主动队列管理机制的性能分析

张鹤颖　窦文华
（国防科学技术大学计算机学院　长沙410073）

摘　要　主动队列管理(AQM)是拥塞控制中一个热点。通过NS仿真器，深入研究了几个AQM算法的性能。仿真结果显示，没有一个AQM算法在所有网络条件下是最好的。它们存在响应速度、链路利用率等性能不足。分析了性能局限性的原因，指出了今后的研究方向。

关键词　拥塞控制，主动队列管理，队列长度，响应性

Comparison and Analysis of Active Queue Management

ZHANG He-Ying　DOU Wen-Hua
(Computer School, National University of Defense Technology, Changsha 410073)

Abstract　Active Queue Management (AQM) is an active area in congestion control. The performances of several typical AQM schemes are verified by NS simulator. The simulation results show that none of the AQM scheme is perfect in all network conditions. They have shortcomings either in responsiveness or in link utilization. The reasons for the performance limitations are analyzed. In addition, the direction for future research is pointed out.

Keywords　Congestion control, Active queue management, Queue length, Responsiveness

1　引言

当前网络中的许多应用要求低延迟、低丢失率的服务，现有的端到端拥塞控制机制难以满足这些性能要求，IETF建议在路由器中采用主动队列管理(AQM)机制[1]。AQM主要是针对传统路由器中的"弃尾"队列提出来的，目的是设计一种有效的拥塞早期检测机制，将路由器中的队列长度控制在较低的水平，避免缓冲区溢出，并减少报文丢失。AQM的作用主要表现在：(1)较早检测拥塞，控制队列长度和队列动荡，从而减小排队延迟和延迟抖动；(2)避免缓冲区溢出，减少报文丢失；(3)使报文丢弃概率与流的到达速率成正比，限制高速流占用过多带宽。

目前已经提出了多种AQM算法，这些算法的主要区别在于拥塞检测方法的不同。最早的RED算法基于平均队列长度检测拥塞并决定报文丢弃概率[2]。许多文献研究了RED算法在各种网络环境下的性能，指出RED主要有两大缺陷：(1)算法性能对参数敏感；(2)队列长度随着负载的增加而增长。针对这些缺陷，提出了一些改进方法。这些改进方法主要可以分为两类：一类是动态调节RED的关键参数，使报文丢弃概率与网络拥塞程度相符合，如自适应RED算法ARED[3]；另一类是采用其它拥塞检测方法，如比例积分控制器PI[4]，随机指数标记算法REM[5]，自适应虚队列算法AVQ[6]等。其中，ARED、PI和REM的目标是将队列长度控制在预先设置的参考点附近，AVQ的目标是将链路利用率保持在特定的值。这些算法都能够在一定程度上解除队列长度与负载的耦合，但是在响应速度、链路利用率等方面的性能并不理想，本文将通过仿真，深入研究这些算法的性能，指出其局限性，并分析产生的原因，为以后的研究指明方向。

2　算法及性能分析

由于RED算法的性能分析已经比较充分，这里将不再赘述。本文主要采用网络仿真器NS研究最近提出的ARED，PI，REM和AVQ算法的性能，性能指标包括队列长度、响应速度、链路利用率以及报文丢弃率。仿真中采用的网络拓扑结构如图1所示。端节点s_i和d_i之间建立连接，路由器r1和r2之间构成瓶颈链路，链路带宽为10Mbps，传输延迟为10ms。其它链路的带宽为30Mbps。发送端的传输控制协议是TCP-Reno，报文的平均长度为1000B，TCP连接的传输延迟均匀分布在40ms到220ms之间。如果没有特殊说明，路由器中的缓冲区最多可容纳100个报文，路由器r1采用显示拥塞指示(ECN)进行拥塞反馈[7]。用FTP流模拟一直有数据发送的长连接。主要通过观察AQM在不同负载条件下的瞬时队列变化来研究它的响应速度、链路利用率和延迟等性能。

