Global Information Assurance Certification Paper

Managing the Commons in Cyberspace: The Promise and Challenges of Quality of Service

David F. Johnson
November , 2002

Abstract

Efficient and fair allocation of the limited resources of the Internet is becoming more imperative, and the many challenges associated with bringing this about are being actively addressed. Considerable progress is being made towards providing specified and desired levels of service to network traffic, ensuring efficient use of limited network resources, and also protecting against misuse of network resources. QoS routing finds routes with adequate unused resources to satisfy the QoS constraints of the desired connection, while efficiently coordinating the use of limited network resources. The promise of QoS is significantly offset by costs, which include increased computational cost, and in support of this, an increased volume of protocol traffic. Directly related to these two costs, are the two main determinants of the degree that a particular approach to QoS will increase network traffic: the path selection algorithm, and the quality of the information regarding network resources that supports the path selection algorithm. Classified based on how the state information is maintained and how path selection is conducted, the following routing strategies may be identified: source routing, distributed routing and hierarchal routing. Unlike hierarchal routing, much progress has been made related to source and distributed routing strategies. Source routing simplifies the path selection process by maintaining a complete global state at each node, allowing for the feasible path computations to be done locally. In distributed routing, intermediate nodes share the responsibility for path selection.

Security should be integrated into QoS mechanisms early on. QoS must incorporate mechanisms to prevent, detect, and recover from attacks. For example if preventative mechanisms are not in place when network resources are allocated to provide QoS, an attacker may reserve for himself as many resources as he pleases, or delete or alter an existing reservation. The attacker may steal resources, as well as deny the use of resources to other users. To prevent such attacks, provision of QoS must include: authorization and authentication mechanisms; enforceable resource allocation mechanisms, which exploit user's price sensitivity; and monitoring mechanisms that effectively detect and respond to attacks.

Table of Contents

1 INTRODUCTION

In his account of *The Tragedy of the Commons*, Garrett Hardin paints a picture of a village where herdsmen were free to let graze on the village commons, as many cattle as they pleased. As perceived by each herdsman, the benefits of placing additional cattle on the commons greatly outweighed any costs. Since this attitude was shared among all of the herdsmen, and there were no mechanisms in place to allocate limited resources, the destruction of the commons was assured. [6] In a somewhat similar manner, the traditional "best effort" service of the Internet effectively treats the Internet as a common pool resource that is vulnerable to inefficient use, as well as misuse. Normal users compete for limited resources, and seek to maximize their quality of service without regard for other users. Malicious users seek to deliberately deny quality of service to other users. [9] To avoid this variation of the 'tragedy of the commons,' mechanisms can be put in place that provide a guaranteed quality of service (QoS) to network traffic, ensure efficient use of limited network resources, and also protect against misuse of network resources, thus making the Internet more flexible, reliable, efficient and secure. [4]

This paper presents an overview of QoS routing security issues, and related promises, problems, solutions, and future challenges. Before confronting the need to integrate security into QoS solutions, a foundation is laid. Different classes of routing strategies, and their strengths and weaknesses are examined. Basic algorithms in each class are elucidated and compared. In confronting the need to integrate security mechanisms into QoS solutions, authorization, authentication, enforceable resource allocation, and detection and response mechanisms – are all put forward as integral and essential to providing effective, secure QoS.

2 BACKGROUND

Beyond bringing the promise of more efficient use of the Internet, QoS also carries the promise of a new generation of much desired applications. The provision of specified, consistent levels of QoS is vital to the successful deployment of many new network applications. Such applications include distributed multimedia; IP-telephony; video conferencing; and real-time distributed simulation, control, and collection of data from sensors. Just as the Internet continues to grow, so does the demand for these more sophisticated applications, and the services that will support them. In addition, the increase in the number of users, traffic levels, topological complexity, as well as economic driving forces all call for more efficient use of network resources. At the same time, research in areas including high-speed networks, image processing and video/audio compression has borne many fruits. This research and the expanding Internet have symbiotically fueled and benefited each other, to the point where, for example, such aspirations as the deployment of applications that provide reliable and timely delivery of digitized audio-visual information may be within reach. The ability to do so is contingent on the ability to efficiently provide an underlying reliable quality of service to these applications. [4, 5, 12]

