# ارائه یک روش جدید تولید ترافیک روی شبکه های کامپیوتری جهت ارزیابی کارآیی آن

احمد تاجدینی

دانشکده ریاضی، گروه علوم کامپیوتر

دانشگاه تبریز (1390)

mehrta.cs@gmail.com

]جهت ارائه به سرکار خانم دکتر لیلی محمدخانلی،

درس شبکه های کامپیوتری[

# چکیده

یک راه برای ارزیابی کیفیت انتقال اطلاعات در شبکه های کامپیوتری ایجاد یک ترافیک (سنگین) بین دو نقطه مختلف شبکه و سنجیدن نرخ از دست رفتن اطلاعات (Loss Rate) است. در این روش یک نرم افزار تولید کننده ترافیک1 روی یکی از گره های شبکه و نرم افزار دیگری جهت دریافت داده های ارسال شده روی گره دیگری از شبکه نصب می شود. بعد از ارسال میزان مشخصی داده توسط نرم افزار تولید کننده ترافیک، نرخ از دست رفتن اطلاعات محاسبه می شود (نسبت تعداد بسته های دریافت نشده به تعداد کل بسته های ارسال شده). اگر نرخ از دست رفتن اطلاعات زیاد باشد نشان دهنده وجود مشکلی در زیرساخت شبکه است که باید به آن رسیدگی شود. در این نوشتار روشی را برای تولید ترافیک و ارزیابی کارایی شبکه ارائه می کنیم که در آن نیازی نیست دو نرم افزار مجزا روی دو گره مختلف از شبکه نصب شود، بلکه فقط با یک نرم افزار به سمت هر کامپیوتر دلخواه ترافیک ایجاد می کنیم و کارآیی شبکه را ارزیابی می کنیم.

1. Traffic Generator

# 1 مقدمه

در شبکه های کامپیوتری آنچه که روی کارآیی شبکه تاثیر اساسی می گذارد سخت افزار و نرم افزاری است که خدمات شبکه را فراهم می آورند. معمولا در شبکه هایی که انتظار می رود در شرایط مختلف کارآیی بالایی داشته باشند، قبل از عملیاتی شدن شبکه میزان کارآیی آن توسط ابزارهای مختلفی که برای این منظور وجود دارد سنجیده می شود. یک دسته از این ابزار ها، ابزارهای تولید کننده ترافیک نام دارند. این ابزارها که عمدتا به صورت نرم افزاری موجود می باشند یک ترافیک با الگوی خاص را بین دو گره مختلف شبکه ایجاد می کنند. معمولا هدف از این کار بررسی کارآیی سویچ ها و مسیریاب های میانی در شرایطی که شبکه دچار ازدحام شده است می باشد. یک فاکتور مهم که در شبکه های کابلی نشان دهنده وجود ازدحام است، نرخ از دست رفتن بسته است(در شبکه های بیسیم نرخ از دست رفتن بسته الزاما نشان دهنده وجود ازدحام در شبکه نیست[1]). برای انجام این کار نرم- افزار تولید کننده ترافیک و گیرنده ترافیک را روی دو انتهای مسیری از شبکه که مایلیم کارآیی سخت افزار یا نرم افزار موجود در آن مسیر را ارزیابی کنیم نصب می کنیم (شکل 1). نرم افزار تولید کننده ترافیک میزان مشخصی داده را به سمت گیرنده ارسال می کند2. بعد از به پایان رسیدن ارسال اطلاعات به سادگی می توانیم نرخ از دست رفتن بسته3 (PLR) را محاسبه کنیم:

معمولا به سه دلیل عمده مایلیم که روی شبکه های کامپیوتری ترافیک (مصنوعی) ایجاد کنیم:

1. بررسی میزان کارآیی سویچ ها و مسیریاب های شبکه

2. محتوای اطلاعات ارسالی اهمیتی ندارد.