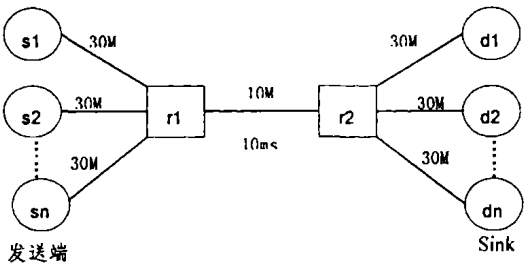

图1　实验网络拓扑

2.1　ARED算法

RED算法的稳态平均队列长度由负载程度和算法参数决定，使得算法性能难以预测。ARED算法根据当前队列长度不断调节最大丢弃概率max_p，使丢弃概率符合网络拥塞程

张鹤颖　博士生，主要研究方向是高速网络互连，网络拥塞控制。窦文华　教授，博导，主要研究方向为网络安全，高速网络互连，CIMS，并行与分布处理等。

度，从而将队列长度保持在目标值附近。ARED 采用"加式增加，乘式减少"方法调节 max_p，具体如下所述：

```
if (avg>target and max_p≤0.5)
    max_p← max_p+α;
else if (avg<target and max_p≥0.01)
    max_p←max_p * β;
```

其中，avg 为平均队列长度，target 为目标队列长度，其取值范围为：

$[min_{th}+0.4*(max_{th}-min_{th}), min_{th}+0.6*(max_{th}-min_{th})]$

增加因子 α 的取值范围为：$\min\{0.01, max_p/4\}$，减小因子 $\beta=0.9$，$min_{th}=\max\{5, delay_{target}C/2\}$，$max_{th}=3min_{th}$，$Wq=1-\exp(-1/C)$。其中，C 为链路带宽。

仿真开始时，FTP 流的数目是180，100秒时160个 FTP 流退出，ARED 和 RED 的瞬时队列变化分别如图2(a)和图2(b)所示。从图中可以看出，当负载较重时，ARED 与 RED 的性能比较接近，队列动荡比较大。在负载较轻时，ARED 能够明显减小队列动荡。表明 ARED 在轻负载时的性能优于重负载时的性能。另外，由于低队列阈值 min_{th} 比较小，导致在负载比较重的情况下，ARED 的队列长度经常为0，这意味着链路利用率小于1。如果增大 min_{th} 的值，可以提高链路利用率，同时将增大排队延迟。因此 min_{th} 的设置需要在利用率与延迟之间做出折衷。只有在队列动荡比较小的情况下，这两方面的性能才能够得到较好的保证。所以，保持较小的队列动荡也是 AQM 机制的重要性能指标之一。ARED 的优点是响应速度快，几乎没有队列长度很高的过渡阶段，而且负载对响应速度没有显著的影响。

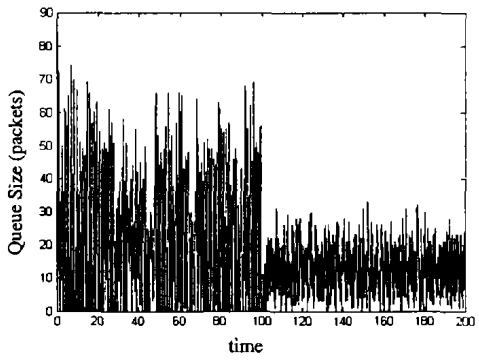

图2(a)　ARED 瞬时队列变化

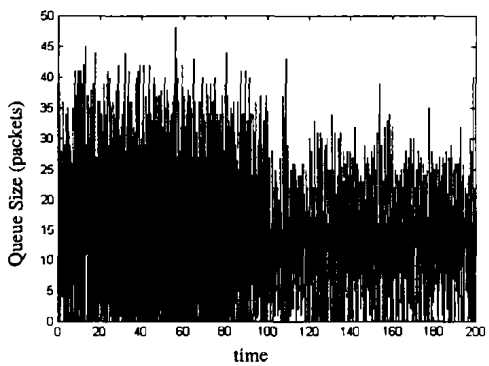

图2(b)　RED 瞬时队列变化

2.2　PI 算法

通过对 TCP-RED 系统的控制论分析，发现 RED 中计算平均队列长度的低通滤波器引起的滞后是系统响应速度慢的原因之一。PI 算法用比例积分(PI)控制器取代低通滤波器，基于瞬时队列长度决定报文丢弃概率。由于积分环节能够消除系统的稳态误差，所以 PI 能够将队列长度控制在某个参考值附近。与 ARED 相比，PI 的实现非常简单，其丢弃概率计算方法如下所述：

$$p=a*(q-q\_ref)-b*(q\_old-q\_ref)+p\_old$$

其中，a，b 为常数，q_ref 为参考队列长度，q_old 为前一次采样的队列长度，p_old 为前一次计算的丢弃概率。仿真中 a=0.00001822，b=0.00001816，q_ref=20。

