

## EVALUATING QOS IN NETWORKS WITH ACTIVE QUEUE MANAGEMENT ALGORITHMS

Awos Kh. Ali<sup>1\*</sup>

<sup>1\*</sup>Department of Computer Science, Education College for Pure Science, University of Mosul, Mosul, Iraq

E-mail:<sup>1\*</sup> [a.k.ali@uomosul.edu.iq](mailto:a.k.ali@uomosul.edu.iq)

(Received January 16, 2022; Accepted February 22, 2022; Available online March 01, 2022)

DOI: [10.33899/edusj.2022.132790.1211](https://doi.org/10.33899/edusj.2022.132790.1211), © 2022, College of Education for Pure Science, University of Mosul.

This is an open access article under the CC BY 4.0 license (<http://creativecommons.org/licenses/by/4.0/>)

### ABSTRACT

Packets queuing and scheduling in network routers is a key point of overall network performance. Many applications, especially applications that require Quality of Services (QoS) need techniques to pass their packets throughout routers and control and/or avoid congestions in highly congested routes. Therefore, many Active Queue Management (AQM) algorithms have been developed to avoid or control congestion in routers and provide fairness among traffic flows. This paper provides an extensive evaluation performance analysis of three well-known AQM algorithms including RED, REM and traditional Drop-Tail with QoS application requirements. The evaluation performance is conducted by employing network simulator version 2 (NS2). The network performance is measured with Voice over Internet Protocol (VoIP) traffic and three performance metrics including throughput, latency, and PSD (Probability of Sequential Drop). The analysis shows no AQM algorithm achieves all the VoIP QoS requirements, A new AQM is needed to fulfil QoS requirements and manage queue to handle unresponsive flows.

**Keyword:** AQM, RED, REM, Drop-Tail

### الخلاصة

يعد وضع أجزاء البيانات في طوابير وجدولتها في أجهزة توجيه الشبكة نقطة أساسية لأداء الشبكة بشكل عام. تحتاج العديد من التطبيقات، خاصة التطبيقات التي تتطلب جودة الخدمات (QoS)، إلى تقنيات لتبرير بياناتها عبر أجهزة التوجيه والتحكم و / أو تجنب اختناق المسارات شديدة الازدحام. لذلك، تم اقتراح العديد من خوارزميات إدارة الطوابير النشطة (AQM) للتحكم في الازدحام في أجهزة التوجيه وتوفير العدالة بين تدفقات حركة المرور. تقدم هذه الورقة تحليلاً شاملاً لأداء التقييم لثلاث خوارزميات AQM معروفة جيداً والتي تتضمن RED و REM و Drop-Tail مع متطلبات الـ QoS. يتم إجراء تقييم الأداء باستخدام الإصدار الثاني من محاكي الشبكة (NS2). يتم قياس أداء الشبكة من خلال تطبيق تفاعلي للهاتف عبر بروتوكول الإنترنت (VoIP) وثلاثة مقاييس للأداء بما في ذلك، الإنتاجية، مدة التأخير، و PSD (احتمالية الفقدان المتسلسل للبيانات). يظهر التحليل عدم وجود خوارزمية AQM تحقق جميع متطلبات جودة خدمة VoIP، هناك حاجة إلى خوارزمية AQM جديدة تستطيع تلبية متطلبات QoS وإدارة طوابير البيانات للتعامل مع التدفقات غير المستجيبة.

### 1. INTRODUCTION

The demand for Internet services is growing rapidly and continuously, which makes servers experience a bottleneck problem leading to increased latency and drop rate due to the increase in demand for the services, this could have an impact on interactive application performance on those with QoS requirements [1]. Hundreds or even thousands of nodes are connected together and generate heavy network traffic, which may cause a network congestion problem [2]. Most Internet applications employ the Transmission Control Protocol (TCP) as a transport protocol to detect congestion on the path from packet loss. However, the TCP congestion mechanism only detects

congestion, post-drop event, and could cause instability in buffer length on routers which lead to latency variation and low link utilization [3]. Therefore, many AQM algorithms have been proposed to avoid or control congestion on buffer in the Internet gateways such as RED [4] and REM [5]. AQM algorithms are proposed to improve network performance compared to the traditional Drop-Tail queues. TCP and AQM algorithms are designed to avoid and control congestion on gateways, AQM algorithm avoids congestion by informing the TCP sender to reduce transmission window by setting an Explicit Congestion Notification (ECN) congestion bit in the packet header. Most of the proposed AQM algorithms are designed and implemented to cooperate with responsive traffic such as TCP, in case of unresponsive traffic that employs User Datagram Protocol (UDP) that could cause unfair bandwidth utilisation among traffic flows that share the same link [6]. Usually, UDP traffic carries interactive applications' traffic such as VoIP which is sensitive to bursty drop and delay. Voice packets that are generated at sources and pass through the Internet could face many impairments that could influence voice quality, one of the impairments is network traffic congestion on gateways.