This quality of service is provided by reserving the required resources, and in the case of real-time traffic, providing connection-oriented service; thereby enabling the placing of limits on the allowable end-to-end delay, delay variation (jitter), and the packet loss rate, while guaranteeing minimum data throughput. This is in contrast to the current Internet where network resources are inefficiently shared by packets from different sessions. Further, these packets may make their way to their destination along different paths. Through the process known as QoS routing, the path to be used by the packets of a flow is selected so that a guaranteed network bandwidth and response time is provided. [1] Use of a path selection process that is sensitive to ensuring the needs of the overlying application provides a high quality of service to users, and also increases network efficiency and network utilization. QoS routing increases the carrying capacity of a network, both in terms of the number of flows, as well as the total amount of bandwidth. Effective QoS routing optimizes use of network resources and endeavors to distribute loads equally among paths, while efficiently reacting to changing traffic patterns and failures of network elements. [2, 4]

3 QoS ROUTING CONSTRAINTS

QoS routing seeks to find a feasible path or tree – a path or tree with adequate unused resources – to satisfy the QoS constraints of the desired connection. Further, most QoS routing algorithms select from among all feasible paths or trees; that which will optimize use of network resources. This optimization may be based on minimization of an actual dollar cost, or cost as measured by buffer or bandwidth utilization. This cost is measured as a sum of the costs of all of the links that form the path or tree. [2, 4]

QoS is dependent on the establishment and maintenance of a reliable connection that meets certain requirements. For unicast communication, the connection consists of a network path between two end users, and the routing problem consists of finding the best feasible path from source node s, to destination node t, such that QoS constraints C, are satisfied. For multicast, a multicast tree, emanating from the sender to all desired receivers, must be established. Here, the routing problem is to find the best feasible tree from source node s, to the set of destination nodes R, such that QoS constraints C are satisfied. In either case, the provision of QoS requires that connections satisfy a set of constraints. These constraints include the following:

1) Link Constraint – Specifies a restriction on links along a network path. For example, a restriction may be placed on a minimum bandwidth that is available on the links that compose the path of a unicast connection.
2) Path Constraint – Specifies the end-to-end QoS requirement (e.g. maximum delay) on a single path.
3) Tree Constraint – Specifies the QoS requirement for an entire multicast tree. A restriction (e.g. maximum delay) may be placed on a maximum end-to-end delay from the sender to any receiver in a multicast tree. [2]

4 QoS ROUTING COSTS

While there is abundant research supporting the tremendous potential of QoS routing, doubts linger about whether the intended improvement in quality and performance truly will offset the added costs. These added costs are composed primarily of: (1) increased computational cost, and in support of this, (2) increased volume of protocol traffic. These two QoS routing costs are listed in Table 1, along with their components and the respective tradeoffs. QoS routing requires an increased sophistication and frequency of path selection computations, as well as an increased volume of protocol traffic. This increased protocol traffic is the result of an increased dependence on distributing updates across the network on the state of network resources (e.g. available link bandwidth) – which are the basis for selecting the optimal path. [1]

Table 1 QoS Routing Costs

Cost	Tradeoff
Computational Cost	
Path Selection Criteria	Computational complexity vs. ability to identify best, cheapest path.
Trigger for Path Selection Computations	Per request computational cost vs. "goodness" of selected paths.
Flexibility in Supporting Alternate Path Selection Choices	Allowing for inaccurate information vs. resultant computational cost.
Protocol Overhead	
Triggers for Network State Updates	Volume of updates vs. accuracy of state information.
Update Contents	Adding unnecessary traffic vs. eliminate future updates. (A node's update message may include: specific link or all links/ actual or quantized value)
Bandwidth	Allotted bandwidth (and QoS) vs. amount of network traffic

In contrast to increased computational costs, which can be mitigated by faster processors and larger memories, increased protocol overhead presents a more complex and multifaceted problem with no easy remedy. Protocol overhead is therefore often considered the major stumbling block to the realization of QoS routing. Protocol overhead components include higher costs associated with: bandwidth, storage, update processing and context switching. [1]

5 QoS ROUTING: LINK ROUTING and PATH ROUTING

Routing problems may be characterized by the nature of the QoS metric of interest. The state of a path for a metric of interest is determined by one of the following two methods:

1) Identification of some bottleneck value of the metric along a proposed path.
2) Summation of values of the metric for each link along the proposed path.

