Packet Sequencing: A Deterministic Protocol for QoS in IP Networks

Sean S. B. Moore and Curtis A. Siller, Jr., Cetacean Networks, Inc.

ABSTRACT

We describe a deterministic protocol for routing delay and loss-sensitive traffic through an IP network. Unlike traditional approaches, the method described here — packet sequencing — does not rely on queue management. Instead, it uses a temporally-based deterministic protocol to coordinate and switch IP packets on a system-wide basis. As a result, end-to-end throughput is guaranteed, without packet loss, loss variance, or accumulated performance impairment; additionally, end-to-end delay is minimized, and jitter is essentially eliminated. We also show that packet sequencing can complement conventional IP networks: sequencing does not negate the use of queue management QoS methods that are the subject of considerable ongoing study. This article describes the fundamental approach, issues associated with scalability, illustrative performance in the context of storage networking, and attributes related to the security and reliability of IP networks.

INTRODUCTION

IP networks are today tasked to handle traffic types for which the original Internet was not designed. The literature describes many of the associated newer Internet Protocol (IP) applications, which include at least the following: voice; videoconferencing, video collaboration, and video streaming; circuit and private line emulation; distributed "twitch" games; collaborative computing; and mission-critical data management. While readers could readily add to this list, it is worth noting that when looking across the breadth of these applications, new service attributes emerge that are not readily accommodated by conventional IP networks, such as significant delay constraints, susceptibility to packet jitter, vulnerability to packet loss, and the need to guarantee and manage throughput, especially for data. Even more difficult is the prospect of concurrently satisfying all these requirements, as would be necessary in a multiservice converged IP network built on a single infrastructure and handling a wide variety of different traffic types.

The challenge mentioned above is often described as the quest to attain quality of service (QoS) in IP networks. Research and development into this topic has spanned decades, is still a popular area for active research at universities and major corporate and government laboratories, but largely remains an elusive goal. Our market research and the view of others (e.g., [1]) make clear that the most popular QoS schemes currently being promoted have yet to achieve significant market penetration. These QoS methods can be categorized as resource reservation or prioritization. Traffic engineering is sometimes also mentioned, although it is not a standalone QoS technique, but rather an adjunct to those just mentioned. Moreover, overprovisioning is best viewed as a fallback position rather than a robust, rigorous QoS approach per se, and may not prove economically viable over the long term. All four of these (resource reservation, prioritization, overprovisioning, and traffic engineering) have been the subject of considerable study and are amply described in journal articles, books, and white papers. (Recent Feature Topics on IP QoS in *IEEE Communications Magazine* are especially apropos [2].) For brevity we note that some methods have been deemed unscalable (e.g., Resource Reservation Protocol-based integrated services, IntServ) while others rely upon some aspect of prioritization (differentiated service, DiffServ). Furthermore, there are a number of newer, typically proprietary approaches that depend on traffic shaping to mitigate traffic burstiness or "climb the protocol stack." As a consequence, those latter methods are usually application-specific, can become complex, are less scalable, and may be more expensive due to implications of violating the philosophical principles of IP network design [3].

It is well known that prioritization degrades under load. Multilevel prioritization often translates into higher-priority traffic dominating lower-priority traffic, and so forth [4]. Other research shows that it is complex, and in some

