International Journal of Computer Applications (0975 – 8887)
Volume 56– No.8, October 2012

Performance Evaluation of Traffic Meters: Token Bucket Marker and Two Rate Three Color Marker (trTCM) QoS Admission Control

Oyetunji M.O.
Department of Computer Science and Information Technology, Bowen University, Iwo.

Oladeji F.A.
Department of Computer Science, University of Lagos, Akoka, Lagos.

Emuoyinbofarhe O.J.
Department of Computer Science and Engineering, LAUTECH, Ogbomoso.

ABSTRACT

The key role of traffic meter for quality of service admission control internet is to control the amount of traffic injected into the differentiated service network so that congestion can be avoided to the barest minimum in order to meet certain performance requirements. This paper presents performance evaluation and analysis of two traffic meters: token bucket marker (TBM) and two rate three color marker (srTCM) QoS admission control using user datagram protocol (UDP) and transport control protocol (TCP) traffic agents. The performance measures used for the evaluation and analysis were throughput, fairness and losses.

An in-depth study of the aforementioned admission control mechanisms that can enforce service level agreement (SLA) was carried out in order to suggest to the IETF which of the proposed meter would best provide QoS to users. A network simulator, ns-2.35 to be precise, was used to run the simulation to showcase the analysis and evaluation of the traffic meters.

For TCP traffic agent, the analysis based on fairness, throughput and losses showed that the two rate three color marker with percentages 13.15 for fairness, 98.85 for throughput and 0.14 for losses was better in all ramification than the token bucket marker traffic meter with percentages of 12.50 for fairness, 96.66 for throughput and 1.34 losses. For UDP traffic, reverse was the case. The token bucker marker traffic meter with 24.54% of fairness, 62.69% of throughput and 37.37% of losses was better than the two rate three color marker traffic meter with 13.99% of fairness, 21.10% of throughput and 78.99% of losses.

General Terms
Token Bucket, Differentiated Services, Traffic Meter

Keywords
QoS, Admission Control, Token Bucket Marker, trTCM

1. INTRODUCTION

The real-time traffics are always confronted with congestive messages such as “try again or network busy” due to weaknesses of ancient TCP/IP protocol suite to provide quality of service (QoS) in a differentiated manner based on different traffic demands of incoming traffic stream. For decades now, research efforts have been concentrated on the extension of TCP/IP protocol for QoS assurance to accommodate multimedia applications (Voice over IP (VoIP), video over IP (VIP), web learning and remote login). This extension ushered in integrated services and differentiated services architecture as proposed by Internet Engineering Task Force (IETF) [6]. The integrated service architecture is characterized by resource reservation for each session by the router leading to scalability problems when thousands of applications are requesting for reservation at the same time [2]. The weakness of integrated service brought about the advent of differentiated service architecture which makes provision for aggregates traffics to be classified and conditioned at the edge router of the diffserv domain on the basic of performance in order to improve QoS for data and multimedia application in the IP network.

The admission control mechanism or conditioner introduced at the boundary of the diffserv domain checks whether a service request is to be granted or rejected. [4]. The diffserv model uses the mechanism at the edge router to shape, mark, police and drop packets if necessary. The operations are based on service level agreement (SLA) between the ISP and the subscriber. This paper considers the performance of token bucket traffic conditioner and the two rate three color marker traffic conditioner and the analysis were based on throughput, fairness and losses.

The remainder of this paper is organized as follows. Session two discusses the basis of traffic conditioner. Session three talks about token bucket traffic conditioner and its parameters: committed (CIR) information rate and committed burst size (CBS). Session four discusses two rate three color Marker traffic conditioner and its parameters: committed information rate (CIR), peak information rate (PIR), committed burst size (CBS) and peak burst size (PBS). Session five discusses the experimental setup of the simulation. Session six compares the two meters in terms of throughput, fairness and losses and come out with the better meters. We concluded with the summary of the experimental result.

