Billing Users for TCP

R.J. Edell, N. McKeown, P.P. Varaiya
Department of Electrical Engineering and Computer Sciences
University of California, Berkeley
{edell, nickm, varaiya}@eecs.berkeley.edu

***Abstract*- This paper introduces a system for billing users for their TCP traffic. This is achieved by postponing connection establishment while the user is contacted. Connection establishment continues after the user has confirmed that he is prepared to pay. The user is presented with cost and price information, therefore, the system could be used for cost recovery and to encourage efficient use of network resources. No changes to existing protocols or applications are required. A preliminary analysis of statistics collected from a one day trace of traffic between the University of California, Berkeley and the rest of the Internet demonstrate that such a billing system is practical and introduces acceptable latency.**

I. Introduction

An estimated 20 million people are reachable by electronic mail. The NSF funded backbone network "NSFNET" is frequently used to transport electronic mail messages[2], serving a multitude of academic and commercial organizations. Currently, the annual NSF subsidy for NSFNET (and various regional networks) is about $20 million. The estimated total cost of the Internet is about $200 million per year. It is expected that this NSF subsidy, and hence NSFNET, will soon be gone; that the Internet will then be run entirely by commercial and non-profit enterprises; and that these enterprises will recover their costs through charges and pricing schemes.

If all users within an organization generated similar amounts of traffic, then cost recovery could be equitably and simply achieved by dividing the cost equally among users. We have shown that hosts and users generate widely varying amounts of WAN traffic [5], demonstrating that to recover costs fairly from this diverse population of users requires the capability to meter individual user traffic. In addition, presenting users with prices and charges can encourage more efficient use of network resources [7]. In this paper, we present a billing system that can be used to meter users' wide area TCP (Transmission Control Protocol [4]) traffic, involving the users in the decision to consume resources and making them accountable for the traffic that they generate.

Several network billing systems have been previously proposed. An excellent discussion of methods, costs and benefits of usage metering as well as a proposed billing infrastructure is given in [12]. In [2], a flow-based accounting mechanism is defined based on IP data that flows between end hosts. But neither of these proposals include a mechanism for identifying and involving indi-

---

1. Research funded by a grant from Pacific Bell and the California State MICRO Program 93-152.

2. The NSFNET also transports other types of electronic messages.

vidual users. In 1990, Estrin et al. suggest some research topics towards usage billing and feedback. In particular, the instrumentation of current networks to determine the performance implications of usage metering is proposed. Furthermore, they highlight the problem of identifying and authenticating the end user.

Our billing system is designed to meter users' TCP connection-oriented protocol traffic [1],[4], which represents almost all of the traffic on the Internet. It is because a connection is established prior to the transfer of data that we are able to reliably identify the originating user and confirm that they are prepared to pay for their traffic. An important feature of this billing system is that it requires no modification to applications or the TCP/IP protocols.

A usage-based billing system in which the user is directly involved will likely affect the way in which the Internet is used. Indeed, several schemes have been proposed for using pricing to controlling congestion [8]. These schemes require that traffic is metered and that the user is involved in the feedback mechanism. However, it has not been previously shown that real-time metering of traffic for high-speed networks is feasible, or that it is practical to involve the user in the feedback.

A billing system that controls the access of individual users will also affect the way in which the Internet grows. For example, those users within an organization who require a higher bandwidth connection to the commercial data carrier can pay for and get preferential use of the improved service.

Billing users for their network traffic is a controversial subject [1],[14]. However, this paper does not intentionally argue for the widespread adoption of usage billing. Instead our aim is to demonstrate the technical feasibility of a billing system that involves users and makes them accountable for their traffic.

In Section II we provide a detailed description of the proposed billing system. In Section III we describe an preliminary feasibility study of the system, based on a trace of the traffic that leaves and enters the Berkeley campus. We draw some conclusions from our study in Section IV.

II. GENERIC SYSTEM DESCRIPTION

We believe that any billing system should have the following essential features:

- *No changes to existing Internet protocols* — Because of the huge installed base of end systems, bridges and routers we believe that it is crucial that an Internet billing system work with the existing Internet protocols. Specifically, the billing system should not require the use of any special option fields (for example, IP options or TCP options).
- *No changes to existing applications* — Many applications such as ftp, email, gopher and mosaic are in widespread use today and collectively contribute to most of the traffic on the Internet [3]. A billing system should be able to meter traffic for these and other existing applications without change.
- *User involvement* — If users are to be charged for their network usage, it is important that they be better involved in the decision to consume resources. This not only helps ensure credibility of the system, but also encourages efficient use of network resources. A billing system should

notify the user of the cost of network usage and require explicit user approval prior to consuming resources. If this approval is authenticated securely (e.g. with Kerberos [13]), then the billing system can be credible and users can be held accountable. The billing system can reinforce accountability by providing accurate on-line feedback to users as they consume resources. These requirements imply that the metering of traffic should be exact and, in particular, should not be based on traffic sampling.