0163-6804/03/$17.00 © 2003 IEEE

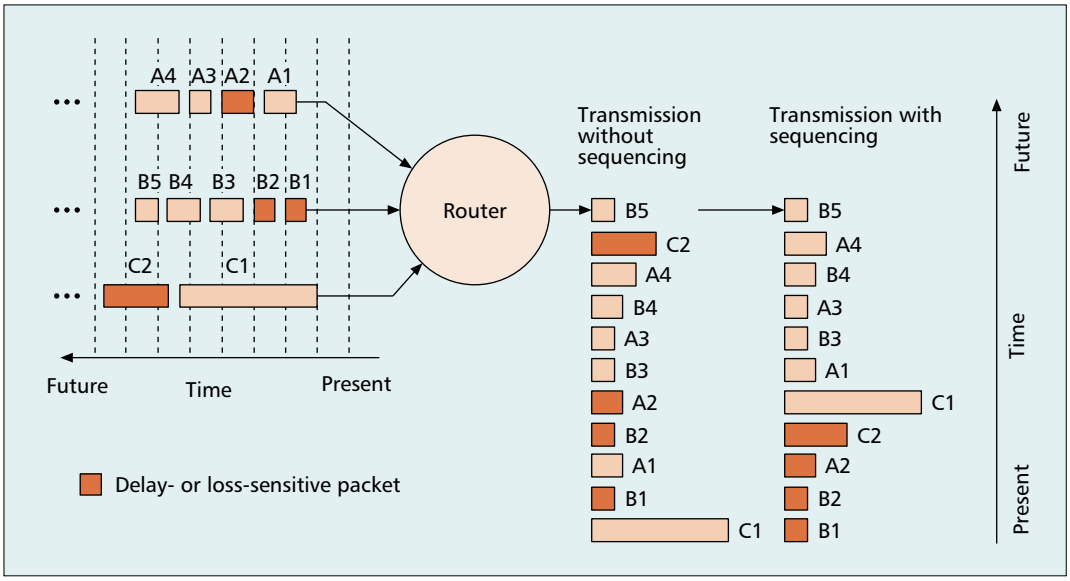

Figure 1. *A depiction of packet processing in a router, illustrating FIFO and sequence queue processing.*

cases impossible, to provide proportional QoS for multiple classes of service at an individual router, let alone on an end-to-end basis [5], or to simultaneously provide proportional QoS across multiple performance metrics (delay, jitter, loss, loss variance) [6]. Evidence also suggests that while prioritization-based QoS architectures and implementations such as DiffServ may scale, the associated QoS performance and management do not. These observations have been born out in an actual service network and suggest a questionable business proposition for both customers and service providers [7]. Additionally, our own market research affirms (as do others [1]) that not only is there little penetration of these QoS techniques, but also that overprovisioning remains the "technology" of choice. Finally, interviews we have held with nearly 50 large enterprise customers in finance, healthcare, and education markets reveal that many information technology (IT) organizations are unwilling to identify one class of applications or services as more important than another, inferring that while prioritization techniques might work in principle, there are organizational impediments to their actual implementation.

Multiprotocol label switching (MPLS) has been the subject of numerous papers, standards initiatives (e.g., in the Internet Engineering Task Force, IETF, and MPLS Forum), and articles in the trade press. A thorough introduction to the subject can be found in [8]. As previously suggested, MPLS-associated traffic engineering is not generally viewed as a standalone QoS technique, and concerns have been raised about its scalability and ability to respond to dynamic changes in network conditions. In practice, MPLS borrows from asynchronous transfer mode (ATM) by using PVC-like connection-oriented IP paths, augmented by dynamic or offline route optimization. Of the two approaches, offline traffic engineering typically produces better solutions but is slower and can be complex and expensive (e.g., see IETF Internet drafts related to this subject). Although judicious traffic engineering is prudent with or without MPLS, current approaches do not meet the requirements of a comprehensive QoS solution. Some deployments of MPLS have been deferred on the basis that it has not been technically or economically justified.

Finally, at the most fundamental level User Datagram Protocol (UDP) traffic (e.g., voice, video) and Transmission Control Protocol (TCP) traffic (data) are known to not mix well. Combining them, especially increasing the proportion of UDP, without benefit of the technology described here raises concerns about another collapse of the Internet [9].

This article describes a deterministic protocol for attaining ideal QoS. As noted before, not every application requires QoS, but those that do — and have less critical performance requirements — can concurrently exploit some of the other methods mentioned above. A fundamental understanding of the technology is provided in the following section. This is followed by further elaboration of system performance, especially in terms of:

- Scalability
- Illustrative performance of packet sequencing in the context of storage networking (an application of packet sequencing to distributed throughput denial of service, DoS, attacks, and scheduling and bandwidth management of TCP)
- Special attributes of the protocol as they relate to reliability and security of IP networks

Concluding remarks briefly summarize the unique advantages of packet sequencing.

A Primer on Packet Sequencing

Figure 1 facilitates a basic understanding of packet sequencing. A single router is depicted, with three inputs links (A, B, and C) funneling IP packets to a single output link. Without

At the most fundamental level UDP traffic and TCP traffic are known to not mix well. Combining them, especially increasing the proportion of UDP, without the benefit of the technology described here raises concerns for another collapse of the Internet.