2. BASIS OF TRAFFIC CONDITIONER

The diagram in figure 1 describes traffic conditioner which contains meter, marker shaper or dropper and policer for conditioning functions [1]. The classifier is used to select a class for each traffic flow as the incoming packets pass through it at the boundary of the network. The conditioner can re-mark the incoming packets stream or discard or shape traffic stream to change the temporary characteristic of the traffic stream and make it possible to comply with a traffic profile specified by the network administrator. [9]

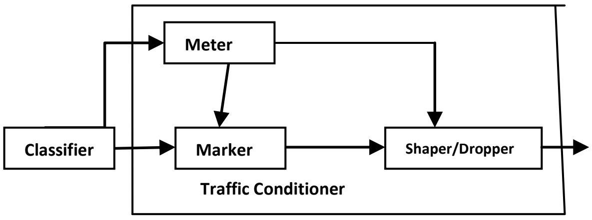

Figure 1: component of a Traffic Conditioner Culled from (source: [8])

3. TOKEN BUCKET

The token bucket marker (TBM) is a control mechanism that dictates when traffic can be transmitted, based on the presence of tokens in the bucket - an abstract container that holds aggregate network traffic to be transmitted. The bucket contains tokens, each of which can represent a unit of bytes or a single packet of predetermined size. It measures a traffic stream based on two parameters: committed information rate (CIR) and committed burst size (CBS). The meters is specified in terms of token bucket with size CBS and token rate CIR [7][8]. Tokens are generated at committed information rate and are added to the token bucket. When packet arrives and there are enough tokens in the bucket, the packet is deemed to be in profile and the relevant numbers of tokens are removed from the bucket. If there are not enough tokens in the bucket the packet is out of profile. The conditioner marks a packet 'green' when it is in profile and 'red' when it is out of profile [8].

4. TWO RATE THREE COLOUR MARKER

This conditioner utilizes four conditioning parameters: committed Information rate (CIR), committed burst size (CBS), peak information rate (PIR) and peak burst size (PBS) [8]. These parameters correspond to two token buckets employed to do the metering. The conditioner assigns one of three colors (green, yellow, or red) based on the states of the two internal token buckets. A packet is marked red if it exceeds the Peak Information Rate (PIR). Otherwise it is marked either yellow or green depending on whether it exceeds or doesn't exceed the committed information rate (CIR). The trTCM is useful in policing a service, where a peak rate needs to be forced separately from a committed information rate [10]. The Meter measures each packet and passes the packet and the metering result to the Marker. The Meter works in one of two modes: Colour-Blind mode and Color-Aware mode. In the Color-Blind mode, the Meter predicts that the packet stream is uncolored. In the Color-Aware mode the Meter assumes that the packet stream has been precolored. The Marker (re)colors the packet according to the results of the Meter. According to [10], the behavior of the Meter is specified in terms of two token buckets, P and C, and its mode with two rates PIR and CIR, respectively. The maximum size of the token bucket P is PBS and the maximum size of the token bucket C is CBS. The token buckets P and C are initially (at time 0) full, i.e., the token count Tp(0) = PBS and the token count Tc(0) = CBS. Later, the token count Tp is incremented by one PIR times per second up to PBS and the token count Tc is incremented by one CIR times per second up to CBS. According to [10], when a packet of size B bytes arrives at time t, the following happens if the trTCM is configured to operate in the Color-Blind mode:

- If 0 > Tp(t)-B, the packet is red, else
- If 0 > Tc(t)-B, the packet is yellow and Tp is decremented by B, else
- The packet is green and both Tp and Tc are decremented by B.

  When a packet of size B bytes arrives at time t, the following happens if the trTCM is configured to operate in the Color-Aware mode:
- If the packet has been precolored as red or if 0 > Tp(t)-B, the packet is red, else
- If the packet has been precolored as yellow or if 0 > Tc(t)-B, the packet is yellow and Tp is decremented by B, else
- The packet is green and both Tp and Tc are decremented by B.

The real implementation of a Meter doesn't need to be modeled according to the above formal specification.
The Marker reflects the metering result by setting the DS field of the packet to a particular code point. In case of the AF PHB the color can be coded as the drop precedence of the packet [5].

5. EXPERIMENTAL SETUP

The experiments were performed in a discrete event simulation environment namely ns-2 version 2.35 to be precise.

5.1 Simulation Topology

The topology was designed and simulated to showcase the two traffic meters: TBM and stTCM in order to determine the better one out of the two in terms of the following parameters: throughput, fairness and losses in a differentiated service domain using TCP and UDP traffic agents. Four identical topologies with the same parameters were designed and simulated as illustrated in figure 2.