The combination of (i) explicit and authenticated user approval, (ii) on-line feedback to users, and (iii) exact metering, leads to a closed-loop system. If this is coupled with an intelligent pricing policy, then the system can encourage the efficient use of network resources.

In addition, it would be desirable for a billing system to have the following features:

- *Provide on-line reporting of aggregate network usage* — This enables schemes that control network congestion based on global traffic measures. The control may be implemented using priorities or by a time varying pricing policy [8].
- *Allow continued sharing of information and resources* — The growth of the Internet can be attributed to applications that encourage the sharing of information and resources between remote sites. It would be advantageous if billing systems could cooperate to identify the user and bill them for their traffic.

A. Billing System Components

Figure 1 is a schematic diagram of the billing system. This figure shows a single administrative domain, connected to the outside world via a Billing Gateway (BGW) and a billed link. The administrative domain contains a collection of networks, users and hosts. The billing system controls users' access to the billed link by allowing or disallowing TCP connections and by metering users' TCP traffic once a connection has been established.

The components in Figure 1 fall into three categories. The first category contains only the Access Controller. The Access Controller coordinates the operation of the billing system. The detailed discussion of this operation is deferred until the next subsection.

The second category is user involvement. This contains three components: User Identification Daemon, User Name Server and Purchasing Agent. Together these components identify the originating user, obtain verification from the user and provide feedback to the user as resources are used.

- The User Identification Daemon (Useridd) is a system daemon process that runs on all multiuser machines within the administrative domain. Request messages are sent to the Useridd containing the unique connection identifier. The Useridd examines kernel data structures to determine and return the name of the user who originated the connection.
- The User Name Server is a standard network naming service which when supplied with a user name will return the location of the user's Purchasing Agent.

- The Purchasing Agent[3] is a user-level process responsible for verifying that the user is prepared to pay for the TCP connection. In its simplest form, the Purchasing Agent could just pop up a dialog box on the user's screen asking for a simple accept/reject response. Alternatively it may be configured to automatically respond to some or all of the requests that it receives from the Access Controller. For example, it could be configured by the user to: (i) automatically accept all connections below a certain price, but verify with the user before accepting more expensive connections; (ii) automatically accept all connections to specific well-known ports, but verify with the user before accepting others; or (iii) automatically accept all connections to certain destinations. The Purchasing Agent proves its authority to represent the user by including a Kerberos authenticator in its response.

The third category is metering. The two components in this category are the BGW and Billing Records.

- The BGW is a specialized router that meters TCP traffic.[4] The BGW maintains a table of all established connections; each connection entry is keyed by its unique connection identifier. This means that the connection tables maintained by the BGW may be much larger than IP routing tables. It also means that the key into the connection table is much larger: for an IP router the key would normally just be the 32-bit destination network address. For a BGW with TCP/IP the key is 96-bits. In Section III.D we will consider how large the connection tables need to be and compare several schemes for looking up entries in the connection tables.

  The BGW must understand TCP connections so that it can determine which connection record to update. New connections are detected when TCP messages are received for any connection that the BGW has not seen before. This includes connections that establish via other BGWs.

  The BGW must determine when connections have closed so that the entry in the connection table can be freed. Connections may close in a number of different ways and the TCP FIN messages that close the connection may travel via a different BGW. So that connections can be removed from the tables in a timely manner, the BGW times out connections that are inactive for a long period. If the connections become active again, they are added back into the table.

- The Billing Records are responsible for maintaining records of metered connections and for providing on-line feedback to the user as the connection progresses.

B. Billing System Operation

The basic operation is as follows: when a user attempts to establish a TCP connection with the outside world, the BGW postpones the establishment of the connection while it tries to identify the originating user. The system contacts the user, verifying that they want to establish the connection and that they agree to pay for it. If the user accepts the connection, the normal connection establishment is allowed to continue and the BGW will begin metering this connection's traffic.

We will describe the operation in detail by way of an example and by referring to the communications marked on Figure 1 and the corresponding time diagram in Figure 2b. The timeline in

3. For security purposes, this process is usually run on the host at which the user sits.

4. The BGW must also perform the usual IP routing functions.

Figure 2b is an extension of Figure 2a showing all the communications involved in setting up a connection between the originating local host and the remote host.