图3显示了 FTP 流的数目为20的队列变化，N 表示 FTP 流的数目。可以看出队列长度收敛到参考点所需要的时间比较长，表明即使在负载较低的情况下，PI 的响应速度也很慢。表1比较了不同负载和不同缓冲区大小条件下的响应时间。结果表明增大缓冲区能够加快响应速度，但是与 ARED 相比，PI 的响应速度仍很缓慢。PI 的优点是队列动荡比较小，链路利用率高。

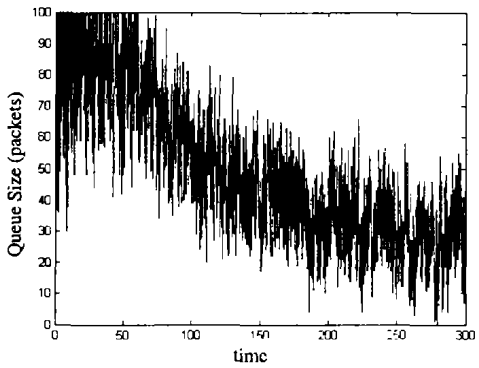

图3　N=20瞬时队列长度

表1　PI 的响应速度

缓冲区大小	N=20	N=180
100	200s	400s
1000	100s	120s

2.3　REM 算法

REM 将报文到达速率与链路带宽的差值和瞬时队列长度与目标队列长度的差值的加权组合定义为链路"价格"，由"价格"决定报文丢弃概率，每个流端到端的丢弃概率是该流经过的所有链路的"价格"之和。REM 的目标是使报文到达速率与链路带宽相匹配，从而清空缓冲区，即所谓的"match rate，clear buffer"。由于计算报文到达速率需要保存一定的状态信息，为了避免计算报文到达速率，将速率差值用队列差值近似。链路 l 的价格 $p_l(t)$ 按下式计算：

$$p_l(t+1)=[p_l(t)+\gamma(b_l(t+1)-(1-\alpha_l)b_l(t)-\alpha_l b^*)]^+$$

其中，$\gamma>0$，$\alpha_l>0$，$[z]^+=\max\{0,z\}$，$b_l(t)$ 为链路 l 的队列在 t 时刻的瞬时队列长度，b^* 为目标队列长度。

链路 l 的队列在 t 时刻的标记概率为：$m_l(t)=1-\phi^{-p_l(t)}$。ϕ 为常数，且 $\phi>1$，报文端到端的标记概率为：$1-\phi^{-\sum_l p_l(t)}$。

一般情况下取 $\gamma=0.001$，$\phi=1.001$，$\alpha_l=0.1$，$b^*=20$。

图4显示了 FTP 流的数目为20的队列变化。与 PI 相比，REM 收敛到参考点的时间有所减小，但是仍然比较长。表2比较了不同负载和不同缓冲区大小条件下的响应时间。REM 的响应时间也随着负载的增加而增长，增大缓冲区能够减小响应时间。REM 的性能与 PI 非常相似，但是 REM 的响应速度比 PI 快。此外，REM 的队列动荡略大于 PI，链路利用率比较高。

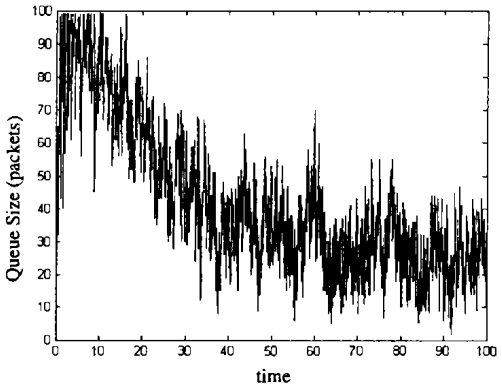

图4　N=20瞬时队列长度

表2　REM 的响应速度

缓冲区大小	N=20	N=180
100	60s	120s
1000	35s	70s

2.4　AVQ 算法

AVQ 算法建立了一个链路带宽小于实际链路的虚队列，对每个到达的报文，按照虚容量 $\tilde{C}$ 计算虚队列的长度，如果虚队列的长度大于缓冲区的最大容量，将丢弃(或标记)到达的报文。虚队列的容量按下式调节：$\dot{\tilde{C}}=\alpha(\gamma C-\lambda)$。其中，$\gamma$ 为目标链路利用率，α 为阻尼因子，λ 为报文到达速率。α 的取值决定了报文丢弃概率跟踪网络状态变化的速度。

