby monitorovací infrastruktura pracovala v tomto heterogenním prostředí a byla snadno rozšiřitelná, bylo v rámci projektu SCAMPI [1] vytvořeno prostředí *MAPI* (Monitoring Application Programming Interface) pro snadnou tvorbu přenositelných monitorovacích aplikací. V následném projektu LOBSTER [2] bylo toto prostředí rozšířeno o podporu distribuovaného monitorování v *DiMAPI* (Distributed MAPI). Architektura LOBSTER je znázorněna na obrázku 1.



Obr. 1: Architektura LOBSTER

Každá aplikace je preložena s knihovnou DiMAPI stub. Více aplikací může běžet současně na stejném nebo na různých počítačích a mohou při tom sdílet monitorovací adaptéry, které mohou být instalovány na vzdálených monitorovacích stanicích v různých uzlech sítě. Knihovna DiMAPI stub předává přes síť v rámci TCP spojení požadavky na monitorovací funkce do vlastní implementace DiMAPI v démonu `mapid`. Síťovou komunikaci zprostředkovává démon `agent`. Je kladen důraz na to, aby ze vzdálených uzlů s monitorovacími adaptéry byly přenášeny již pouze výsledky monitorovacích funkcí, nikoliv vlastní pakety. Prostředí DiMAPI je k dispozici na Internetu [3] a je možné jej začít používat s běžnými síťovými kartami, speciální monitorovací adaptéry lze nasadit později.

Monitorovací funkce

Aplikace vždy nejprve otevře jeden nebo více *toků dat* (flow). Každý tok z počátku představuje všechny pakety přicházející do zadané množiny monitorovacích adaptéru (scope). Potom aplikace nasadí na každý tok posloupnost *monitorovacích funkcí*. Tyto funkce mohou provádět například klasifikaci paketu do tříd, samplování, vyhledávání reťezcu v paketech, anonymizaci údajů v hlavičkách, výpočet objemových a časových statistik, atd. DiMAPI automaticky využije podporu monitorovacích funkcí zabudovanou v jednotlivých použitých hardwarových monitorovacích adaptérech. Pokud potřebné funkce nejsou v daných adaptérech k dispozici nebo je nelze použít vzhledem k poradi nebo počtu monitorovacích funkcí nasazených na toky dat, potom DiMAPI implementuje příslušné funkce softwarově. Pro aplikaci je tento proces zcela transparentní. Uživatel může definovat svoje vlastní monitorovací funkce a rovněž může přidat podporu pro nový typ monitorovacího adaptéru s určitou zabudovanou hardwarovou podporou.

Následující fragment kódu vytvoří tok dat ze dvou vzdálených monitorovacích adaptéru (běžné síťové karty a karty DAG) a pomocí dvou monitorovacích funkcí sleduje počet paketu přicházejících do zadané subsítě:

```
fd=mapi_create_flow("pc1:eth1, pc2:/dev/dag0");
fid0=mapi_apply_function(fd, "BPF_FILTER", "dst net 10.0.2.0/24");
fid1=mapi_apply_function(fd, "PKT_COUNTER");
```

Inspekce paketu

Jedním ze žadanych úkolu monitorování je detekce aplikací, které používají dynamicky volené porty (nikoliv pevne pridelená čísla portu). Tento úkol vyžaduje použití kombinaci hlavickové filtrace s hledáním retezcu v paketech. Za tímto účelem obsahuje prostředí DiMAPI samostatnou knihovnu tracklib pro detekci aplikacních protokolů. K dispozici je detekce známých aplikací pro sdílení souboru jako Gnutella, BitTorrent, atd. DiMAPI se opět snaží využít hardwarové podpory v monitorovacích adaptérech, je-li to možné. Uživatel může přidat svoje vlastní funkce do knihovny tracklib pro detekci dalších aplikací.

Následující fragment kódu sleduje objem dat šířených aplikací Gnutella ze zadané subsítě:

```
fd=mapi_create_flow("pc3:/dev/dag0");
fid0=mapi_apply_function(fd, "BPF_FILTER", "src net 10.0.3.0/24");
fid1=mapi_apply_function(fd, "TRACK_GNUTELLA");
fid2=mapi_apply_function(fd, "BYTE_COUNTER");
```