In the example, the user sits at Host A but is logged in to Host B. The user's application on Host B attempts to establish a TCP connection to a remote host. Considering each communication in turn:

1) Host B initiates the connection by sending a TCP SYN message. The BGW recognizes the TCP SYN message as an attempt to establish a new connection to the outside world. The BGW holds onto the TCP SYN message while it determines whether or not the connection should be allowed and access granted to the billed link. This is achieved by communications (2-11) to identify and contact the user, determine whether the connection will be allowed and to set up the necessary state for metering and billing records. The connection is always referred to by its unique identifier: [(source (address,port), destination (address,port)].

2) The BGW contacts the Access Controller which is responsible for determining whether the connection should be allowed. It achieves this by communicating with each of the components in turn.

3,4) The Access Controller asks the User Identification Daemon (Useridd) to identify the user.

5,6) The Access Controller asks the User Name Server to locate the user's Purchasing Agent.

7) The Access Controller asks the user's Purchasing Agent to verify that the user wishes to establish and pay for this connection.

8,9) The Purchasing Agent may be configured to respond automatically on behalf of the user, or may request the explicit authorization of the user by means of a dialog box on the user's screen.

10) The Purchasing Agent responds to the Access Controller authenticating its reply using Kerberos.

11) If the user confirms that they wish to establish and pay for the connection, the Access Controller tells the BGW to allow access to the billed link and meter the connection.

12) The BGW creates an entry for this connection in its connection tables and forwards the TCP SYN message towards the remote host.

Now that the BGW knows to allow this connection, all future messages for this connection in either direction will be forwarded without further delay. This means that when remote host responds with a TCP SYN+ACK, the BGW will forward the message, updating its connection table entry.

C. Simpler Cases

There are a number of possible simplifications that could be used to reduce the number of communications described above. Each simplification corresponds to bypassing communications shown in Figure 2b, "S1" - "S3". The simplifications reduce the number of communications involved in establishing a connection, thus reducing the latency seen by the user.

1) *No user interaction* — Do not explicitly involve the user in the connection acceptance. This was discussed earlier and would be achieved by configuring the Purchasing Agent to accept connections automatically. The advantages are that it would make the connection setup faster and would not burden the user with involvement in every connection.

2) *No Purchasing Agent interaction* — Configure the Access Controller so that connections from some users or for some applications are accepted automatically. This eliminates communications (5-10).

3) *By machine billing* — Configure the Access Controller so that connections from some machines are accepted automatically. This would be used when a machine is used by a single user, or if it has been agreed that the traffic will always be paid for regardless of the user. In fact, this would be necessary for machines that cannot support the Useridd process (this would include single-process personal computers). In this case, we eliminate communications (3-10).

It is important to note that both simplifications 2 and 3 compromise the security of the billing system as neither obtains authenticated feedback from the user.

D. Complications

We believe that the billing system described above is feasible and implementable. However, there are several things that can make the implementation more difficult, or make the billing model less appropriate. We consider some of these below.

- *Time varying route* — The packet-switched Internet allows datagrams within a connection to be delivered via different routes. If the change is due to load balancing, we can expect the price for delivery to be unchanged. But if there is a change in topology, the price may change and should be communicated back to the user.
- *Connections on behalf of, but not originated by a user* — The main reason for involving the user is to make them accountable for their traffic. The basic system achieves this by billing the originator of a TCP connection. However, it would be more appropriate to bill the beneficiary of the connection. In practice, the *originator* of a connection is not necessarily the *beneficiary*. For example, if a process on one host originates a connection so that it can transfer data to or from a remote host, who is the beneficiary of the transfer?

III. Feasibility Study

In this section, we will use the results from a detailed study to demonstrate that our billing system is feasible even for large campus networks. Our results are derived from a trace of network traffic on the Berkeley campus FDDI backbone network.[5]

Most of our billing system’s complexity is connection related. Therefore, our feasibility study measures several key statistics about TCP connections. These statistics are: connection setup

5. During the month of September 1994, Berkeley contributed the 13th largest amount of WAN traffic to the NSFNET out of over 22,000 registered networks [11].

time, connection setup intensity, active connection count, and complexity of searching the connection table. Under our billing system, connection establishment is delayed while locating the user and then verifying that they will pay; therefore, we measured connection setup *latency* without the billing system and estimate the additional connection setup latency introduced by the billing system. Several components of our billing system are activated once for every connection; therefore, we measured connection setup intensity to determine the required *throughput* for these components. The BGW maintains a table of active connections; therefore, we measure the evolution of connection count to determine the required *size* of the connection table. The connection table is frequently searched; therefore, we measure the performance of different search methods to determine the *complexity* of connection table lookup.

The BGW processes every datagram for metering; therefore, our feasibility study measures datagram intensity to determine the required datagram *throughput* for the BGW.