A system of SSRs would usually be configured with a period and appointment size common for all links. Hence, the number of appointments available to sequenced flows depends on link speed.

actively managing the router queue the incoming packets are simply processed on a first-in first-out (FIFO) basis, resulting in a packet flow C1, B1, A1, B2, A2, B3, A3, B4, A4, C2, B5…. It is evident that larger packets (e.g., C1) preempt the expeditious routing of smaller packets (B1, A1, B2…), resulting in both packet delay and delay variation (jitter). If the packet queue depth is exceeded, packets are lost and problematic bursty packet loss can also result.

Figure 1 obviously describes an especially simple example of router behavior. In fact, there are numerous approaches to queue management (sometimes described as scheduling, and not to be confused with the approach described in this article) with new techniques being proposed almost daily — certainly a fertile area of queuing theory research.

In contrast we now interpret the behavior of the sequenced switch router (SSR — the reason for change in terminology will be evident momentarily) in Fig. 1 by a priori assuming deterministic, temporal sequencing of certain (i.e., delay-sensitive and/or loss-sensitive) flows, whether such sequencing is first established at a client endpoint or the first downstream SSR in a network. For illustration we focus on packets A2, B1, B2, and C2 (shown shaded in the figure) as having been temporally sequenced — note that each of these correspond to distinct packet flows. The packets can be of different size, and there can be multiple independent sequenced flows on any link. Because the packet service times have been precisely predetermined (temporally pre-aligned — later discussion explains how this is accomplished) and given unequivocal precedence over unsequenced packets, we can stipulate that they do not overlap in the time domain with other sequenced packets and are immediately serviced based on scheduled arrival. Under these assumptions, the SSR output becomes B1, B2, A2, C2, C1, A1, B3, A3, B4, A4, and B5…, where we again emphasize that with prior knowledge of a sequenced packet's anticipated arrival, an SSR will process and switch that packet at highly precise times (called *appointments* — see discussion below) regardless of the presence of unsequenced packet traffic.

The Vocabulary of Packet Sequencing

We allude above to several concepts that require further discussion. The first concept is that of a measurement unit for packet size, different from the customary byte or maximum segment size (MSS). For packet sequencing, an *appointment* is the discrete size unit and is defined as some number of bytes. For example, a 238-byte G.711 Ethernet voice packet (composed of a 160-byte G.711-encoded payload, 12-byte Real-Time Protocol, RTP, header, an 8-byte UDP header, a 20-byte IP header, and a 38-byte Ethernet frame) would conveniently fit into a 250-byte appointment; on the other hand, given a 50-byte appointment the same voice over IP (VoIP) packet measures five appointments. Other applications (e.g., videoconferencing) have variable packet sizes, each of which could measure a different number of appointments. Selection of appointment size is a design decision based on a number of factors. For example, small appointments make for finer granularity (and larger appointments can always be built up from contiguous smaller ones) but increase the size of data structures, which potentially effects computational efficiency.

A second point is that of defining time modulo a *schedule period*, that is, the system concept of time is that it repeats with fixed periodicity. As with selection of appointment size, selection of schedule period is a design decision. For example, selecting a 20 ms period simplifies the engineering of a sequenced network supporting a VoIP application with 20 ms G.711 packets, because each individual VoIP flow produces one packet per schedule period. Clearly, however, there is no technical restriction that limits a schedule period to only one sequenced packet for an individual application flow. In fact, some sequenced applications could require more. However, it is intrinsic to the system architecture that every sequenced application map to a fixed number of appointments at a specific periodicity. For asynchronous applications, it might seem that this approach leads to inefficient link utilization. As explained later in this article, backfill obviates this effect.

A system of SSRs would usually be configured with a period and appointment size common for all links. Hence, the number of appointments available to sequenced flows depends on link speed. For example, for a system configuration of 250-byte appointments and 20 ms schedule periods, a 1 GigE link has 10,000 appointments per period (1 Gb/s * 20 ms * [1 appointment/250 bytes] * [byte/8 bits]). A sequencing system views a schedule period as being composed of an integer number of appointments that are delineated by precise times within the schedule period. The collection of appointments and their assignment to various flows composes a *schedule*. Each router output port and link has its own schedule.