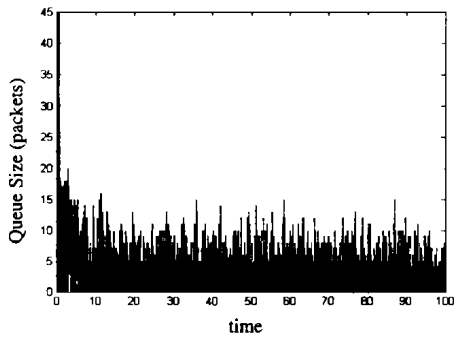

图5(a)　N=20瞬时队列长度

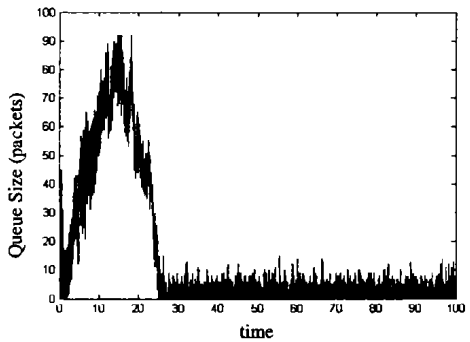

图5(b)　N=180瞬时队列长度

仿真中取 $\alpha=0.15$，$\gamma=0.895$，图5(a)和图5(b)显示了FTP 流的数目为20和180的队列变化。AVQ 算法能够将队列长度控制在很低的水平，而且稳态队列长度不受负载变化的影响。但是，AVQ 收敛到稳定状态的时间也随着负载的增加而增长。而且，AVQ 的链路利用率低，实验得到的实际利用率与目标利用率 γ 的关系如表3所示。

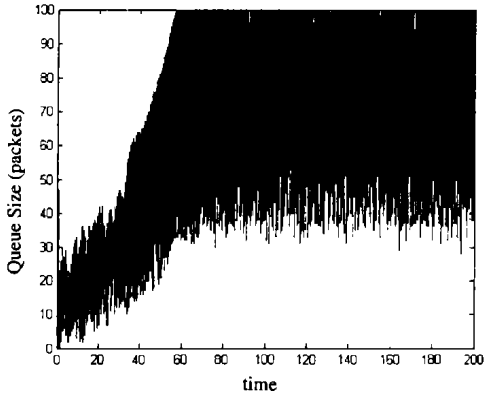

图5(c)　B=100瞬时队列长度

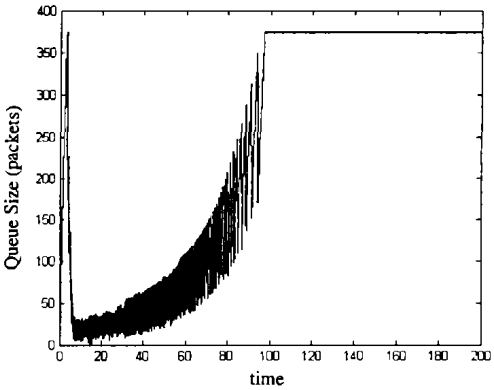

图5(d)　B=1000瞬时队列长度

表3　AVQ 的实际利用率与目标利用率的关系

目标利用率	实际利用率	目标利用率	实际利用率
0.895	0.936	0.955	0.99
0.915	0.936	0.975	1.0
0.935	0.97	0.995	1.0

实验得到的实际利用率略大于目标利用率，而且目标利用率越高，实际利用率也就越高。当 $\gamma=0.975$时，链路利用率为1。在这种情况下，当 FTP 流的数目为20时队列变化如图5(c)所示，将缓冲区增大到1000时队列变化如图5(d)所示，图中 B 表示缓冲区大小。仿真结果表明如果提高链路利用率，即使在负载很轻的情况下，队列长度也很大，而且增大缓冲区对算法性能没有显著影响。