A. About the trace

The Berkeley campus community is composed of approximately 40,000 students, faculty and staff members. The campus network interconnects approximately 22,000 hosts of which 3,172 hosts participated in WAN TCP connections during the study period. The campus network is connected to the Internet by two T1 (1.5Mbps) links and a single boundary router. We consider everything on the other side of this router as the “outside world.”[6]

Our trace was obtained by logging all TCP/IP headers that appeared on the Berkeley backbone network. A 50ns timestamp was attached to each log entry. The analysis presented in this paper considers all WAN connections in the 24 hour period commencing at midnight Thursday September 15, 1994. During our study period, we measured 18 gigabytes of WAN traffic in 98 million datagrams. Of this WAN traffic, 94%of the bytes and 92% of the datagrams were for TCP traffic. Of this WAN TCP traffic, 83% of the bytes and 80% of the datagrams were from multi-user hosts (see Table 1). Figure 3 shows how the average rate of inbound and outbound datagrams changed over our study period.

Host Type	Number of Hosts	Per Host Averages			Percentage in Category	
		Connections	Datagrams	Bytes	Datagrams	Bytes
Multi-user Computers	1,541	273	35,768	7,183,160	80.3%	83.2%
Personal Computers	629	25	3,406	740,954	3.1%	3.5%
unknown	972	103	11,719	1,814,937	16.6%	13.3%

TABLE 1. Amount of TCP WAN Traffic by Host Type. During the trace period there were approximately 530,000 connection attempts of which only 370,000 were successful.

6. This definition of “outside world” includes some geographically and topologically nearby sites such as Lawrence Berkeley Laboratories.

B. Additional Connection Setup Latency

Figure 2a and Figure 2b indicate the additional communication introduced by our billing system. Inevitably this additional communication will increase connection setup time. The relative significance of this additional delay depends on the distribution of connection setup times. To estimate the relative significance, we measured the distribution of setup times for outbound connections without the billing system. Our results are shown in Figure 4, indicating that 80% of the connections took between 25ms and 520ms to establish, with a median time of 107ms, and an average time of 411ms.

To estimate the average time for the rest of the connection setup overhead, we measured the average response time of the “User Identification Daemon” (Useridd) implemented as a user-level process. Over the period of our trace, the average response time was 54ms. We shall assume that each of the other three request/reply pairs will take approximately the same amount of time resulting in an average additional delay of 216ms. Therefore, the average connection setup time with the billing system would be 627ms, an increase of approximately 50%.

C. Connection Throughput Requirements

This section discusses the throughput requirements for the functional blocks. Most of the billing system functional blocks are activated once per connection setup, making the peak intensity of connection setup an important parameter. Figure 5 shows the distribution of connection setup intensities over one second intervals. The peak rate was 52 connection setups per second, and 99.9% of seconds had 31 or fewer connection setups. Therefore, the throughput requirements for the functional blocks can be easily achieved.

D. Size of Connection Table

An entry must be maintained in the BGW connection table for all active connections. For the billing system to be feasible, these tables must not be too large. We found that inactive connections must be timed-out to control the size of the connection table. This is because some types of hosts do not properly shut-down TCP connections. Figure 6 shows the evolution of connection count if connections are timed-out after 30 minutes. With this time-out policy the maximum connection count is below 1,100 entries.

E. Complexity of Connection Lookup

The BGW meters every TCP message that it forwards, requiring it to perform a lookup in the connection table whenever a message arrives. It is therefore important that connection table entries are located quickly. Entries in the table are keyed by a 96-bit connection identifier (two 32-bit IP addresses and two 16-bit TCP port identifiers). Therefore, using the key to directly index a 2^{96} entry table in RAM or using a 96-bit wide CAM (contents addressable memory/associative memory) is impractical.

In [5], we describe effective hashing functions for hardware and software implementations using either CAM or RAM based connection tables. Our results indicate that even with simple hashing schemes, an average number of lookups very close to one can be achieved.

F. Datagram Throughput Requirements

The BGW processes every datagram in a way significantly different from normal boundary routers. Specifically, the BGW must lookup the TCP connection record, update the byte and packet counters, and store the results back in the connection table. Figure 7 shows the average datagram intensities as well as the datagram intensities for the busiest 100ms intervals. Our analysis showed that the BGW must be capable of processing datagrams at rates up to 10,000 packets per second. Depending on the architecture of the BGW, this seems quite feasible with current commercial routers.