For example, if a desired path is constrained to meet some minimum residual bandwidth requirement, the link with the least available bandwidth along a proposed path defines the available bandwidth for that entire path. There will be one or more links along a path that presents a bottleneck, and that bottleneck value will therefore characterize the available bandwidth for that path. In addition to residual bandwidth, another QoS metric that fits this category is residual buffer space. For QoS metrics of this type, two basic routing problems can be identified:

1) Link-optimization routing – Find a path with the largest bandwidth (or other desired QoS metric) on the bottleneck link.
2) Link-constrained routing – Find a path with a bottleneck bandwidth (or other desired QoS metric) above a certain value. [2]

Solution of the link-optimization routing problem is based on use of Dijkstra's algorithm or the Bellman-Ford algorithm, while the link-constrained routing problem can be reduced to the link-optimization routing problem[1]. [2]

For some QoS metrics, determination of the state of the path involves summation over all component links of the path. Such metrics include delay, jitter and cost. For example, the delay associated with a path, is simply the sum of the delays of each of those links. For QoS metrics of this type, two basic routing problems can be defined:

1) Path-optimization routing – For, example, find a path with minimal total cost.
2) Path-constrained routing – For example, find a path where the delay is less than some specified value. [2]

Both of these problem classes are directly solvable by Dijkstra's algorithm or the Bellman-Ford algorithm. The above-identified four routing classes form the components of more involved routing problem classifications. For example, a set of problems may be identified, called link-constrained path-optimization routing problems, in which the optimal path has the least amount of delay and some minimum amount of bandwidth. [2]

6 QoS ROUTING PERFORMANCE

Directly related to the above mentioned two main costs associated with QoS routing (computational cost and protocol overhead), are the two main determinants of the degree that a particular approach to QoS will increase network efficiency. These determinants, which accordingly figure prominently in the many proposed solutions to QoS are:

1) The quality of the information regarding network resources that supports the path selection algorithm.

[1] Treatments of Dijkstra's algorithm and the Bellman-Ford algorithm include:
http://www.scit.wlv.ac.uk/~jphb/comms/routing.html
http://ciips.ee.uwa.edu.au/%7Emorris/Year2/PLDS210/dijkstra.html
http://www.isat.jmu.edu/common/coursedocs/ProgrammingContest/shortestpaths.pdf
http://www.tutor.ms.unimelb.edu.au/dijkstra/island.html

2) The path selection algorithm. [1]

Good performance of a QoS routing algorithm is dependent on having accurate, up-to-date network state information, which is extremely difficult to do in the very dynamic, expanding environment of today's Internet. The number and diversity of QoS path selection algorithms reflects the many different types of problems that must be confronted in providing QoS. For example, distributed applications have very diverse QoS constraints on parameters including delay, jitter, loss ratio and bandwidth. The problem of simultaneously satisfying all of these constraints may be intractable. However, as is illustrated below, progress has been made in sidestepping the seeming intractability of some multiple constraint problems. [1]

Another issue to be confronted is that QoS traffic shares the same network resources with best-effort traffic, complicating efforts to optimize performance. Solutions must be developed that will allow the distribution of the two types of traffic to be independent, while ensuring that QoS traffic does not crowd out best-effort traffic. It should also be pointed out that the performance of QoS routing is also strongly dependent on the network topology and high-level admission control policies. [1]

7 QoS ROUTING and NETWORK STATE INFORMATION

Because routing involves selecting the best feasible path, with network state information providing the basis for making this decision, it is imperative that the state information be collected and stored so that it is accurate and up-to-date. Effective feasible path selection is strongly dependent on the degree that the actual, current state of network resources is reflected by the state information provided to the path selection algorithm. [2]

There are three main approaches to collecting and storing state information:

1) Local State – The state of a particular node, which includes queuing delay, propagation delay, and residual bandwidth of its links.
2) Global State – The combined local states of all nodes. This may be accomplished through the use of either a link-state protocol or a distance-vector protocol.
3) Aggregated Global State – To any particular node, specific, detailed information on adjacent nodes is available, and only aggregate information is available on non-adjacent nodes. To achieve scalability, network information is aggregated, and decisions are made based on exchange of information within a hierarchal model of the network. Information on adjacent nodes are aggregated into a hierarchy of levels of abstraction. [2]

Classified based on how the state information is maintained and how path selection is conducted, the following routing strategies may be identified:

1) Source Routing – Each node maintains the complete global state, including network topology and the state information of every link. A feasible path is locally

computed at the source node. A control message is then sent along this path, and a link-state protocol is used to update the global state at every node.