The final notion is that of flow *itinerary*. Briefly, the link-by-link assignment of appointments to an individual flow makes up a flow's end-to-end itinerary. The appointments in a flow's itinerary specify precisely when each SSR in the flow's path will service the flow's packets. The assignment must take into consideration the point in time at which an endpoint commences sequenced packet transmission, the propagation delay between network nodes (i.e., SSRs), and delays within an SSR's internal switching fabric. A sequence agent (SA) coordinates the itinerary creation task for all flows within a switching domain by using a special-purpose signaling protocol, in conjunction with a special appointment on every link, referred to as a *heartbeat*, where the latter is used to maintain system-wide schedule frequency and phase accuracy.

Sequenced Network Architecture

A sequenced network is generally made up of multiple SSRs, one or more SAs, and numerous sequenced endpoints (SEPs). (As mentioned before, endpoints need not be sequenced; per-

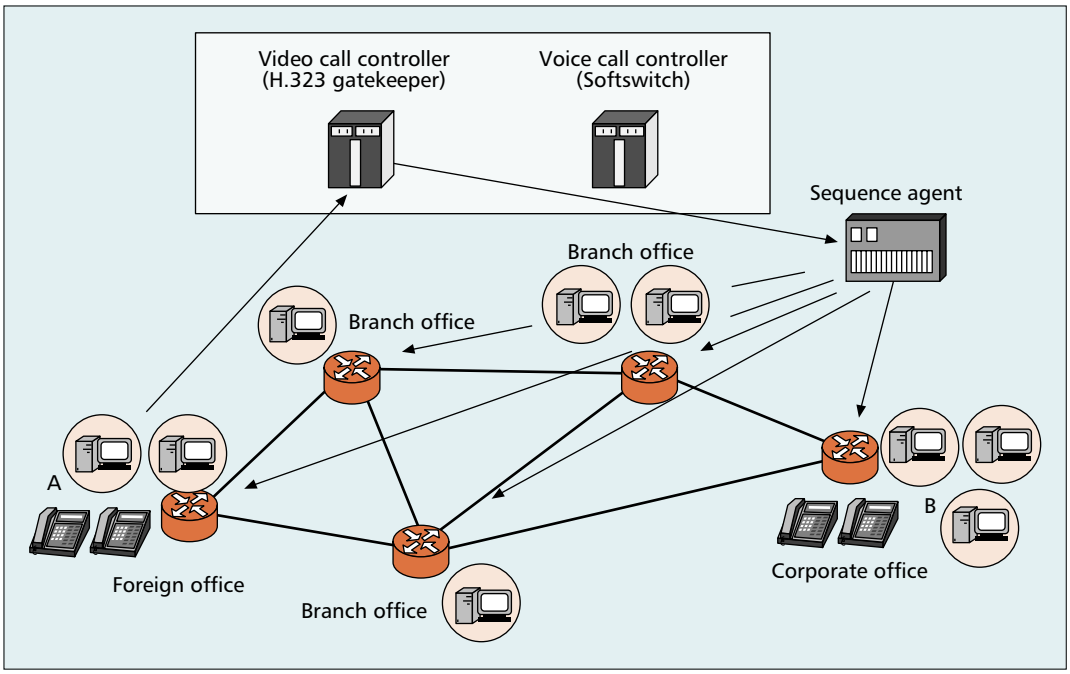

Figure 2. *A representative enterprise network.*

When an itinerary is torn down, the appointments are freed and become available for satisfying new itinerary requests. Note that the itinerary search and establishment process was designed to avoid the causes of scalability problems in RSVP.

flow temporal alignment can be provided at the first downstream SSR in a network.) When a session begins, the sequencing process is initiated by the SEP. The SEP requests an itinerary lease from the SA, either directly or through an application server (e.g., a voice softswitch, a video gatekeeper, or a surveillance/law enforcement server). The request includes information such as source and destination IP address, maximum packet size, packet rate, and service parameters. The SA validates the request in accordance with network policies and calculates an itinerary through the network. This calculation is based on network topology, existing itineraries, and network policies. The SA then distributes relevant itinerary information to the SEPs and the SSRs along the itinerary path using a reserve-and-commit process. Once the itinerary is set up, the sequenced packet flow is maintained entirely by the SEPs and SSR; the SA's involvement is limited to lease renewal and teardown. When an itinerary is torn down, the appointments are freed and become available for satisfying new itinerary requests. Note that the itinerary search and establishment process was designed to avoid the causes of scalability problems in Resource Reservation Protocol (RSVP), which also reserves network resources, specifically by:

- Using hard state instead of soft state for the connection
- Localizing a more efficient itinerary search process in a network-attached host instead of distributing the path discovery process across the network

A process akin to that described above is described below with the aid of Fig. 2. This simplified network architecture illustrates a single SA in conjunction with two call controllers (a video gatekeeper and a VoIP softswitch) and five customer locations in an enterprise network that supports both VoIP and videoconferencing applications. A typical sequence for call setup and takedown, applicable to both the video and voice applications (video is depicted), proceeds as outlined below and is not dependent on any particular signaling protocol, whether H.323 (noted in the figure), Session Initiation Protocol (SIP), or another.

1) Terminal A (video or voice) at the foreign office indicates to the appropriate call controller its desire to establish a communication session with terminal B at the corporate office.

2) The call controller consults local policies (its own or external policy server) and grants (or rejects) the call request.

3) Terminal A signals session setup to terminal B. This signal is preferably routed through the call controller to achieve a greater degree of control over available network resources.

4) Terminal B signals, via the call controller, acceptance (or rejection) of the session.

5) The two terminals negotiate parameters of the media flows (audio, video) for the session. Depending on the signaling session chosen, the negotiation happens during steps 3 and 4, or in separate steps after the session establishment phase. In any case, the media flow parameters become known to the call controller.

6) The call controller makes a service request to the SA via a service request interface (not shown). Depending on the application there can be multiple requests, one for each data (voice, video, file transfer, etc.) flow.

7) If the request(s) succeeds the session progresses further. If it does not, the call controller (based on service/application logic) can decide if the session should be terminated.

8) When the session is finally terminated by either a terminal (A or B) or the call controller, the latter asks the SA to release resources that were used for the session.

The role of the SA in this scenario, steps 6

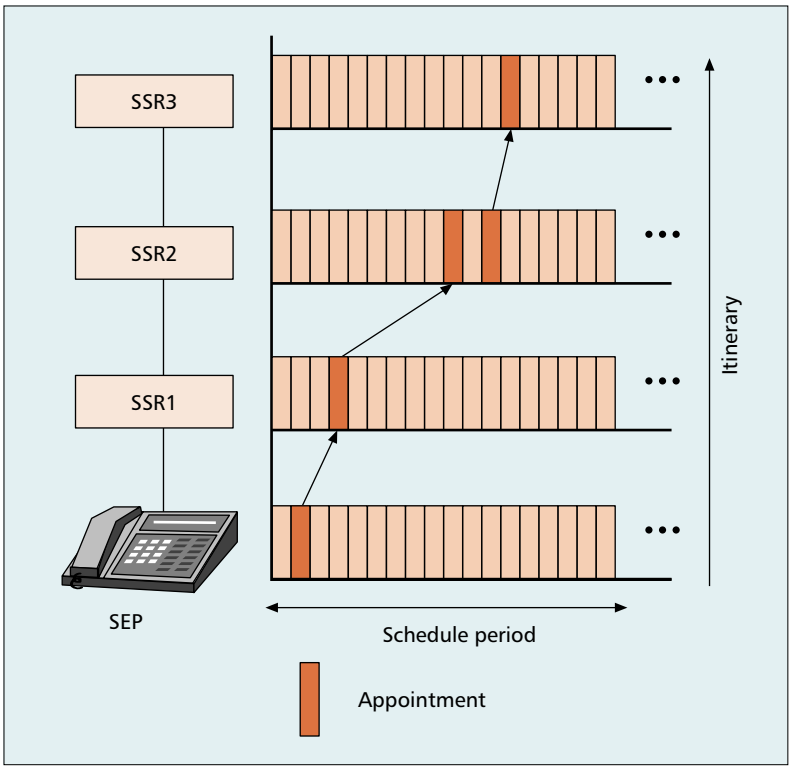

Figure 3. *SSR-by-SSR packet appointments and itinerary in a network segment.*

and 8, should be especially noted. Furthermore, this figure illustrates the role of an SA in a single-domain network. As a network grows in size, it will eventually exceed the processing capability of a single SA. Consequently, we have developed a *multidomain* signaling architecture based on a peering relationship among SAs, with intradomain signaling and itinerary establishment occurring concurrently with SA-to-SA communication among domains.