IV. Discussion

About 90 percent of an estimated $200 million annual Internet cost is today recovered from users in the form of a variety of organizational charges and user prices. These charges and prices are designed for administrative convenience rather than to promote efficient use of network resources. Prices designed to achieve efficient allocation or fair cost recovery, however, require a billing system that can meter the individual user's traffic and that can present the user with information that encourages efficient use. This paper presents a billing scheme that can identify and authenticate users, monitor individual user traffic, and present the user with real time pricing information. Statistics collected from a two week trace of all TCP traffic between the University of California, Berkeley, and the "outside world" demonstrates that the billing scheme is practical, and introduces acceptable latency. Furthermore, in [5] we describe an implementation of the scheme based on the BayBridge [9],[10] a prototype high-performance router.

It is an open question whether the traffic demands and corresponding costs of operating the Internet will grow sufficiently to make pressing the need for an efficient pricing and cost recovery scheme. That will depend on the proliferation of bandwidth consuming applications on the Internet. In turn, that will depend upon the ability of Internet protocols, links and routers to support such applications in competition with other network service providers, including cable TV and telephone companies.

V. Bibliography

[1] Braden, R; "Requirements for Internet Hosts -- Communication Layers", *RFC 1122*, Internet Engineering Task Force, 1989.

[2] Braun, H-W; Claffy, K.C; Polyzos, G.C; "A Framework for Flow-Based Accounting on the Internet", *Proc. of the Singapore Intl Conference on Networks*, (1993).

[3] Claffy, K; Braun, H-W; Polyzos, G; "Tracking Long-Term Growth of the NSFNET" *Communications of the ACM*, (1994) Vol. 37, No. 8, 34-45.

[4] Comer, Douglas E.; "Internetworking with TCP/IP — Volume I", Prentice-Hall, 2nd Ed., 1991.

[5] Edell, R.J.; McKeown, N.; Varaiya, P.P.; "Billing Users and Pricing for TCP", *IEEE Journal on Selected Areas of Communication*, to appear.

[6] Estrin, D.; Zhang, L; "Design Considerations for Usage Accounting and Feedback in Internetworks", ACM Computer Communication Review, (1990) Vol. 20, No 5, pp56-66.

[7] Mackie-Mason, J, Varian, H; "Economic FAQs About the Internet", Journal of Economic Perspectives, (1994) Vol.8, No.3, 75-96.

[8] Mackie-Mason, J; Varian, H; "Pricing the Internet", in: *Public Access to the Internet* (Kahin, B. and Keller, J. eds., Prentice-Hall, 1994).

[9] McKeown, N; Edell, R; Le, M.T; "The BayBridge: a high speed bridge/router". (Protocols for High-Speed Networks III. IFIP WG6.1/WG6.4 Third International Workshop, Stockholm, Sweden, 13-15 May 1992). *IFIP Transactions C (Communication Systems)* (1993), vol.C-9, 203-18.

[10] McKeown, N; Edell, R; Le, M.T; "The BayBridge: A High Speed Bridge/Router between FDDI and SMDS Part I — System Architecture and Performance", Submitted.

[11] Merit Network, Inc., "NSFNET Backbone Service Traffic Distribution by Internet Network Number," ftp://nic.merit.edu/nsfnet/statistics/1994/nsf-9409.pnets. September 1994.

[12] Mills, C.; Hirsch, D; Ruth, G; "Internet Accounting: Background", *RFC 1272*, Internet Engineering Task Force, 1991.

[13] Steiner, J; Neuman, C; Schiller, J; "Kerberos: An Authentication Service for Open Network Systems", *Proc. of USENIX Winter Conference* (1988), 191-202.

[14] Taxpayer Assets Project - Information Policy Note May 7, 1994.