2) Distributed Routing – Path selection is conducted using a distributed computation, in which state information stored at each node is used collectively and control messages are exchanged between nodes.
3) Hierarchal Routing – A hierarchy of clustered nodes is established, and each node maintains an aggregated global state. Source routing is used to select a feasible path, along which a control message is subsequently sent. [2]

These algorithms will be further examined in the following sections.

8 QoS ROUTING: SOURCE ROUTING

In addition to maintaining a global state at each node, source routing algorithms characteristically transform the routing problem to a shortest-path problem, which is then solved by Dijkstra's algorithm or the Bellman-Ford algorithm. By transforming a distributed problem into a centralized one, source routing presents a simple approach that assures loop-free routes and is typically easy to implement, evaluate, debug and upgrade. Some NP-complete routing problems can be much easier to resolve using a centralized approach.[2] [2]

However the apparent simplicity of source routing has many drawbacks. For example, the need to frequently update the state information at each node, leads to excessively high communication overhead for large-scale networks. Also, due to propagation delay of state messages as well as large overhead, the link-state algorithm can only provide approximate global state information. Therefore, QoS source routing may be too cumbersome and imprecise to find a feasible path. Furthermore, the computation overhead at the source is excessive, especially in the case of multicast routing or when multiple constraints must be met. Thus, the cost of maintaining detailed state information at all nodes becomes impractical for large networks, severely limiting the usefulness of source routing. [2]

An example of a unicast source routing algorithm is the Wang-Crowcraft algorithm. This algorithm finds a bandwidth-delay-constrained path by Dijkstra's shortest-path algorithm. First, links with bandwidths less than the requirement are eliminated, and then the shortest path in terms of delay is selected. The path is feasible only if the delay constraint is satisfied. The Wang-Crowcraft algorithm is summarized in Table 2, along with other unicast routing algorithms. [2]

[2] A treatment of NP-complete problems may be found at: http://bit.umkc.edu/demo/course/cs352/lessons/t3528/learning/lectures/np-c/title.html

Table 2 Unicast Source Routing Algorithms

Algorithm	Description
Wang-Crowcraft	Algorithm finds a bandwidth-delay-constrained path by Dijkstra's shortest-path algorithm. 1) links with bandwidths less than the requirement are eliminated 2) shortest path in terms of delay is selected 3) path is feasible only if delay constraint is satisfied [2]
Ma-Steenkiste	Transforms the NP-complete problem of finding a path satisfying bandwidth, delay, jitter, and buffer space constraints into a problem of polynomial complexity, solvable a modified version of the Bellman-Ford algorithm. This was made possible by the discovery of a functional relationship between the above parameters and the selected path and the traffic characteristics. [2] [8]
Guerin-Orda	Guerin and Orda studied the bandwidth-constrained and delay-constrained routing problems with imprecise network states and show that, using a shortest-path algorithm, they can find the feasible path that is most likely to accommodate the requested bandwidth. [2]
Chen-Nahrstedt	Chen and Nahrstedt proposed a heuristic algorithm for the NP-complete, multi-path-constrained (MCP) routing problem (e.g. both cost and delay are constrained). This algorithm reduces the MCP problem to one solvable in polynomial time, which in turn may be solved by an extended Dijkstra's or Bellman-Ford algorithm to find a feasible path. [2]
Awerbuch et. al.	Awerbuch et. al. proposed a throughput-competitive routing algorithm for bandwidth-constrained connections, which seeks to maximize network throughput. [2]

9 QoS ROUTING: DISTRIBUTED ROUTING

By distributing path selection among all nodes between the source and destination, distributed routing is more scalable than source routing. The routing response time is typically shorter and multiple paths may be searched simultaneously, increasing the chances of success. With most existing distributed routing algorithms, each node maintains global network state, and routing decisions are made on a hop-by-hop basis. Because distributed routing shares the characteristic of requiring each node to maintain global state, it also shares the above-mentioned disadvantages of source routing. [2]