This research is inspired by the absence of evaluation of AQM algorithms with QoS traffic and how could application performance be influenced by congestion on network gateways, and how different AQM algorithms could handle such unresponsive QoS network traffic. Simulations employing NS2 [7] platform ] are conducted with a VoIP traffic representative and analysed, network performance measures, including throughput, delay and PSD (Probability of Sequential Drop) are employed for the evaluation process.

## **2. BACKGROUND AND RELEATED WORK**

Most of the proposed queueing algorithms could be classified into two major classes [8] :passive queueing and active queueing. In passive queueing, packet drop event occurs post queueing process, which indicates that buffer exceeds its capacity as in traditional Drop-Tail queueing, while in active queueing, most proposed queueing algorithms have pre-congestion detection mechanism via set ECN bit in the packet header to inform traffic sources to reduce their transmission rate or drop packet based on pre-calculated probability. The aim of this paper is to bring some light on network performance with QoS traffic and some of well-known AQM algorithms.

Random Early Detection (RED)[4] was introduced in 1993 by Floyd and Jacobson, the main aim of RED is to control average queue length to avoid congestion on the buffer. RED designed to achieve global synchronisation avoidance alongside biases against bursty network flows. RED employs two calculated values, namely minimum and maximum thresholds which work as congestion indicators. Basically, RED uses three scenarios to maintain congestion control, firstly, when Average Queue Length (*AQL*) is less than the minimum threshold (no congestion signs), all arriving packets will be accepted (no drop). Secondly, the *AQL* value between minimum and maximum threshold, all arriving packets will be dropped with a calculated probability *P* (as a linear function of *AQL*) to avoid congestion and cause overflow. Finally, if *AQL* value is more than maximum threshold, all arriving packets will be dropped, in other words drop probability (*P*) is set to 1. *P* is calculated as follows where *maxprob* is the maximum probability to drop a packet:

$$P = \text{maxprob} * \frac{(AQL - \text{minth})}{(\text{maxth} - \text{minth})} \quad (1)$$

The final drop probability (*P<sub>final</sub>*) is:

$$P_{final} = \frac{P}{(1 - cnt * P)} \quad (2)$$

Where  $cnt$  is summation of packets post last dropped packet.

RED's weak points could be the congestion indicator based on the calculating value of  $AQL$  (no. of flows) rather than the actual incoming packets load which could have an impact on network performance in terms of throughput and delay. Additionally, RED does not have the ability to stabilise  $AQL$  value on sudden surpass of number of flows (bursty traffic).

Lapsley and Low proposed Random Early Marking (REM) [5] in 1999. Similar to RED, REM try to keep the queue length low and stabilise arrival flows around link capacity regardless of number of flows to reduce drop rate and delay. REM measures queue congestion by calculating *price*, based on *price* drop probability calculated, the *price* value is updated every time interval based on two factors: firstly, the difference between link capacity and arrival traffic, secondly, the difference between target and queue length. For queue  $L$ , *price* of  $L$  is  $PL(t)$  for period  $t$  and updated according to the equation 3.

$$PL(t + 1) = [PL(t) + \gamma(\alpha L(b_L(t) - b * L) + x_L(t) - c_L(t))]^+ \quad (3)$$

where  $\gamma > 0$  and  $\alpha_L > 0$  are small constants and  $[z]^+ = \max\{z, 0\}$ .  $b_L(t)$  is the current queue  $L$  occupancy in period  $t$  and  $b * L \geq 0$  is target queue length,  $x_L(t)$  is the *arrival* traffic rate to queue  $L$  in time window  $t$ , and  $c_L(t)$  is current bandwidth of queue  $L$  in time  $t$ . The difference  $x_L(t) - c_L(t)$  measures rate of difference, the measurement of queue mismatch is  $b_L(t) - b * L$ . More details can be found in [5] and [9].