VI. FIGURES

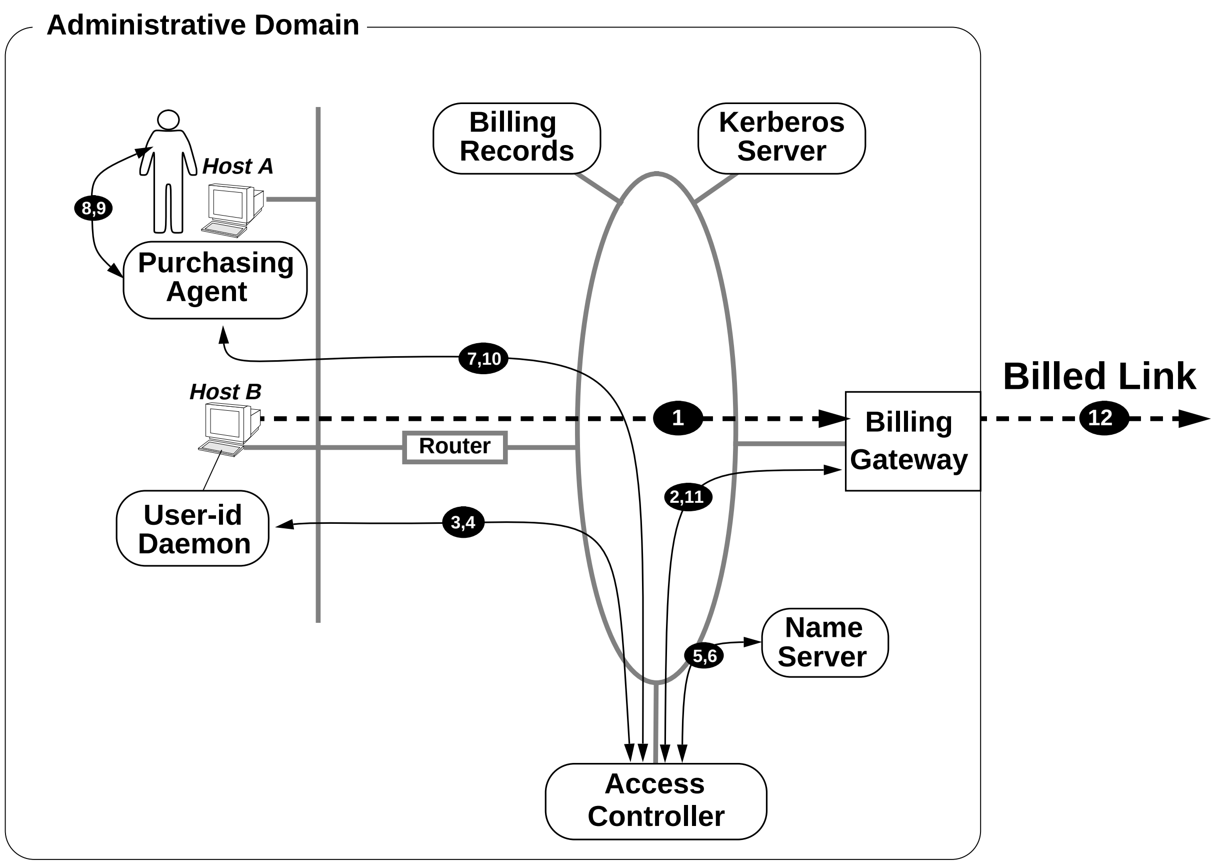
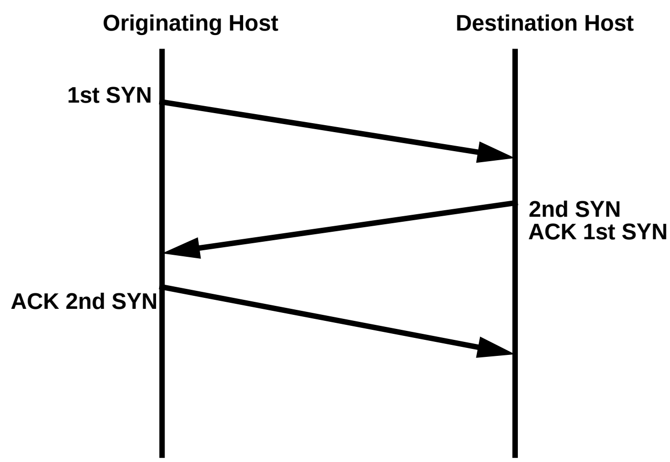
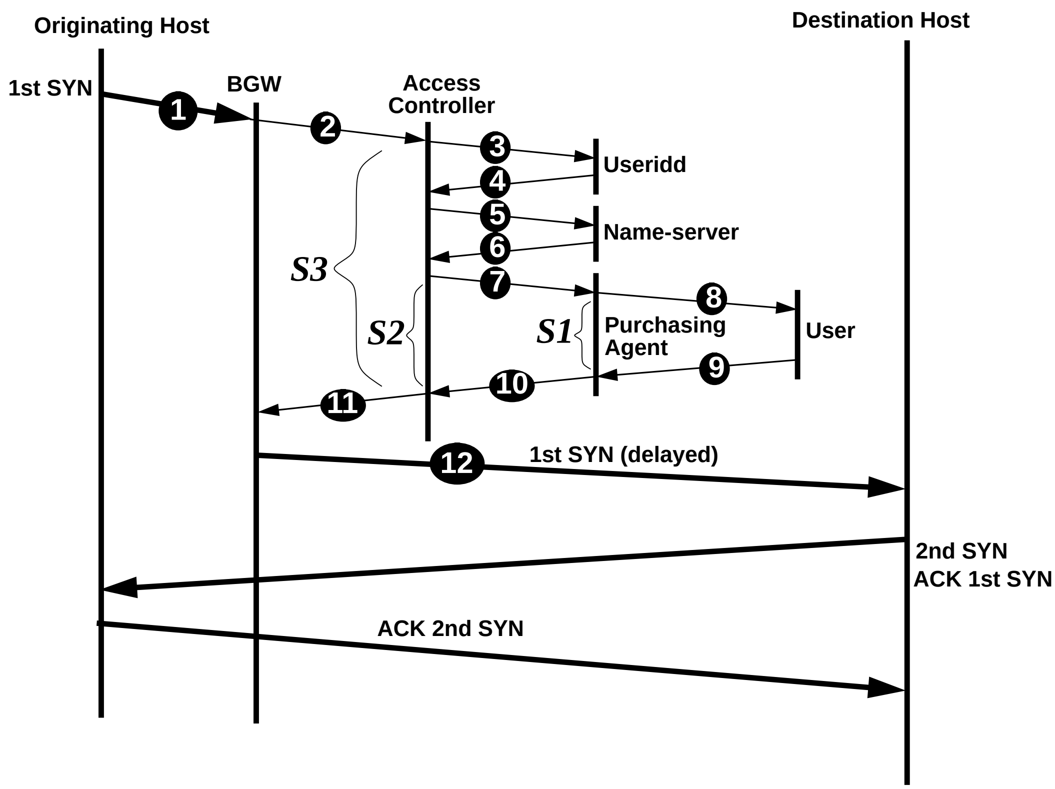