3. Packet Loss Rate, PLR

2. بررسی میزان کارآیی دیوارهای آتش4 نصب شده

3. بررسی میزان کارآیی یک پیاده سازی از یک پروتکل خاص

شکل 1: ایجاد ترافیک از گره 1 (تولید کننده ترافیک) به گره 2 (گیرنده ترافیک)

در رابطه با مورد اول، اگر ترافیک ایجاد شده روی شبکه بیش از توان مسیریاب های شبکه باشد معمولا در مسیریاب ها سرریز بافر5 رخ می دهد و مسیریاب به ناچار تعدادی از بسته ها را دور می ریزد. بنابراین باید اطمینان حاصل کنیم که مسیریاب های شبکه می توانند چنین ترافیکی را تحمل کنند. بررسی مورد دوم نیز بسیار مهم است. اگر دیوار آتش به صورت کارآیی طراحی و پیاده سازی نشده باشد، ممکن است زمان پردازش بسته ها طولانی شود و تعدادی از بسته ها از دست برود (مثلا به دلیل پر شدن بافر). در مورد سوم نیز ممکن است یک برنامه نویس شبکه برای ارزیابی کارآیی یک نرم افزار تحت شبکه یا یک پیاده سازی از پروتکلی خاص نیاز به ترافیک شبکه داشته باشد.

در مواردی که مایلیم میزان تحمل شبکه در برابر ترافیک سنگین را بررسی کنیم، نرم افزار را به گونه ای پیکربندی می کنیم که بیشترین ترافیک را روی شبکه ایجاد کند. این مدل تولید ترافیک، مدل حریص6 نامیده می شود. همچنین برای تولید ترافیک می توانیم از الگوهای مختلف آماری نیز استفاده کنیم. به عنوان مثال می توان از توزیع احتمال پواسون به منظور الگوی تولید ترافیک استفاده کرد.

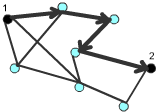
4. Firewall

5. Buffer Overflow

6. Greedy Source Model

# 2 محدودیت های ابزارهای موجود

ابزارهای موجود تولید کننده ترافیک به صورت دو تکه طراحی شده اند. یک قسمت از نرم افزار در یک سوی شبکه (تولید کننده ترافیک) و قسمت دیگر آن در سوی دیگر شبکه نصب می شود. همین موضوع باعث ایجاد محدودیت هایی می شود. اولین مشکل احتمال در دسترس نبودن کامپیوتری از شبکه است که مایلیم به سمت آن ترافیک ایجاد کنیم. مثلا ممکن است کامپیوتر مقصد یک کامپیوتر دور دست7 باشد یا اینکه مجوزهای لازم برای نصب نرم افزار روی آن را نداشته باشیم. در این حالت نمی توان نرم افزار گیرنده ترافیک را روی آن نصب کرد.

مشکل دوم ممکن است به دلیل عدم تطابق پلتفرم ها رخ دهد. برای مثال فرض کنید شبکه ای داریم که سیستم عامل نصب شده روی سرویس دهنده8 آن lInux و سیستم عامل نصب شده روی سرویس گیرنده ها9 windows باشد. اگر مایل به ایجاد ترافیک از سمت یکی از سرویس گیرنده ها به سمت سرویس دهنده باشیم، احتمالا با مشکل نصب نرم افزار روی کامپیوتر ها مواجه می شویم، زیرا ابزارهایی که در محیط سیستم عامل windows اجرا می شوند روی سرویس دهنده نصب نمی شوند و همچنین ابزارهایی که در محیط سیستم عامل linux اجرا می شوند روی سرویس گیرنده ها نصب نمی شوند. در این شرایط باید ابزارهایی پیدا کنیم که قسمت تولید کننده ترافیک آن روی سیستم عامل windows و قسمت گیرنده آن روی سیستم عامل linux قابل اجرا باشد.

البته اگر هدف صرفا تولید ترافیک روی شبکه باشد و آگاهی از اینکه بسته های ارسالی به دست گیرنده می رسند یا خیر اهمیت چندانی نداشته باشد می توانیم از سمت فرستنده یک ترافیک UDP تولید کنیم. در این حالت نیازی به نصب نرم- افزار خاصی روی گره مقصد نیست زیرا پروتکل udp یک پروتکل بدون اتصال است و برای ارسال دیتاگرام های udp نیازی به ایجاد هماهنگی و ایجاد اتصال بین طرفین وجود ندارد.

7. Remote Computer

8. Server

9. Client

همانطور که ذکر شد طراحی نرم افزار تولید کننده ترافیک به صورت دوتکه مشکلاتی را اینجا می کند. در ادامه با استفاده از دو پروتکل رایج شبکه های کامپیوتری روشی را ارائه می کنیم که در آن با استفاده از یک نرم افزار به سمت هر گره دلخواه در شبکه ترافیک ایجاد می کنیم و نرخ از دست رفتن اطلاعات را محاسبه می کنیم بدون اینکه نیاز به نصب نرم افزار روی کامپیوتر مقصد باشد.