To evaluate network performance, the simulation model must configure to represent reality. There are factors that increase realism in simulations, in case of AQM application traffic model, interactive application traffic and network topology, in current state of the art, one of them is often neglected which could lead to less representative results.

Järvinen and Kojo [10] evaluate network performance with CoDel, PIE and HRED algorithms. They use TCP flows only, while the authors in [11] compared CoDel, RED and Drop-Tail in wired-cum network with FTP traffic flows (unresponsive traffic). In [12], authors asses performance of Gentle Blue against DGRED, ERED, BLUE and Adaptive Max Threshold algorithms in absence of realistic application network traffic. Marin et al. evaluate RED and SAP-LAW employing a random heterogeneous traffic model (various traffic loads of both greedy TCP and bursty UDP), however, the network traffic is picked randomly and does not represent an application traffic [13]. Irawan and Surantha [14] compared the performance of three AQM algorithms (Drop-Tail, BLUE and CoDel) on a realistic video streaming application namely Youtube with QoS metrics such as throughput, packet loss and latency, authors use real network topology for the evaluation. However, the authors neglected various network loads which a real network could experience and have an impact on the service. A paper by Chaudhery focuses on Quality of Experience (QoE) requirements to simulate VoIP network traffic in the absence of a multi-level network load. Barczyk M. et al. [15] measured performance of a real neatwork, the experiment lasted for one month, however, they drew a conclusion that the network performed in better way with one of AQM algorithms compared with no AQM algorithms.

Moreover, most of the previously stated related work has used traditional measurement metrics to assess network performance such as average packet loss and average delay. These traditional metrics do not fully reflect the actual network performance. In this paper cumulative distribution function has been employed to illustrate network performance in terms of delay,

and Sequential Drop Rate. It is calculated based on the Probability of Sequential Drop (PSD) events of same flow which give a better indication of QoS VoIP application.

### 3. SIMULATION SETUP

To add realism factor in the simulation model various levels of network load (flows) have been employed which vary from 25 to 150 flows. The simulation runs for 100 seconds, flows start running in cumulative pattern and stops when simulation ends. The simulated traffic model is formed according to Cisco VoIP implementation [16] with packet size of 200 bytes and 50 pps. These configurations could be varied according to Codec scheme. The simulated network topology is illustrated in Fig 1.

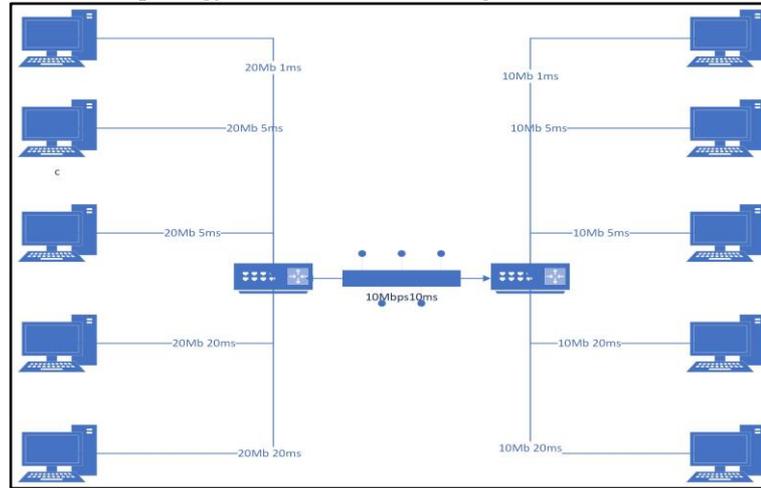


Fig.1: Simulated network topology

### 4. RESULTS AND DISCUSSION

The network performance is analysed using throughput, delay, and PSD measures, all described previously in section 2. The simulation scenario is initiated with an accumulating number of network traffic connections on the configured topology. Fig 2 illustrates PSD for the selected AQM algorithms.