FIGURE 1. Components of the billing systems showing the major communications required to establish a connection.



(a) Normal 3-Way handshake to establish connections



(b) Billing System Communications Involved in Establishing a Connection

FIGURE 2. The communications required to establish a connection without and with the billing system.

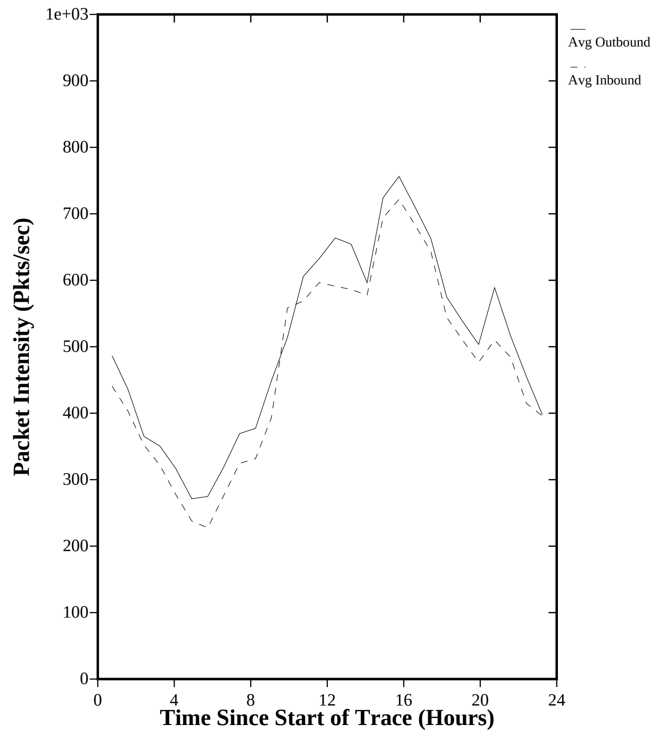


FIGURE 3. Average intensity of datagrams entering and leaving the Berkeley campus.

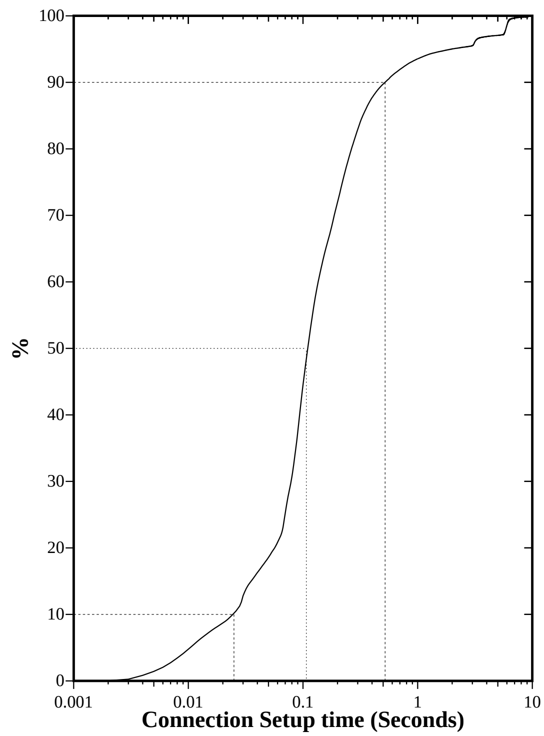


FIGURE 4. Cumulative Distribution of Setup Latencies for Connections that originate on the Berkeley campus and connect to a host elsewhere on the Internet.