Visualizing Packet Flow through a Sequenced Network

In a sequenced network, each device port has a repeating schedule. Typically all port schedules in a network are configured with the same schedule period and appointment size. Packet flow through a network is illustrated in Fig. 3, where the schedule at each node along three segments of a network path is shown. In Fig. 3, the packet size shown measures one appointment, but in general packets may measure more than one appointment, in which case a block of contiguous appointments is allocated to service the packets.

A packet is transmitted by the SEP (shown in the figure as a telephone) at the beginning of an appointment block assigned by the SA. Using prior flow knowledge, SSR1 expects the packet to arrive at the beginning of the first appointment in its corresponding appointment block. (The packet appointment is shown darkly shaded in Fig. 3.) Consulting its sequence schedule, SSR1 knows to then forward the packet during the corresponding appointment block on its output port to SSR2. Using its flow knowledge, SSR1 will also hold off (queue) any unsequenced traffic on that port that would otherwise interfere with transmission of the scheduled packet. As soon as the sequenced packet transmission is complete, the port is available for unsequenced traffic.

SSR2 expects the packet at the beginning of its corresponding receive appointment block. However, in this case the packet is not switched out immediately. Instead, the output port specifies a small delay — a phase shift of some small number of appointments — in order to line up with SSR3's receive appointment block. This per-node phase shift is an integral part of the packet's itinerary and is permitted during itinerary search in order to reduce blocking probability. Analysis shows that a phase shift of as few as 10 250-byte appointments profoundly reduces the likelihood of itinerary blocking. (On an OC-3 link, 10 250-byte appointments corresponds to 0.13 ms. Traversing 10 nodes with a max-case 10-appointment alignment buffer at each introduces only 1.3 ms of additional end-to-end delay, a small fraction of what can accrue from transport. Fiber optic transport delay corresponds to approximately 8 ms/1000 mi, so 1.3 ms is equivalent to only 160 mi. At higher link speeds, the delay introduced by phase shift is even less.)

The timing precision of the system is bounded by very small variances introduced by the hardware switching fabric and small drifts in link transport time. To account for these variances, each appointment block's leading edge begins with a small "guard band" during which an SSR looks for the arrival of an expected sequenced packet on an ingress port. (The guard band is not shown in the figure.) Arrival of a packet during the guard band interval identifies it as the expected sequenced packet. If no packet arrives during the guard band interval, the SSR concludes that no sequenced packet is being switched during that appointment block, so the appointment block can be used to forward unsequenced traffic. This ability to route traditional unsequenced traffic using pre-allocated but unused appointments is known as *backfill* and ensures that transmission capacity is not "wasted." Backfill has been verified in laboratory tests and is part of the vocabulary of packet sequencing.

One final attribute of this technology should be mentioned: SSRs are *simultaneous dual-mode switch routers*. At the same time that they provide ideal QoS to critical flows using sequence technology, they can concurrently function as conventional IP routers (using standard routing protocols and traditional QoS and traffic engineering techniques — RSVP, IntServ, DiffServ, MPLS, etc.) for unsequenced flows. This is possible because sequencing provides complete isolation between individual sequenced flows and, of course, between sequenced and unsequenced flows. The latter are conventionally routed. Additionally, packet sequencing makes no changes to the IP protocol; thus, sequenced flows as well as unsequenced flows that are switched into conventional routers do not require any packet transformations.

System Attributes and Performance

This section describes performance attributes associated with itinerary availability and endpoint reachability, both aspects of packet sequencing scalability. This is followed by remarks on the per-flow robustness of packet sequencing, using storage networking as an illustrative application. Furthermore, we provide a detailed overview of the approach as it relates to providing some unprecedented security capabilities in IP networks.