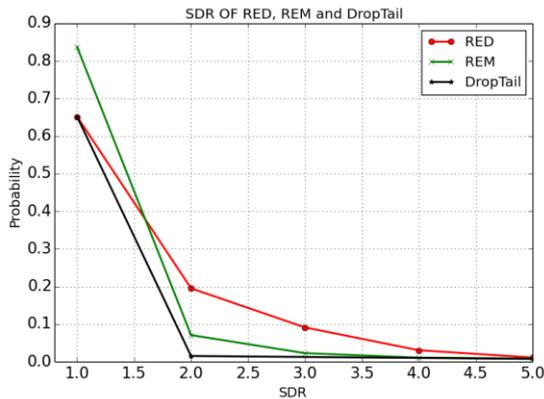


Fig2: short PSD for RED, REM and Drop-Tail.

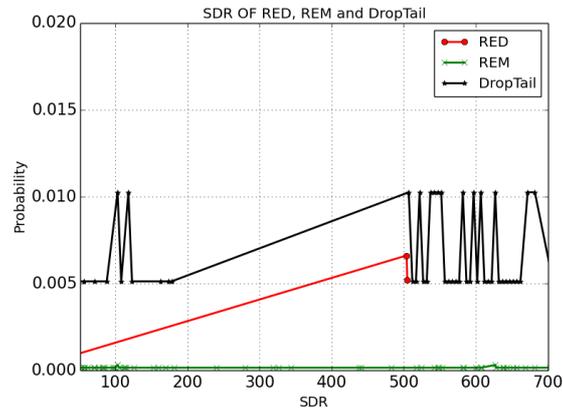
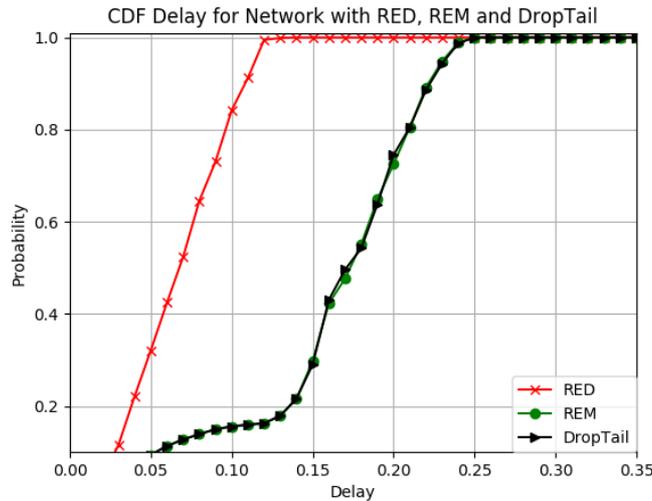


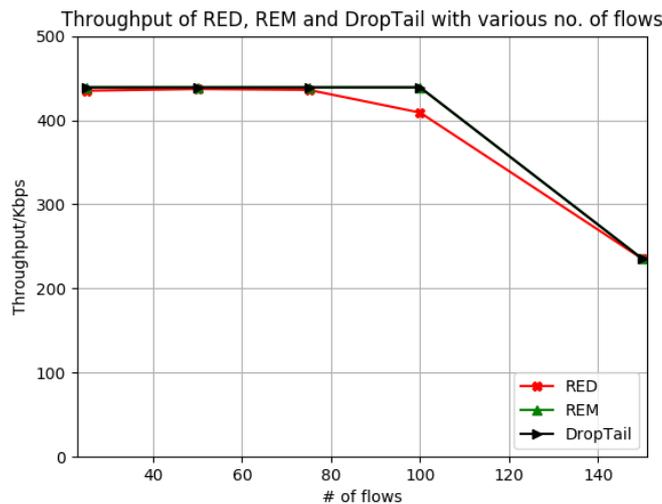
Fig3: long PSD for RED, REM and Drop-Tail.

- In terms of PSD, it has been observed that the network with REM avoided long sequential drop. This is because REM handle increased load and queue growth by pushing *price* variable value up to mark packets rather than dropping them which led to increasing queue length (see Fig 6 and Fig 7), hence, increased delay, unlike RED which tries to keep queue length at the lowest point which keeps delay shorter (see Fig 3). With Drop-Tail, the poor performance is due to buffer blot phenomena (filling buffer) consequently all arriving packets will be dropped, although it outperforms RED and REM in short PSD which is not affecting network performance (see Fig 2).