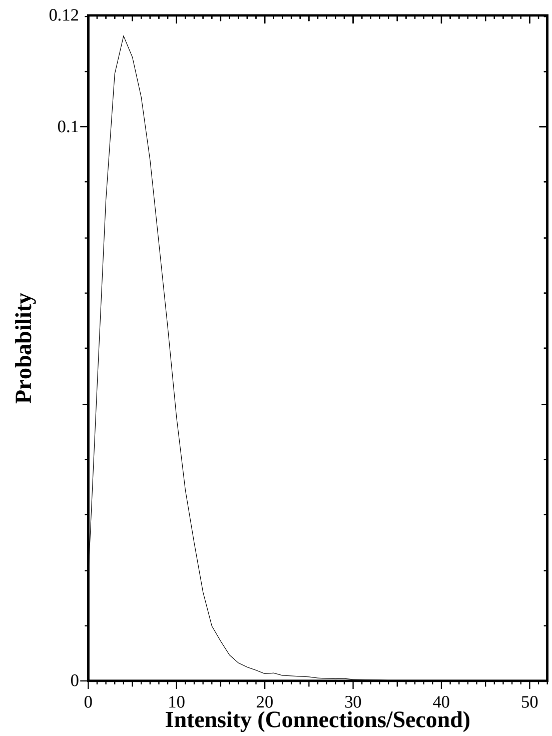


FIGURE 5. Rate at which connections are established to and from the Berkeley campus.

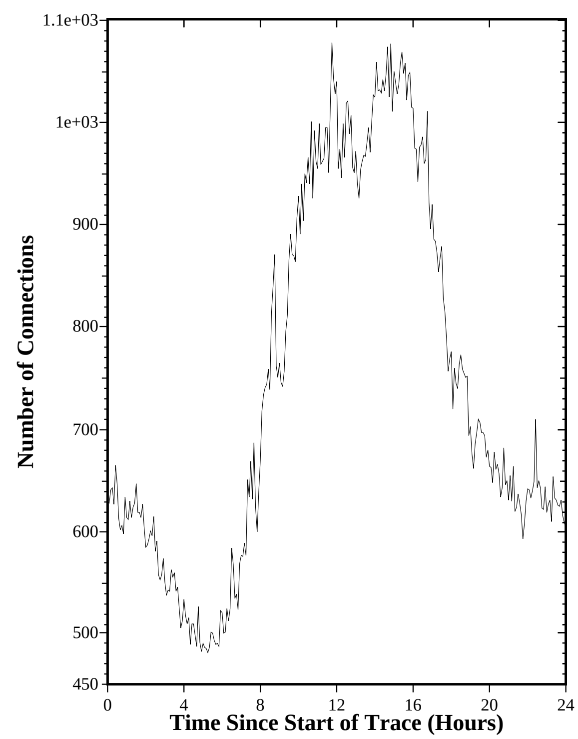


FIGURE 6. A count of the number of currently established connections. Connections are timed-out after an inactivity period of 30 minutes.

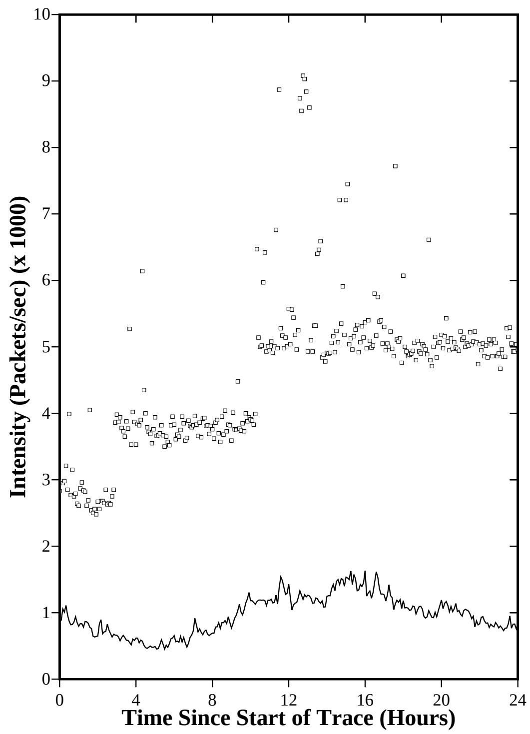


FIGURE 7. Rate at which datagrams leave the Berkeley campus.