2.5　TCP 流的报文丢弃率

TCP 流是一种拥塞响应流，当网络发生拥塞时，无论路由器采用的拥塞反馈方法是标记报文还是丢弃报文，TCP 流都会采取回退策略，减小发送速率以缓解网络拥塞。在标记策略下，四种算法的报文丢弃率都较低。ARED 在负载较轻时，丢弃率为0，当负载较重时，由于队列长度超过最大阈值，导致报文被丢弃。PI 和 REM 在队列到达稳态之前，由于缓冲区溢出，导致大量报文被丢弃。到达稳态之后，报文丢弃率为0。AVQ 在标记策略下报文丢弃率几乎为0，因为只有当实际队列溢出时，AVQ 才丢弃报文，而队列溢出很少发生。在丢弃策略下，FTP 流的数目从20变化到120时，100秒内报文丢失总数如图6所示。四种算法的丢弃率都比较高，ARED，PI，REM 的丢弃率高于弃尾队列，AVQ 的丢弃率在负载较轻时也高于弃尾队列。从仿真结果可以看出，ECN 标记策略能够使 AQM 算法减小报文丢弃率，在丢弃策略下，AQM 的报文丢弃率比

(下转第73页)

600，测试软件为思博伦通信公司的应用代理专用测试软件TeraCaw。TPF策略配置为最小规模策略集。测试时长为120秒。对http协议代理的测试结果如下：

在50%吞吐量下网路仿真测试下平均处理延时为150微秒。

最大并发连接数为32.8万。在追求吞吐量的仿真测试中，每秒实际处理完成的最大请求数为989条。

TPF与代理的应用层实现及IP网关的最大吞吐量的对比测试数据见图7。

结束语 TPF的内核级实现方式，克服了存在于传统代理防火墙的诸多弱点，提高了自身安全性，很好地改善了代理防火墙的网络处理性能，同时也使得代理支持的应用协议数量扩充变得更加容易。下一步，我们将在ESK中引入代理程序与内核隔离机制，同时对协议栈快速通道的处理流程做进行优化，并改进协议报文分析程序的效率，以期进一步提高TPF的自身安全性和性能。

感谢 南京大学计算机系研究生赵静凯、卜宏、郭晓芳和软件工程中心职工朱佳来、李论等人在TPF的原型开发中承担了大量的编码工作。在此对他们的辛勤劳动表示衷心的感谢！

参考文献

1 Herrin G. Linux IP Networking. May 2000. http://kernelnewbies.org/documents/ipnetworking/linuxipnetworking.html
2 Anand V, Hartner B. TCP/IP Network Stack Performance in Linux Kernel 2.4 and 2.5 2002. http://www.linuxsymposium.org/2002/view_txt.php?text=abstract&talk=91
3 Stevens W R. TCP/IP Illustrated, Volume 2: Implementation. Addison Wesley Longman, Inc, 2000
4 毛德操，胡希明. Linux 2.4内核源代码情景分析. 浙江大学出版社，2001
5 Payne C, Markhan T. Articeture and Application for a Distributed Embedded Firewall. 2002. www.acsac.org/2001/papers/73.pdf
6 Cisto Systems, Evolution of the Firewall Industry 2002 http://www.cisco.com/univercd/cc/td/doc/product/iaabu/centri4/user/scf4ch3.htm
7 茅兵，等. 863计划《基于Linux的操作系统安全增强技术的研究与开发》课题技术报告，2003
8 蔡圣闻，等. 江苏省软件和集成电路专项《高安全高性能的网络防火墙》项目技术报告，2003
9 Srinivasan V. Fast and efficient internet lookups[D]: [Ph. D Thesis]. Washington University, 1999
10 卜宏，黄皓. 一种应用层协议报文解析算法. 计算机应用研究. 2003
11 Gallatin A, Chase J, Yocum K. Trapeze/IP: TCP/IP at Near-Gigabit Speeds, 1999 USENIX Annual Technical Conference
12 Epstein J, Thomas L, Monteith E. Using Operating System Wrappers to Increase the Resiliency of Commercial Firewalls. In: 16th Annual Computer Security Applications Conf. 2000
13 Kang J-M, et al. Extended BLP Security Model Based on Process Reliability for Secure Linux Kernel. 2001. http://www.computer.org/proceedings/prdc/1414/1414toc.htm
14 Barford P, Crovella M. Critical Path Analysis of TCP Transactions. In: Proc. of the 2000 ACM SIGCOMM Conf. Sep. 2000