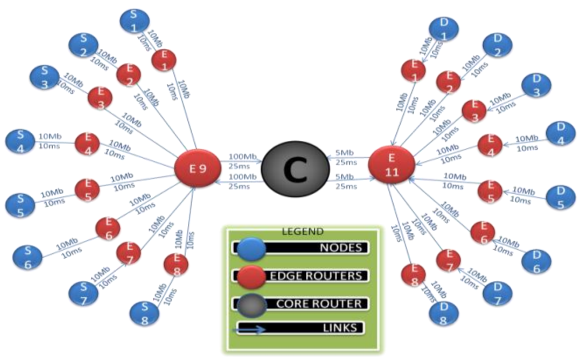

Figure 1 Topology

The nodes in the simulation were connected as shown in figure 2. The topology consists of 34 nodes. The nodes combination can be grouped into four categories: source nodes, edge router nodes (ingress and egress edge nodes), core router node and destination nodes.
The nodes were linked together by 10Mbs bandwidth and 5ms propagation delay except for the ingress edge router to core router which was 100Mbs bandwidth and 25ms delay; and the core router to egress edge routers (congestive link) was 5Mbs bandwidth and 25ms delay. The sources (S_1, S_2, S_3, S_4, S_5, S_6, S_7, S_8), send traffic streams through the edge and core routers to their respective destinations (D_1, D_2, D_3, D_4, D_5, D_6, D_7, D_8). At the diffserv domain, ingress edge router classifies and conditions packets using associated DSCP while core router examines the packets based on their DSCP and forwarding them accordingly to the egress router node and from there to

their various destinations respectively. Each source can either make use of UDP or TCP traffic agent to indicate the type of traffic to be generated. TCP acknowledges its sent packets but UDP traffic does not. Nodes were connected via duplex transmission mode while the link between core router and edge routers were connected via simplex transmission modes. Each source node uses a Drop Tail queue type, while the edges and the core nodes use RED queue type in order to manage buffer overflow and control incipient congestion [3].
The traffic streams were generated by file transfer protocol (FTP) for TCP traffic agent and Constant Bit Rate (CBR) for UDP traffic agent. The Packet size used for the simulation was 2000mb.

5.2 Traffic Policer Settings

A policer measures incoming traffics, enforces compliancy and downgrades packets if necessary to lower preference level (DSCP). The meter settings used in this simulation are

- *For Token Bucket*
  CIR 1000000
  CBS 2000
  Rate 3000000
  With only one precedence level

- *And Two Rate Three Color Marker*
  CIR 1000000
  CBS 2000
  PIR 2000000
  PBS 3000
  Rate 3000000
  With two precedence levels

6. RESULT ANALYSIS

The experiment was carried out for 80 seconds and the statistics (outputs) were taken after every 20 seconds. The dispatch policy employed was priority

6.1. Analysis Based on Fairness

With reference to Jain's Fairness Index, the formula stated below was used to calculate the fairness among the queues:

$$F = \left(\frac{\sum(\mathrm{xi})^2}{N * \sum \mathrm{xi}^2}\right) * 100$$

Where
***"F"** is the fairness percentage, "**xi"** is the queue, "**I"** is the throughput and "**N"** is the number of queue.*

The analysis shows that TBM with UDP traffic agent had the highest fairness percentage, the second was trTCM UDP, the third was trTCM with TCP traffic agent and the fourth was TBM TCP.

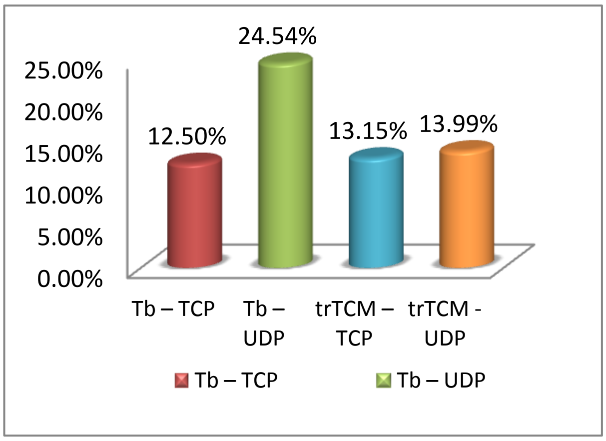

Figure 3: Fairness Graph of all simulated Topologies

Table 1: Table showing average fairness for all the simulated topology

Topology	Tb – TCP	Tb – UDP	trTCM – TCP	trTCM – UDP
Fairness	12.50%	24.54%	13.15%	13.99%