Table 3 Distributed Routing Algorithms

Algorithm	Description
Wang-Crowcraft	A hop-by-hop distributed routing scheme: 1) Complete global state is maintained at each node. 2) Every node pre-computes, and updates periodically, a forwarding entry (the next hop) for every possible destination. The calculation is done using a modified Bellman-Ford algorithm, which finds the shortest-widest path. 3) The routing path consists of the forwarding entries, for the desired destination, at all of the intermediate nodes. [2]

Algorithm	Description
Salama et. al.	A distributed heuristic algorithm, offering an efficient solution to the NP-complete, delay-constrained least-cost routing problem: 1) A cost vector and a delay vector are maintained at every node via a distance-vector protocol. The vectors contain the next node, for each destination, on the least-cost/delay path. 2) A control message is sent from the source toward the destination. 3) At each intermediate node, one of two alternate outgoing links is chosen. One link (i,j) is the least-cost path, and the other (i,k) is the least-delay path. 4) The least-cost path is chosen, so long as there is a feasible path between node j and the destination – which does not violate the delay constraint. 5) Loops are removed by rolling back the routing process to a node where the least-cost path was followed, and then proceeding along the least-delay path. [2]
Sun-Landgendorfer	Similar to Salama et. al., but loops are avoided, not detected and removed. 1) A control message is sent from the source to construct a routing path. 2) The path is constructed along the least-delay path until a node is reached where the delay of the least cost path satisfies the delay constraint. 3) The path then is completed form that node to the destination, along the least least-cost path. [2]

10 QoS ROUTING: HIERARCHAL ROUTING

By only requiring each node to maintain a partial global state, hierarchal routing confronts the scalability issue and combines advantages of both source routing and distributed routing. Source-routing algorithms are used at each hierarchal level to find feasible paths based on an aggregated state. As in distributed routing, routing computation is shared by many nodes. The increased scalability comes at the cost of a loss of precision. Aggregation of data means that decisions must be made based on incomplete, imprecise information, significantly impairing the ability to ensure QoS routing. Problems associated with aggregation of data are compounded when multiple QoS constraints must be satisfied. Further, solutions do not yet exist for how to aggregate information in a group of nodes and effectively account, for example, for the case where path A has superior bandwidth availability and paths B and C have smaller delay. [2]

11 QoS and SECURITY

As previously mentioned, the requirements for providing QoS include placing limits on the allowable end-to-end delay, delay variation, and the packet loss rate. Some proponents of QoS advocate the inclusion of security as an additional primary requirement. After all, on top of existing network security concerns, implementation of a new network capability such as QoS will bring new vulnerabilities. Knowing this, security should be integrated into the design from the very beginning. However, QoS mechanisms have typically been developed without addressing security issues. [4]

Consideration that an organization's productivity is directly tied to the performance of its computer network also reveals the importance of both QoS and security, and the inextricable link between them. Information technology systems must be available, high-performance and secure. These needs place demands on both the internal staff and on service providers. The internal staff must ensure that the internal network is hardened, and further protected using a defense in depth approach that layers complimentary, overlapping technologies, which form a fortified, consistent overall security architecture that fully enforces a sound, comprehensive security policy. Service providers must be prepared to provide levels of reliability, performance, and security that satisfy their customer's requirements. [7]

Two types of vulnerabilities and attacks associated with QoS may be identified:

1) Control Flow Attacks – acts at the connection level, interferes with signaling/ control protocol for network resource reservation and connection setup.
2) Data Flow Attacks – acts at the packet level, interferes with the dataflow in such a way that the reserved resources are not efficiently used. [4, 12]

The goal of providing QoS must include installing mechanisms to prevent, detect, and recover from both control and data flow attacks. For example if preventative mechanisms are not in place when network resources are allocated to provide QoS, a malicious user could reserve as many resources as they please, and delete or alter an existing reservation. A malicious user may steal resources, as well as deny the use of resources to other users. [4, 12]

To prevent such attacks, provision of QoS must include:

- Authorization and authentication mechanisms.
- Enforceable resource allocation mechanisms.
- Monitoring mechanisms that effectively detect and respond to attacks. [4, 12]

Each of these mechanisms is discussed below.