Sledování zátěže linky

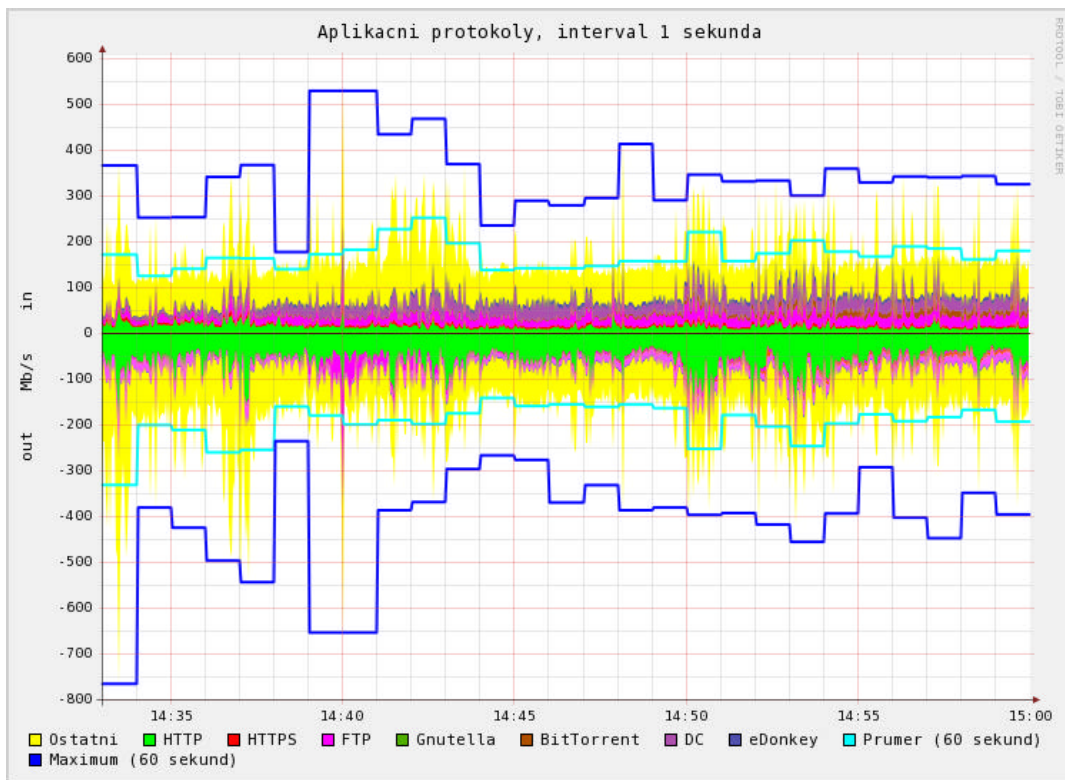
Jedním z nejdůležitějších výkonnostních parametru linky resp. trasy v síti je její aktuální zátěž nebo naopak volná kapacita. Tuto informaci potřebujeme při hledání příčin nízké propustnosti, pro plánování infrastruktury síte a případně pro tarifkaci uživatele.

Aktivní zátěžové testy například programem `iperf` měří *dosažitelnou propustnost*. To je užitečná charakteristika, je však třeba si uvědomit, že měřením dochází k ovlivnění provozu uživatele, který je dnes převážně přenášen protokolem TCP, který je tzv. elastický, t.j. jednotlivé toky spolu soupeří o *instalovanou kapacitu* trasy. Je to tedy metoda intruzivní a měří odlišnou charakteristiku od *volné kapacity* síte, která je doplnkem *využité kapacity* trasy do instalované kapacity.

Aktivní nezážžové testy například programy `pathrate` nebo `pathload` se snaží odhadnout instalovanou nebo volnou kapacitu trasy pomocí malého počtu paketu odeslaných v přesných časových rozestupech, kde změna těchto rozestupů po průchodu trasou je analyzována. Existuje rada programu tohoto typu, většinou však poskytují nepřesné výsledky ve vysokorychlostních sítích s bohatou dynamikou uživatelského provozu.

Často používanou metodou sledování využití kapacity linky je periodické čtení čítacu odeslaných a přijatých bajtu na jednotlivých rozhraních směrovacím protokolem SNMP. Tato metoda je spolehlivá, umožňuje však sledovat jen celkový objem provozu bez rozdělení na protokoly a aplikace a nemožuje detekci krátkodobých špic. Smerovace totiž standardně aktualizují svoje čítace s periodou nekolika sekund. Nejkratší rozumný interval přes který můžeme počítat průmernou zátěž linky je tak v řádu desítek sekund.

Pasivní monitorování umožňuje sledovat jak je kapacita linky využívána jednotlivými protokoly a aplikacemi a to i ve velmi krátkých intervalech. Příklad grafického znázornění využití kapacity linky různými aplikacemi s periodou jedna sekunda a celkovým průmerným a maximálním zatížením s periodou 60 sekund je na obrázku 2.



Obr. 2: Vzdálené monitorování objemu dat jednotlivých protokolů

Další aplikace, které lze realizovat pasivním monitorováním v prostředí DiMAPI zahrnují například měření ztrátovosti paketu, detekci šíření viru, detekci různých druhů síťových útoků a anomálií v provozu.