6.2. Analysis Based on Losses

The loss packets has to do with the packets that were enqueued at the ingress edge router but were unable to reach the destination as a result of congestion or bridging of service level agreement between the service provider and the subscriber. The losses were calculated based on the formula stated below:

$$\boldsymbol{Lo} = \frac{(\boldsymbol{Pkt\ enq} - \boldsymbol{pkt\ rec})}{\boldsymbol{pkt\ rec}} * \mathbf{100}$$

Where ***Lo*** *is the percentage packet loss,* ***pkt enq*** *is the packet enqueued at the ingress router and* ***pkt rec*** *is the total packet that gets to the destination*

6.2.1. Loss Rate of Token Bucket using TCP

This was the total packet loss recorded during the simulation of TBM for 80seconds using TCP traffic agent. 282 packets were lost out of 21146 packets received at the ingress edge router. The average packet loss percentage was 1.34%. No packet was dropped early. The remaining packets that were dropped lately were from non-compliant packet.

6.2.2. Loss Rate of Token Bucket using UDP

This was the total packet loss recorded during the simulation of TBM for 80seconds using UDP traffic agent. 44819 packets were lost out of 119928 packets received at the ingress edge router. The average packet loss percentage was 37.37%.

6.2.3. Loss Rate of trTCM using TCP

This was the total packet loss recorded during the simulation of trTCM for 80seconds using TCP traffic agent.36 packets were lost out of 24479 packets received at the ingress edge router. The average packet loss percentage was 0.1470%. All the packets dropped were early drops.

6.2.4. Loss Rate of trTCM using UDP

This was the total packet loss recorded during the simulation of trTCM for 80seconds using UDP traffic agent. 94616 packets were lost out of 119928 packets received at the ingress edge router. The average packet loss percentage was 78.89%.

6.2.5. Average Losses for all the Simulated Topology

The analysis shows that trTCM with UDP traffic agent had the highest loss percentage, the second was TBM UDP, the third was TBM with TCP traffic agent and the fourth was trTCM TCP.

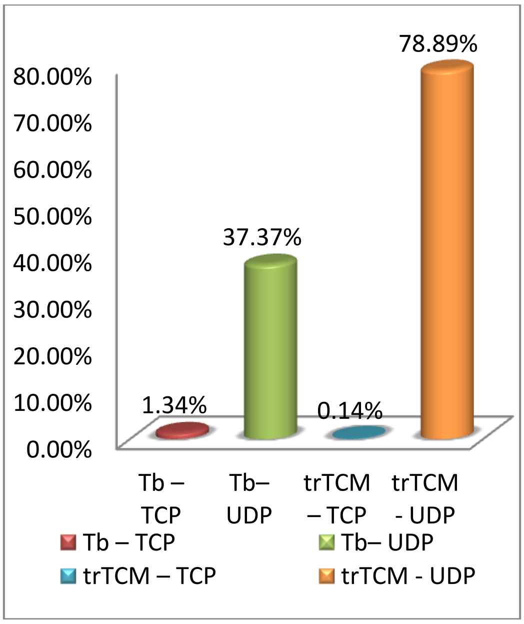

Figure 2 Average losses for all the simulated topologies.

6.3. Analysis Based on Throughput

The throughput result for this simulation was recorded after running the simulation for 80 seconds. The throughput percentage is the ratio of the received packets to the enqueued packets. Its formula is clearly stated below:

$$\boldsymbol{Tru} = \left(\frac{\sum \boldsymbol{Xn}}{\sum \boldsymbol{Y}}\right) * \mathbf{100}$$

Where **Tru** is the throughput percentage, **Xn** is the total packets that got to the destination, and **Y** is the total packet enqueued at the ingress edge router

6.3.1 Token Bucket Throughput using TCP

The total packets enqeued in all queues were 21146 and the total packets that reached the destination were 20864.

$$\boldsymbol{Tru} = \left(\frac{\mathbf{20864}}{\mathbf{21146}}\right) * \mathbf{100}$$

= 98.6664145

Hence, the throughput of the Token Bucket Marker using TCP traffic agent was 98.66%.