12 QoS AUTHORIZATION and AUTHENTICATION MECHANISMS

A typical IP network may be considered to be composed of access networks and one or more core networks. Access networks have relatively higher bandwidths and traffic volumes than core networks. Accordingly, different technologies and protocols are used in each. Resource Reservation Protocol (RSVP) has been proposed as the fundamental means of ensuring QoS in access networks, while Differentiated Services (DiffServ) is the primary protocol proposed for ensuring QoS in core networks. RSVP provides the signaling that allows an application to reserve network resources for individual connections (microflows) from source to destination. QoS is provided through RSVP-enabled routers that schedule and prioritize packets. DiffServ allocates resources and provides QoS for aggregates of connections (flows), while requiring less overhead and offering more scalability than RSVP. [5, 9, 10, 11]

Steps involved in allocating network traffic priorities for a temporary high-bandwidth connection (e.g. a connection supporting IP telephony) include:

- A reservation message is transmitted using RSVP from source to destination.
- The message is intercepted at each intervening router, where a policy request is made using COPS. This request is then forwarded to a policy server.
    - COPS is a proposed standard protocol for exchanging network policy information between a policy decision point and policy enforcement points.
        - policy decision point - server controlled directly by the network administrator who enters policy statements about which kinds of traffic (voice, bulk data, video, teleconferencing, and so forth) should get the highest priority.
        - policy enforcement points – routers or switches that implement the policy choices as traffic moves through the network.
- The policy server returns an admission control decision.
    - If the decision is positive, router reserves resources for requesting connection, and the RSVP reservation message is forwarded to the next router.
    - If the decision is negative, message is propagated to user and no reservation is established. [9, 10]

Not included in this scheme is a mechanism to prevent inefficient use and wasting of resources, as a result of users sending unnecessary or bogus requests. Authorization and authentication mechanisms to protect QoS control signaling are required to distinguish between users to ensure that users limit their requests, resources are allocated fairly, resources are not wasted, and DoS attacks are prevented. To this end, the following has been recommended:

- Users must be authorized through an authentication server before being allowed to reserve resources. Once authenticated, the authentication server issues an authentication ticket that verifies the ability of the user to pay a certain price.
- The ticket is included in reservation requests sent from an authenticated user to a policy server.
- The policy server uses the ticket to verify that the request is authorized.
- The response to the user includes the price being charged for that resource. [4, 9]

This solution has been demonstrated to support the continually evolving network model of today's Internet. This solution allows for multiple carriers, multiple service providers and allows businesses to cooperate to provide services to their customers without divulging secrets or relinquishing control. [9]

13 QoS and RESOURCE ALLOCATION MECHANISMS

Resource pricing is a means for efficiently and fairly allocating limited resources, based on the willingness and ability of users to pay for those resources. Also, by providing a

means for authorizing and controlling resource allocation, resource pricing prevents control flow attacks. Resource pricing accommodates both:

1) Users seeking basic service at a low price.
2) Users seeking predictable service, where they are guaranteed a fixed resource amount. [4]

To accomplish this, each resource is split into "reservable" and "available" categories. Resource prices are separately established for each category, and periodically updated, with the goal of making supply equal demand. Each user specifies whether they want either lower priced service or predictable service. Through resource pricing, users who are willing to pay a higher price are guaranteed a predictable quality of service. At the same time, leftover unreserved resources are made available at a lower price, to be shared by users with no requirements that resources be provided in a predictable manner. [4, 9]

To ensure that allocation of resources remains stable for those applications desiring stability, prices – once updated – must hold for a sufficient period of time. However, this dulls the ability of the feedback process to keep the demand for resources matched to the supply, and requires that demand be predicted. As a compromise, network resources are divided into two pools: (1) resources in one pool have stable prices, based on demand that is predicted; and (2) resources in a second pool will have prices that are free to vary, and are based on current demand. [4, 9]

A first step of resource pricing is to provide a means for allocating resources through the establishment of a price per unit of resource. An owner of network resources seeks to:

- Maximize revenues and ensure that use of the resource approaches 100%.
- Ensure that resources are used efficiently.
- Ensure that, to a good degree, the desired quality of service is delivered to each customer.