Itinerary Availability and Endpoint Reachability

Attaining the exceptional level of QoS possible using packet sequencing relies on:

- Identifying flow itineraries
- Signaling that information to nodes and endpoints in the network

Both aspects of the problem have been studied in detail, with salient observations provided below. A detailed mathematical analysis is inappropriate for this publication and will be the topic of subsequent papers, but some insights are provided below.

Looking first at itinerary availability, we capitalized on the notion of phase shift at switching nodes (as illustrated in Fig. 3) to reduce blocking probabilities in our itinerary search algorithms. With this insight the compute time necessary to identify itineraries was determined, subject to a very high confidence level. Under some reasonable assumptions, for the case when all sequenced flows have the same packet size, blocking probabilities can be computed using Erlang's B formula [4], an observation that reflects the similarity of this case with finding connections in telephone networks. For the case when sequenced flows have different maximum packet sizes, computation of the itinerary search blocking probabilities is a complex process that will be detailed in forthcoming papers. In either case, we found that for very long routes (up to 40 nodes) and at high levels of sequenced flow link utilization (approaching 90 percent), itineraries can readily be computed in at most a few milliseconds (using a standard workstation configuration). A casual study of path lengths and number of node traversals in IP networks suggests that actual end-to-end paths are more likely to be 12–16 nodes, for which the itinerary compute time is more typically microseconds. This performance has been borne out in laboratory implementations.

Having confirmed that itineraries are readily available in heavily utilized networks, the next step is to analyze the expected time to signal the establishment of the necessary connections from endpoint to endpoint across networks with global reach. This analysis was undertaken by first modeling a global-scale IP telephony network and then identifying signaling issues related thereto. Regarding the modeling aspect, we took as a paradigm the concept of local access and transport areas (LATAs) which came into popular use as the Bell System was being dismantled. LATAs vary significantly in size and typically manage 1–5 million subscribers. Assuming an individual SSR accesses 250,000 endpoints, a LATA translates to 5–20 SSRs, which we know from laboratory evaluation can be administered by a single SA. An (acyclic) network of 5–20 SSRs with moderate interconnectivity will likely have a diameter (maximum path length) of 4–5 hops. Consequently, by modeling a hierarchical inter-LATA network as a quaternary tree of depth six (which has 1365 LATA nodes) and assuming each LATA has 2.5 million endpoints and a diameter of 4 hops, the inter-LATA network has 3.5 billion endpoints and a diameter of 11 LATAs, giving 44-hop maximum-length paths.

As a next step and using well-known methods, we developed a multidomain signaling architecture that allows for concurrent intra-LATA and inter-LATA connection establishment, with reasonable time estimates allocated for signaling transport and processing. The analysis concludes that an itinerary can be established between any pair of globally distant endpoints in less than 2 s, well within internationally accepted guidelines.

Storage Networking

As a further performance assessment, this technology was tested in the context of storage networking by conducting a set of file transfers using a network-attached storage (NAS) configuration. This involved sequencing of TCP flows, which not only ensures minimal communication delay since there is no packet loss (and therefore no system delay associated with retransmissions), but also permits throughput guarantees, throughput provisioning, and goodput maximization. The network configuration was that of two clients performing sequential 64-kbyte block reads over TCP/IP, interspersed over a 100 Mb/s Ethernet link, first with a network of conventional routers and then with a network of (sequenced) SSRs. Background noise traffic, which can also be interpreted as throughput DoS attack traffic, was injected so as to compete with file transfer traffic for link resources. The conventional routers were tested both with and without the native QoS features enabled. When the QoS features were enabled, the noise traffic was associated with a lower priority than the file transfer traffic. Prioritization is one proposed method for creating some immunity to DoS attacks (but depends on the assumption that the attacking traffic has lower priority, which may not be the case). Figure 4 shows the comparative throughput performance as a function of port contention.

For the conventional routed network without QoS, the performance behaves as expected: the file transfer traffic goodput decreases linearly with noise/attack traffic. When QoS is enabled the results are especially interesting: while goodput is stable up to about 40 percent noise/attack traffic load, the goodput rate is significantly reduced from the non-QoS case, possibly due to QoS processing overhead. Above 40 percent, any DoS protection afforded by this QoS technique breaks down. That is, the underlying TCP goodput is unaffected by noise/attack

Regarding the modeling aspect, we took as a paradigm the concept of local access and transport areas which came into popular use as the Bell System was being dismantled. LATAs vary significantly in size and typically manage 1–5 million subscribers.