6.3.2 Token Bucket Throughput using UDP

The total packets enqeued in all queues were 119928 and the total packets that reached the destination were 75109.

$$\boldsymbol{Tru} = \left(\frac{\mathbf{75109}}{\mathbf{119928}}\right) * \mathbf{100}$$

= 62.62844104

Hence, the throughput of the Token Bucket Marker using UDP traffic agent was 62.62%.

6.3.3 Two Rate Three Color Marker Throughput TCP

The total packets enqeued in all queues were 24479 and the total packets that reached the destination were 24443

$$\boldsymbol{Tru} = \left(\frac{\mathbf{24443}}{\mathbf{24479}}\right) * \mathbf{100}$$

= 99.8529352%

Hence, the throughput of the trTCM using UDP traffic agent was 99.85%.

6.3.4. Two Rate Three Color Marker Throughput UDP

The total packets enqeued in all queues were 119928 and the total packets that reached the destination were 25316

$$\boldsymbol{Tru} = \left(\frac{\mathbf{25316}}{\mathbf{119928}}\right) * \mathbf{100}$$

= 21.1093323

Hence, the throughput of the trTCM using UDP traffic agent was 21.10%.

6.3.5. Average Throughput for all the Simulated Topologies

The analysis shows that trTCM with TCP traffic agent has the highest throughput with 99.85%, the second was TBM TCP with 96.66%, the third was TBM with UDP traffic agent and the fourth was trTCM UDP with 21.10%

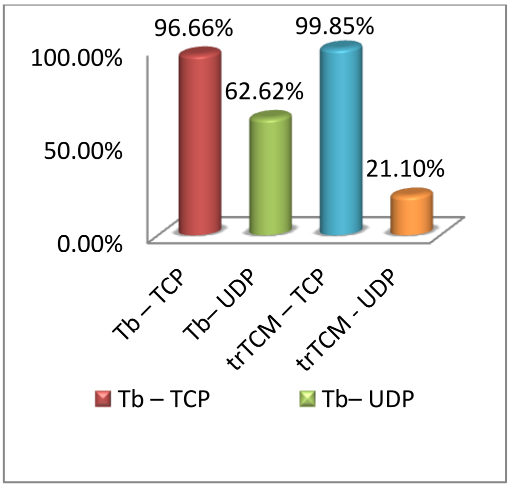

Figure 3: Average throughput for all the Simulated Topology.

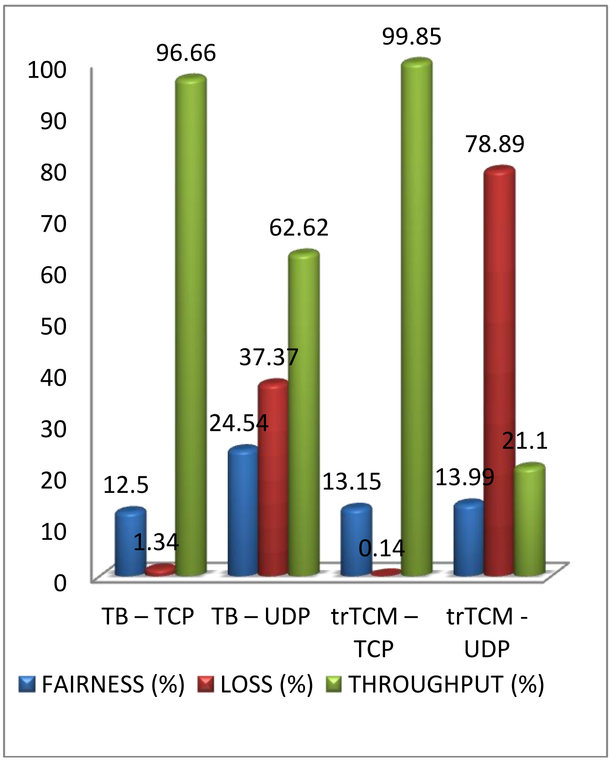

Figure 4 Performance Evaluation of all Topologies

Table 2: Table Showing the Performance evaluation of all Topologies

Traffic Meter	Traffic Agents	Fairness (F)	Losses (Lo)	Through put (Tru)
TBM	TCP	4th =12.50	3rd =1.34	2nd =96.66
	UDP	1st =24.54	2nd =37.37	3rd =62.69
trTCM	TCP	3rd =13.15	4th =0.14	1st =98.85
	UDP	2nd =13.99	1st =78.99	4th=21.10

7. CONCLUSION

This research paper has to do with comparison of two differentiated service policers: Token Bucket Marker and Two Rate Three Color Marker, they were tested under the same environments with the same parameters all true. The performance evaluations of the two traffic meters were based on the following parameters: fairness losses and throughput.