To accomplish this, the price of the resource is determined via *demand-based pricing*, which involves: (1) measurement of demand, (2) price calculation, and (3) price distribution. These three steps compose a feedback loop that may be allowed to iterate until equilibrium is achieved, and therefore, a desired resource utilization level is reached. This iterative process is called a tatonnement process. It has been shown that when equilibrium is reached, no user will be able to receive a greater amount of resource, without another user receiving less. [4]

Further details of demand-based pricing include:

- If there is only one user for a resource, equilibrium is attained when a certain price is reached such that demand for the resource by that user equals the supply. [4]
- If there are multiple users competing for one resource, equilibrium is achieved when a certain price is reached such that the total demand equals the supply, and at that point, the amount of resource allocated to each user is proportional to

their budgets. Here, a "budget" is a weighting factor used to allocate resources. [4]

- The case where multiple users compete for multiple resources is more involved, and its solution is addressed below. [4]
- Note that by convention, any particular user is assumed to require the same amount of each resource. For example, when dealing with bandwidth allocation, each link of a connection requires the same amount of bandwidth. [4]
- A limiting resource is defined to be a resource demanded by a particular user whose equilibrium price is greater than that of any other resource demanded by that user. [4]
- Fairness – refers to how a resource allocation policy determines how a user with a limited budget, allocates that budget. [4]
- max-min fair – Refers to a type of fairness advocated for network congestion control, where users who are resource-limited by the same resource (share the same limiting resource), share that resource equally. [4]
- weighted max-min fair – Refers to a more general type of fairness, in which users u and v have weights (budgets) w_u and w_v, and share the same limiting resource. In this case, the ratio of resources allocated to u and v is w_u / w_v. [4]
- Both max-min and weighted max-min fairness can be attained with pricing. Prices may be calculated as follows:
    - Each resource executes a tatonnement process to compute its equilibrium price.
    - Each user requesting resources is allocated an amount equal to the quotient of the budget for that user, divided by the price of the limiting resource. As previously discussed, each user is allocated the same amount of each resource demanded by that user.
      This is the same as saying that the price charged a particular user is equal to the price of the most expensive resource that user requires, and the amount is proportional to that user's budget.
    - It can be shown that this system achieves a weighted max-min fair solution to resource allocation. Also, this system reduces to the max-min fair system when the budgets of all users are the same. [4]
- The pricing method presented above may be shown to be able to achieve the fairness goals of: proportional allocation, equitable allocation and utility-maximizing allocation.
    - Proportional allocation – (used in TCP congestion control) a resource allocation is fair if it is in proportion to the users willingness to pay. Such an allocation of resources guarantees economic efficiency, since users' utilities (benefits) are maximized. [3, 4]
    - Equitable allocation – occurs when users who share the same limiting resource, place the same value on that resource.
    - Utility-maximizing allocation – results in the maximum aggregate utility of all possible allocations. [4]
- Benefits of the above pricing method include:
    - Flexibility – the same method can support a variety of fairness goals or policies

- Robustness – adapts quickly to changes, has low overhead, and is efficient
- Scalability
- Rapidly converges under dynamic and realistic conditions.
- No assumptions are made about the type of traffic allowed. [4]

14 QoS and DETECTION OF PACKET DROPPING ATTACKS

Packet dropping attacks are one of the more difficult DoS attacks to handle, and involve the malicious dropping of packets from a flow. The packets that are dropped may include both random and important packets. The number of packets dropped is limited in order to hinder detection. Therefore the effects of the attack may be hard to differentiate from normal packet dropping caused by congestion. [13] An attacker may carry out this attack by compromising or congesting intermediate routers. Detection of such an attack may be accomplished by developing the ability to detect anomalous activity by comparing an observed distribution with an established statistical baseline of an expected distribution. Statistical analysis focuses on distinguishing malicious packet dropping from that due to normal TCP behavior. For best results, a combination of multiple statistical measures should be used, including:

- Position Measure – the position or sequence number of reordered packets
- Delay Measure – the average packet delay
- Number Measure – the number of packets reordered [4,13]

15 CONCLUSION

"Freedom in a commons brings ruin to all." This was the conclusion reached by Garrett Hardin in *The Tragedy of the Commons*. Not being able to trust the herdsmen to collectively act for the good of all, Hardin advocated, "mutual coercion, mutually agreed upon" as a means for fairly and efficiently managing the commons. Efficient, fair and secure allocation of the limited resources of the Internet is becoming more imperative, and the many challenges associated with bringing this about are being actively addressed. In this paper we have reviewed and evaluated many of the problems, solutions, security issues and future challenges associated providing effective, secure QoS. We have seen that considerable progress is being made towards providing specified and desired levels of secure service to network traffic, ensuring efficient use of limited network resources, and also protecting against misuse of network resources.