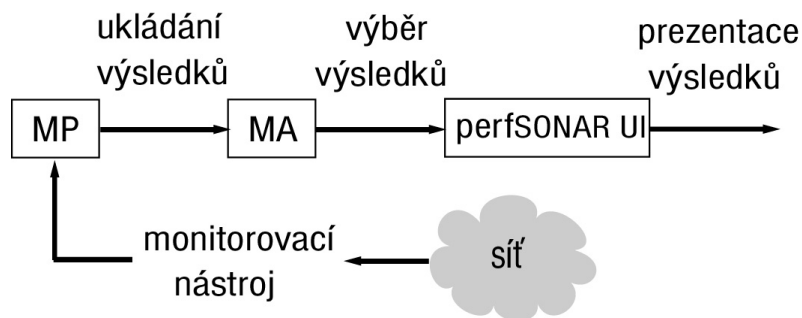
Integrace měření - architektura perfSONAR

Při řešení výkonnostních problémů je často potřeba provést měření řady různých charakteristik ve více sítích přes které probíhá komunikace a provést korelaci těchto měření. Evropské národní sítě pro výzkum a vzdělávání (NREN - National Research and Educational Network) jsou propojené společnou sítí GÉANT2. V České republice provozuje síť NREN sdružení CESNET. V rámci projektu GN2, který buduje síť GÉANT2 je vytvářen systém *perfSONAR* (performance-focused service-oriented network monitoring architecture) [4] pro integraci různých měření výkonnostních charakteristik. Systém je založen na principu web services, jde tedy o množinu nezávisle běžících služeb komunikujících posíláním zpráv ve formátu XML. Jsou k dispozici následující služby:

- Measurement Point (MP) - provádí měření určité charakteristiky v určitém místě
- Measurement Archive (MA) - ukládá výsledky měření
- perfSONAR UI - uživatelské rozhraní pro prezentaci výsledku
- Lookup Service (LS) - registruje umístění ostatních služeb
- Authentication Service (AS) - overuje identitu uživatele

Jednoduchou komunikaci částí služeb znázorňuje obrázek 3. Služba MP provádí měření a ukládá výsledky do databáze pomocí služby MA, která se registruje u služby LS. Služba perfSONAR UI zjistí u služby LS umístění služby MA, ze které přečte požadované výsledky a prezentuje je uživateli. Zprávy ve formátu XML posílané mezi službami jsou většinou

tvoreny ze dvou částí. První částí jsou tzv. *metadata*, která popisují jaká data jsou požadována nebo jaká data jsou naopak posílána. Druhou volitelnou částí jsou vlastní *data*. Metadata sestávají ze *subjektu*, který určuje místo monitorování (monitorovací stanice a její rozhraní) a z *parametru*, které určují podmínky platné pro data, například časové období monitorování, sledované subsítě, argumenty volaných monitorovacích nástrojů a podobně. Základní struktura zpráv odpovídá doporučení pracovní skupiny NMWG (Network Measurements Working Group) sdružení GGF (Global Grid Forum) [5]. Příklad části XML zprávy zahrnující metadata a data je na obrázku 4. Systém perfSONAR je k dispozici na Internetu [4], protože ale projekt GN2 ještě probíhá, implementace není dosud úplná.



Obr. 3: Komunikace v systému perfSONAR

```

<nmwg:metadata id="meta1">
  <netutil:subject id="iusub1">
    <nmwgt:interface>
      <nmwgt:ifAddress type="ipv4">192.168.1.1</nmwgt:ifAddress>
      <nmwgt:ifName>Te1/1</nmwgt:ifName>
      <nmwgt:direction>in</nmwgt:direction>
    </nmwgt:interface>
  </netutil:subject>
  <nmwg:eventType>utilization</nmwg:eventType>
</nmwg:metadata>

<nmwg:metadata id="meta2">
  <select:subject id="iusub2" metadataIdRef="meta1"/>
  <select:parameters id="param1">
    <nmwg:parameter name="startTime">1124250480</nmwg:parameter>
    <nmwg:parameter name="endTime">1124250840</nmwg:parameter>
    <nmwg:parameter name="consolidationFunction">AVERAGE</nmwg:parameter>
    <nmwg:parameter name="resolution">60</nmwg:parameter>
  </select:parameters>
  <nmwg:eventType>select</nmwg:eventType>
</nmwg:metadata>

<nmwg:data id="data1" metadataIdRef="meta2">
  <nmwg:datum value="12345" timeValue="1124250481" timeType="unix" />
  <nmwg:datum value="12349" timeValue="1124250482" timeType="unix" />
  <!-- ... -->
  <nmwg:datum value="32345" timeValue="1124250839" timeType="unix" />
</nmwg:data>
  
```

Obr. 4: Zpráva XML systému perfSONAR

[1] Projekt SCAMPI - <http://www.ist-scampi.org>

[2] Projekt LOBSTER - <http://www.ist-lobster.org>

[3] DiMAPI - <http://mapi.uninett.no>

[4] perfSONAR - <http://www.perfsonar.net>

[5] GGF (Global Grid Forum) - <http://www.gridforum.org>

Dr. Ing. Sven Ubik, <ubik@cesnet.cz>, CESNET, z.s.p.o., Zikova 4, Praha 6, 16000.