（上接第60页）

较高。而ECN对弃尾队列的报文丢弃率没有影响。

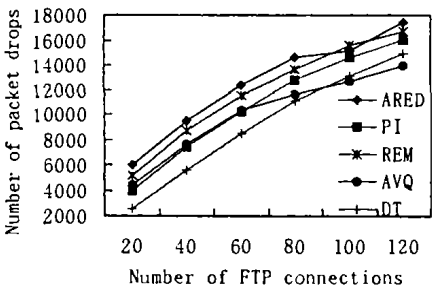

图6 报文丢弃率

结论 本文通过仿真对最近提出的几种重要AQM算法的性能进行了分析和比较，仿真结果表明：

(1)响应速度：ARED算法的响应速度最快，PI，REM，AVQ算法的响应速度随着负载的增加而减慢，其中PI的响应速度最慢。

(2)队列长度的控制能力：这四种算法都能够将队列长度控制在特定的水平，消除了稳态队列长度与负载的耦合。ARED在负载较重时，队列动荡大。PI和REM在收敛到稳态以后，能够将队列长度控制在参考点附近。AVQ只有在链路利用率较低的情况下，才能够将队列长度控制在较低的水平。PI，REM，AVQ队列动荡小。

(3)链路利用率：ARED，PI和REM的链路利用率比较高，AVQ的链路利用率与目标利用率的设置有关，需要在利用率和队列长度之间取折衷。

(4)报文丢弃率：标记策略能够大大降低AQM算法对TCP流的丢弃率。PI和REM在稳态时报文丢弃率为0。AVQ报文丢弃率一直为0，ARED在负载较轻时，报文丢弃率为0，负载较重时，由于队列长度大于最大阈值，导致报文被丢弃。

另外，ARED的实现复杂度是最高的，PI，REM和AVQ都比较容易实现。

在文中所讨论的四种算法中，ARED算法是动态调节参数的典型算法。由于ARED算法是对RED算法的增强和改进，所以算法的参数配置复杂，而且参数配置主要依靠经验方法，而不是基于系统的、科学的分析方法。PI是运用控制论设计的AQM控制器，它的参数选择基于系统的稳定性要求，能够比较容易地确定符合网络状况的参数。但是，PI控制器本身是一种滞后控制，所以不可避免具有响应速度慢的特点，而且PI控制器的响应速度与负载相关。因此，基于控制论的研究应该侧重于提高系统的响应速度和鲁棒性。REM算法是通过求解最优化问题得到的，其形式和性能均与PI相似。一般情况下，基于最优化方法设计的AQM机制需要与端节点的流控机制相结合，才能得到系统的最优性能。AVQ算法本质上是一种基于速率的AQM算法，由于报文到达速率很难反映拥塞的持续状况，所以AVQ算法对队列长度的控制能力有限。如何将报文到达速率与队列长度相结合来判断网络拥塞，是值得研究的问题。

参考文献

1 Braden B, et al. Recomedations on Queue Management and Congestion Avoidance in the Internet, RFC2309, April 1998
2 Floyd S, Jacobson V. Random Early Detection Gateways for Congestion Avoidance, IEEE/ACM Transactions on Networking, 1993, 1(4): 397～413
3 Floyd S, Gummadi R, Shenker S. Adaptive RED: an algorithm for increasing the robustness of RED's Active Queue Management. http://www.icir.org/floyd, Aug. 2001
4 Hollot C, Misra V, Towsley D, Gong W. On designing Improved Controllers for AQM Routers Supporting TCP Flows. In: Proc. of IEEE INFOCOM'01, Anchorage, Alaska, USA, 2001. 1726～1734
5 Lapsley D, Low S. Random early marking: An optimization approach to internet congestion control. In: Proc. of IEEE ICON '99, Brisbane, Australia, 1999. 67～74
6 Kunniyur S, Srikant R. Analysis and Design of an Adaptive Virtual Queue (AVQ) Algorithm for Active Queue Management. In: Proc. of ACM SIGCOMM 2001, San Diego, California, USA, 2001. 123～134
7 Ramakrishnan K K, Floyd S, Black D. The Addition of Explicit Congestion Notification (ECN) to IP. RFC3168, 2001