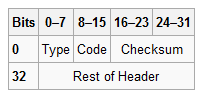
# 3 روش پیشنهادی برای تولید ترافیک

یک راه برای برطرف کردن محدودیت هایی که به آن اشاره شد این است که در سمت گیرنده هیچ نرم افزاری نصب نکنیم و فرستنده نیز ترافیک udp ایجاد کند. در این حالت مسیریاب ها بسته های تولید شده را به سمت گره مقصد هدایت می کنند ولی چون هیچ نرم افزاری در گره مقصد منتظر این بسته ها نیست، کامپیوتر مقصد آن ها را دور می ریزد. با استفاده از این روش می توان روی شبکه ترافیک ایجاد کرد ولی نمی توان نرخ از دست رفتن بسته ها را محاسبه کرد زیرا نمی دانیم چند دیتاگرام udp به سلامت به دست گیرنده رسیده و چند دیتاگرام از بین رفته است.

روشی که در اینجا پیشنهاد می شود نیازی به نصب نرم افزار در سمت گیرنده ندارد و از پروتکل icmp برای اندازه گیری نرخ از دست رفتن بسته ها استفاده می کند. قبل از ارائه این روش یادآوری می کنیم که پروتکل icmp یک پروتکل کنترلی است که برای ارسال پیغام (مانند پیغام خطا) بین گره های شبکه بکار می رود. یکی از کاربرد های پروتکل icmp در echo است، بدین صورت که اگر یک بسته Echo Request10 به سمت هر گره ای در شبکه ارسال شود آن گره موظف است در پاسخ یک بسته Echo Reply11 به سمت مبدا ارسال کند. اگر بسته Echo Request حاوی Payload باشد، گره مقصد باید Payload آن را عینا در بسته Echo Reply کپی کند و سپس آن را ارسال کند.

10. یک بسته ICMP با فیلدهای Type=8 و Code=0

11. یک بسته ICMP با فیلدهای Type=0 و Code=0



شکل 2: سرآیند یک بسته ICMP

معمولا از Echo به منظور تخمین RTT و آگاهی از در دسترس بودن یا نبودن یک گره خاص از شبکه استفاده می شود.

# 3.1 شرح عملکرد

برای ایجاد ترافیک روی شبکه و تخمین نرخ از دست رفتن اطلاعات به صورت زیر عمل می کنیم:

1. ابتدا به سمت گره مقصد یک ترافیک UDP ایجاد می- کنیم. با توجه به هدفی که داریم می توانیم ترافیک سنگین یا سبک ایجاد کنیم. برای مثال اگر مایلیم مسیریاب های شبکه را آزمایش کنیم، می توانیم یک ترافیک سنگین ایجاد کنیم.

2. به صورت همزمان یک ترافیک نسبتا سبک از بسته های Echo Request به سمت مقصد ایجاد می کنیم. گره مقصد هر بسته Echo Request ای را که دریافت می کند موظف است در پاسخ به آن یک بسته Echo Reply ارسال کند. نرم افزار تولید کننده ترافیک با شمارش بسته های Echo Reply ای که دریافت می کند می تواند نرخ از دست رفتن بسته ها در شبکه را محاسبه کند. برای اینکه بتوانیم یک ترافیک ICMP ایجاد کنیم بهتر است پشت سر هم بسته های Echo Request ارسال کنیم و منتظر دریافت بسته های Echo Reply نمانیم. معمولا با توجه به وسعت شبکه، برای هر بسته می توانیم مدت زمانی بین 0.5 تا 3 ثانیه منتظر بمانیم تا بسته Echo Reply مربوط به آن را دریافت کنیم. اگر در طول این مدت بسته ای دریافت نکردیم، فرض می کنیم که بسته از بین رفته است.

# 3.2 محاسبه نرخ از دست رفتن بسته ها

همانطور که قبلا ذکر شد، ترافیک UDP صرفا وظیفه ایجاد ترافیک (و ایجاد ازدحام) در شبکه را دارد و در واقع یک وضعیت "پر ازدحام" را در شبکه شبیه سازی می کند. برای محاسبه نرخ از دست رفتن بسته ها (PLR) نیز از پروتکل ICMP کمک گرفته شده است.