REFERENCES

[1] Apostolopoulos, G., Guerin, R., Kamat, S., Tripathi, S.K., "Quality of Service Based Routing: A Performance Perspective," *Computer Communication Review*, ACM SIGCOMM, Volume 28, Number 4, October 1998. http://citeseer.nj.nec.com/cache/papers/cs/8682/http:zSzzSzwww.cs.umd.eduzSzuserszSzgeorgeapzSz.zSzpaperszSzsigcomm.pdf/apostolopoulos98quality.pdf, Accessed 20 Nov 02.

[2] Chen, S., Nahrstedt, K., "An Overview of Quality of Service Routing for Next-Generation High-Speed Networks: Problems and Solutions," *IEEE Network*, November/December 1998, Pages 64-79, http://citeseer.nj.nec.com/cache/papers/cs/24920/http:zSzzSzwww-sal.cs.uiuc.eduzSz~s-chen5zSzresumezSzIEEENetMag98.pdf/chen98overview.pdf, Accessed 20 Nov 02.

[3] Faulkner, M., "An Overview of Pricing Concepts for Broadband IP Networks," *IEEE Communications Surveys*, Vol. 3, No. 2, April 2000, http://www.comsoc.org/livepubs/surveys/public/2q00issue/falkner.html, Accessed 20 Nov 02.

[4] Fulp et. al., "Preventing Denial of Service Attacks on Network Quality of Service," Jan 2001, http://arqos.csc.ncsu.edu/papers/2001-01-discexII.pdf, Accessed 20 Nov 02.

[5] Grigonis, R., "Working the QoS Puzzle," *Computer Telephony*, 05 Jan 01, http://www.cconvergence.com/article/CTM20001221S0001 , Accessed 20 Nov 02.

[6] Hardin, Garrett, "The Tragedy of the Commons," *Science*, (162), 1968, http://www.dieoff.org/page95.htm, Accessed 20 Nov 02.

[7] Irvine, Cynthia; Levine, Timothy, "Quality of Security Service," 2000, http://www.cise.ufl.edu/~nemo/security/irvine-qoss.pdf, Accessed 20 Nov 02.

[8] Ma, *Q.*, "Routing Traffic with Quality-of-Service Guarantees in Integrated Services Networks," Carnegie Mellon University, January 1998, http://www-2.cs.cmu.edu/afs/cs.cmu.edu/user/qma/www/nossdav98.pdf, Accessed 20 Nov 02.

[9] Reeves, D.S., Wu, S.F., Stephenson, D., "Protecting Quality of Service Against Denial of Service Attacks," Presentation, DARPA FTN Program PI Meeting, St. Petersburg, FL, 17 Jan 2002,

http://arqos.csc.ncsu.edu/talks/2002-01-ftn-meeting.pdf, Accessed 20 Nov 02.

[10] searchSystemsManagement.com, http://whatis.techtarget.com/definition/0,,sid9_gci213825,00.html, Accessed 20 Nov 02.

[11] Webopedia, http://www.webopedia.com/TERM/R/RSVP.html , Accessed 20 Nov 02.

[12] Wu, T-L., et.. al., "Securing QoS: Threats to RSVP Messages and Their Countermeasures," Advanced Research Projects Agency, U.S. Department of Defense, 18 Dec 98, http://arqos.csc.ncsu.edu/papers/1999_10_iwqos99.pdf, Accessed 20 Nov 02.

[13] Zhang, Xiaobing; et. al., "Malicious Packet Dropping: How It Might Impact the TCP Performance & How We Can Detect It," 17 Nov 2000, http://www.nmsl.cs.ucsb.edu/~ksarac/icnp/2000/papers/2000-24.pdf , Accessed 20 Nov 02.