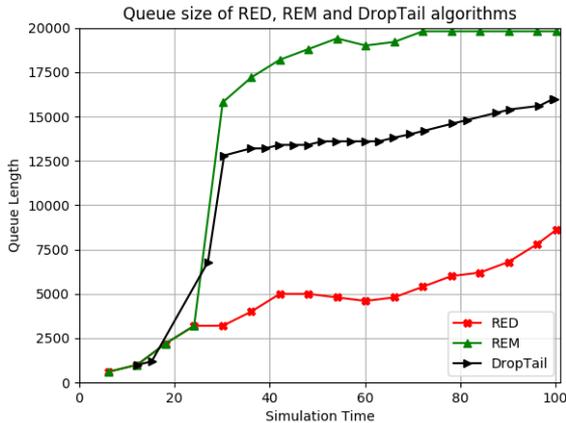
**Fig4:** CDF of delay for RED, REM and Drop-Tail.

- With RED, packets take shorter time (lower delay) to be delivered compared to REM and Drop-Tail which perform similarly (see Fig 4). This is because of the RED mechanism by keeping queue occupancy low, see Fig 6 and Fig 7.

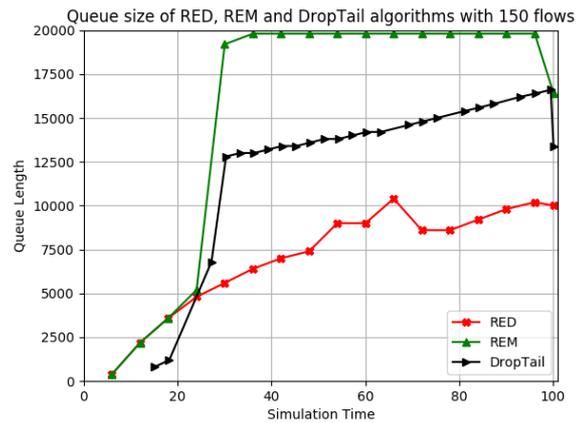


**Fig5:** Throughput for RED, REM and Drop-Tail.

- Despite the weakness in performance with Drop-Tail in terms of delay and PSD, it shows a stable performance in terms of throughput compared to RED which shows a slight poor performance with 100 connections due to the low queue occupancy mechanism, overall, the selected AQM algorithms show similar performance in terms of throughput, see Fig 5.
- All the selected AQM algorithms do not fulfil the VoIP application requirements that includes (high throughput, shorter delay and avoid long PSD), each of them, achieves one or two of the previously stated requirements.



**Fig6:** Queue length for RED, REM and Drop-Tail with 100 flows.



**Fig7:** Queue length for RED, REM and Drop-Tail with 150 flows.

## CONCLUSIONS AND FUTURE WORKS

In this paper a comprehensive evaluation is carried out on a network with different queuing management algorithms to measure its performance on unresponsive data flows, employing VoIP application acting a data flows generator. The results indicate that unresponsive flows have big influence on queue management performance hence the network. Using PSD as performance metric, it is observed that REM outperforms RED and Drop-Tail algorithms. With REM and Drop-Tail, the higher delay is due to queue getting full to utilise queue capacity. With RED the network experience shortest delay compared to other algorithms and stable performance over the simulation.

While no AQM algorithms satisfy VoIP requirements, this makes to come up with some key points to design and implement a new AQM algorithm that have the capability to handle QoS of unresponsive traffic. These key points are as follows:

- Unresponsive traffic has a crucial influence on network performance.
- Keeping queue occupancy low could be employed to reduce delay, however, this could lead to low throughput as utilising the whole buffer causes more packets drop, hence increasing PSD.
- The choice of AQM algorithm influences application performance, especially, real-time applications.
- Despite the fact that it did not measure the fairness score in the simulation analysis, however, it has been observed that RED, REM and Drop-Tail do not implement fairness among traffic flows as they employ randomness in drop decision implementation.

Finally, a new active queue management algorithm needs to be designed, implemented, and evaluated on the same network and more complex and realistic network, employing network measurements including (PSD, throughput, delay, and fairness).

## ACKNOWLEDGEMENT

This research was supported by the University of Mosul / College of Education for pure/ Computer science department science.