برای محاسبه PLR ابتدا متغیرهای زیر را تعریف می کنیم:

PKTsent: تعداد کل بسته های Echo Request ای که نرم افزار فرستاده است.

PKTok: تعداد کل بسته های Echo Reply ای که نرم افزار دریافت کرده است.

PKTlost: تعداد کل بسته هایی که برای آن ها Echo Reply دریافت نشده است.

با توجه به این تعاریف داریم:

PKTsent = PKTok + PKTlost (3.1)

باید توجه داشت که به ازای هر بسته Echo Request که ارسال می شود، اگر بسته به دست گیرنده برسد یک بسته Echo Reply نیز در پاسخ به آن ارسال خواهد شد. پس اگر تمامی بسته های Echo Request به مقصد برسند و همچنین تمامی بسته های Echo Reply به دست گره مبدا (نرم افزار تولید کننده ترافیک) برسد، تعداد کل بسته هایی که ارسال و دریافت می شود برابر 2 \* PKTlost خواهد بود. نرم افزار تولید کننده ترافیک به دو دلیل ممکن است جواب بسته های Echo Request ای که ارسال کرده است را دریافت نکند. حالت اول این است که بسته Echo Request به مقصد نرسد و حالت دوم این است که بسته Echo Request به گیرنده برسد ولی Echo Reply آن به دست نرم افزار نرسد. برای محاسبه PLR باید هر دو حالت را لحاظ کرد. کران پایین مقدار PLR وقتی رخ می دهد که دلیل گم شدن (همه) بسته ها این باشد که Echo Request به مقصد رسیده باشد ولی Echo Reply ها به دست نرم افزار نرسد (یعنی بسته ها در مسیر برگشت گم شوند). در این حالت داریم:

(3.2)

کران بالای plr وقتی رخ می دهد که دلیل گم شدن (همه) بسته ها این باشد که Echo Request به مقصد نرسیده باشد. طبعا در این حالت هیچ Echo Reply ای نیز تولید نمی شود. پس تعداد کل بسته های ICMP که در شبکه تولید می شود برابر خواهد بود. در نتیجه کران بالای PLR برابر مقدار زیر خواهد بود:

(3.3)

از (3.2) و (3.3) رابطه زیر را بدست می آوریم:

(3.4)

در اکثر مواردی که مسیر رفت با مسیر برگشت یکی است، احتمال اینکه بسته Echo Request در مسیر رفت گم شود با احتمال اینکه Echo Reply در مسیر برگشت گم شود تقریبا برابر است. بنابراین برای تخمین PLR می توانیم میانگین کران پایین و بالای رابطه (3.4) را محاسبه کنیم:

(3.5)



برای مثال اگر 100 بسته Echo Request ارسال کنیم و برای 90 بسته Echo Reply دریافت کنیم و برای 10 بسته پاسخی دریافت نکنیم، خواهیم داشت:

PKTsent = 100

PKTok = 90

PKTlost = 10

با توجه به رابطه (3.4) داریم:

در نتیجه

و با استفاده از تخمین (3.5) داریم:

# 4 نتیجه گیری

روشی که برای تولید ترافیک در شبکه های کامپیوتری ارائه شد بر خلاف روش های فعلی می تواند بدون نصب نرم افزار در سمت گره مقصد نرخ از دست رفتن بسته ها را محاسبه کند. این روش ایجاد ترافیک علاوه بر اینکه محدودیت روش- های فعلی را برطرف می کند، برای استفاده نیز سهولت بیشتری دارد زیرا می توان به آسانی به سمت هر گره شبکه ترافیک ایجاد کرد و PLR را سنجید. در شبکه های LAN کابلی نرخ از دست رفتن اطلاعات در شبکه با میزان ازدحام موجود در شبکه رابطه مستقیمی دارد، به این معنی که اگر مقدار PLR بالا باشد یعنی در شبکه ازدحام وجود دارد و برعکس. در مورد شبکه های بیسیم نمی توانیم چنین استدلالی بیاوریم چون ممکن است نرخ بالای خطا به دلیل خطا در ارسال بسته ها (در لایه فیزیکی) باشد[1].

**منابع:**

[1] Hassan, Mahbub(2004). High Performance TCP/IP Networking, Concepts, Issues, and Solutions. The University of New South Wales

[2] Dostálek, Libor(2006). Understanding TCP/IP A clear and comprehensive guide to TCP/IP protocols. Birmingham, Mumbai