From the compiled analytical results, it was revealed that two rate three color marker implemented with the TCP protocol has the highest throughput value of 99.85%, followed by TBM implemented with the tcp protocol as well, it has a throughput value of 96.66%. Two rate three color marker and token bucket implemented with the UDP protocol were third and fourth with values of 62.62% and 21.10% respectively as shown in Table 2. This result was not unexpected due to the fact that the UDP protocol does not require packet acknowledgement before sending another packet, this makes it a lot faster than TCP in terms of packet sending, but also makes it unreliable and wasteful as packets lost in transit cannot be accounted for.

In terms of loss (inverse of throughput), two rate three color marker implemented with the TCP protocol, had the lowest loss rate of 0.14%, seconded by TBM implemented also with the TCP protocol; it had a loss rate of 1.34%, two rate three color marker and token bucket implemented with the UDP protocol were third and fourth with values of 37.37% and 78.89% respectively. In terms of fairness, TBM implemented with the udp protocol had the highest rate of 24.54%, next was trTCM implemented with the UDP protocol with 13.99% followed by TBM and trTCM implemented with the TCP protocol with 13.99% and 12.50% respectively. The result proves the rule that fairness and throughput are inversely proportional, this is because if all packets from all sources are treated fairly, they would be no precedence hence congestion ensues. To buttress this point; from the comparism of all the fairness rates of all the topologies, it's noted that the topologies with the highest fairness rates (trTCM and TBM implemented with UDP) also had the lowest throughput values, and trTCM and Tb implemented with TCP, which have the highest throughput values have the lowest fairness values.

To summarize all this, if fairness is not an issue, the best policer is two rate three color marker implemented with TCP followed by token bucket implemented also with TCP, but if fairness is to be considered then two rate three color marker implemented with UDP followed by token bucket implemented also with UDP. Either way two rate three color marker is the better policer in all aspects.

8. REFERENCES

[1] Blake, D., Black, S., Carlson, M. Davies, E., Wang, Z.,and Weiss, W. 1998. An architecture for Differentiated Services. RFC 2475

[2] Braden, R., Clark, D., Crowcroft, J., Davie, B., Deering, S., Estrin, D., Floyd, S., Jacobson, V., Minshall G., Partridge, C., Peterson, L., Ramakrishnan, K., Shenker, S., Wroclawski, J., and Zhang, L. 1998. Recommendations on queue management and congestion avoidance in the Internet. RFC 2309

[3] Floyd, S., and Jacobson, V. 1993. Random Early Detection gateways for congestion avoidance. *IEEE Transactions on Networking*, 1, 4

[4] Georgoulas, S., Trimintzios, P., Pavlou, G., 2004. Admission control placement in differentiated services networks. Ninth International Symposium on Computer and Communication Proceddings 2 (June 2004), 816-821

[5] Heinanen, J., Guerin, R. 1999. A Single Rate Three Color Marker. Informational RFC 2697.

[6] Oladeji, F., Uwadia, C., Oyetunji, M., and Adeyemi, M. 2010. "The significance of traffic admission control and scheduling mechanisms in a qos internet router", International Journal for the Application of Wireless and Mobile Computing, 2010, ISSN:2141-0720

[7] Stallings, W. 1998. High Speed Networks: TCP/IP and ATM Design Principles. Prentice-Hall.

[8] Strauss,T. DERRICK, G., KOURIE, S., AND OLIVIER, M. 2005. A Simulation Study of Traffic Conditioner Performance. In Proceedings of SAICSIT.

[9] Habib, A., Fahmy, S., and Bhargava, B. 2001. Design and Evaluation of an Adaptive Traffic Conditioner for Differentiated Services Networks. In Proceeding of IEEE International Conference on Computer Communication and Networks (IC3N), Arizona.

[10] Heinanen, J., Finland, T., and Guerin, R. 1999. A Two Rate Three Color Marker. Informational RFC 2698