## REFERENCE:

- [1] L. Alfat, D. Hanggoro, and R. F. Sari, "Performance evaluation of active queue management in fat tree architecture on data center network," *2019 Int. Conf. Inf. Commun. Technol. ICOIACT 2019*, pp. 22–27, 2019, doi: 10.1109/ICOIACT46704.2019.8938510.
- [2] M. M. Hamdi, S. A. Rashid, M. Ismail, M. A. Altahrawi, M. F. Mansor, and M. K. Abufoul,

- “Performance Evaluation of Active Queue Management Algorithms in Large Network,” *ISTT 2018 - 2018 IEEE 4th Int. Symp. Telecommun. Technol.*, pp. 1–6, 2018, doi: 10.1109/ISTT.2018.8701716.
- [3] M. Barbera, A. Lombardo, C. Panarello, and G. Schembra, “Queue stability analysis and performance evaluation of a TCP-compliant window management mechanism,” *IEEE/ACM Trans. Netw.*, vol. 18, no. 4, pp. 1275–1288, 2010, doi: 10.1109/TNET.2010.2040628.
- [4] S. Floyd and V. Jacobson, “Random Early Detection Gateways for Congestion Avoidance,” *IEEE/ACM Trans. Netw.*, vol. 1, no. 4, pp. 397–413, 1993, doi: 10.1109/90.251892.
- [5] D. Lapsley and S. Lows, “Random early marking for Internet Congestion control,” *Conf. Rec. / IEEE Glob. Telecommun. Conf.*, vol. 3, pp. 1747–1752, 1999, doi: 10.1109/glocom.1999.832461.
- [6] V. A. Reguera, F. F. Álvarez Paliza, E. M. García Fernández, and W. Godoy, “Voice over IP quality of service using active queue management,” *2006 Int. Telecommun. Symp. ITS*, pp. 835–840, 2006, doi: 10.1109/ITS.2006.4433387.
- [7] E. Issariyakul, Teerawat and Hossain, *Introduction to Network Simulator NS2*, 1st ed. Springer Publishing Company, Incorporated, 2010.
- [8] M. Baklizi and J. Ababneh, “Performance Evaluation of the Proposed Enhanced Adaptive Gentle Random Early Detection Algorithm in Congestion Situations,” vol. 6, no. 5, pp. 1658–1664, 2016.
- [9] S. Athuraliya, S. H. Low, V. H. Li, and Q. Yin, “REM: Active queue management,” *IEEE Netw.*, vol. 15, no. 3, pp. 48–53, 2001, doi: 10.1109/65.923940.
- [10] I. Järvinen and M. Kojo, “Evaluating CoDel, PIE, and HRED AQM techniques with load transients,” *Proc. - Conf. Local Comput. Networks, LCN*, pp. 159–167, 2014, doi: 10.1109/LCN.2014.6925768.
- [11] T. Jain, B. Annappa, and M. P. Tahiliani, “Performance evaluation of CoDel for active queue management in wired-cum-wireless networks,” *Int. Conf. Adv. Comput. Commun. Technol. ACCT*, pp. 381–385, 2014, doi: 10.1109/ACCT.2014.97.
- [12] O. Almomani, A. Saaidah, F. Al Balas, and L. Al-Qaisi, “Simulation Based Performance Evaluation of Several Active Queue Management Algorithms for Computer Network,” *2019 10th Int. Conf. Inf. Commun. Syst. ICICS 2019*, pp. 7–12, 2019, doi: 10.1109/IACS.2019.8809117.
- [13] A. Marin, S. Rossi, A. Bujari, and C. Palazzi, “Performance evaluation of AQM techniques with heterogeneous traffic,” *2016 13th IEEE Annu. Consum. Commun. Netw. Conf. CCNC 2016*, pp. 194–199, 2016, doi: 10.1109/CCNC.2016.7444755.
- [14] Y. Irawan and N. Surantha, “Performance evaluation of queue algorithms for video-on-demand application,” *Proc. 2020 Int. Conf. Inf. Manag. Technol. ICIMTech 2020*, no. August, pp. 966–971, 2020, doi: 10.1109/ICIMTech50083.2020.9211138.
- [15] M. B. Id and A. C. Id, “AQM based on the queue length : A real- network study,” pp. 1–21, 2022, doi: 10.1371/journal.pone.0263407.
- [16] C. V. I. 2-53, “Cisco Certified Expert Marketing e.” <https://www.ccexpert.us/unified-callmanager/voip-packet-size-and-packet-rate-examples.html>.