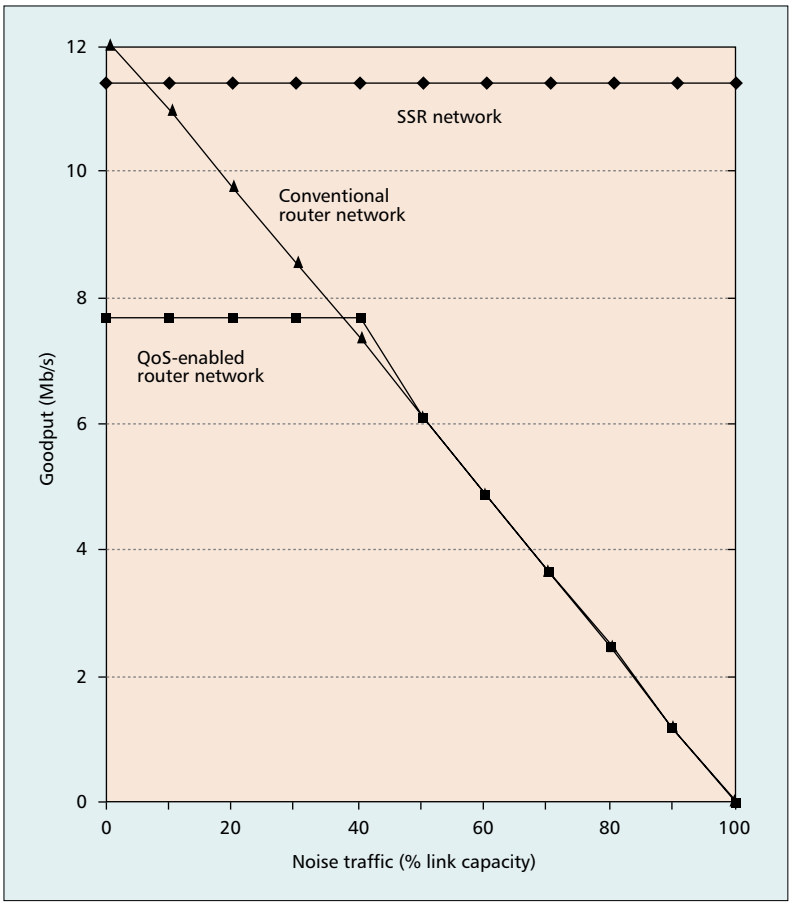

Figure 4. *Network-attached storage: comparison of file transfer performance between a pair of sequenced and conventional routers as noise/attack load is varied.*

traffic in the sequenced network, but is directly and adversely affected in the conventional router network.

The results for the sequenced network are particularly interesting. Not only does the sequenced packet technology permit managed TCP/IP, but the *SSR throughput was completely impervious to traffic on the network*, remaining rock solid to 90 percent port contention, at which point the conventionally routed throughput degraded completely. This performance is the very basis for system robustness and complete immunity to throughput DoS attacks that congest network links. The explanation is particularly simple: packet sequencing technology guarantees per-flow ideal QoS by using a time-based switching architecture. Since sequenced flows are identified and switched based on temporal information, other traffic is completely ignored during committed appointment times, regardless of type, prioritization, IP addresses, and so on. *Once an SSR admits a sequenced flow into the network, it cannot be corrupted and will only be interrupted when the SA tears down the session connection.*

In addition to the data presented above, measurements were also made of average response time for storage backup across a 100 Mb/s Ethernet link, assuming a network configuration identical to that described above and with packet sequencing scheduled for 6450 packets/s. The average time for sequential reads was stable at approximately 55 ms for port contentions ranging up to almost 100 percent. For the QoS-enabled router, the response time was constant at 75 ms up to 40 percent port contention, and then deteriorated rapidly to nearly 450 ms (a factor of 6 degradation) by 90 percent.

SECURITY

In addition to the immunity from throughput DoS attack noted earlier, packet sequencing readily supports numerous other security functions associated with trusted networks. Below we detail an efficient network availability and fault tolerance capability as an example of how the sequencing paradigm can simplify implementation of basic security functions. The impact of sequencing on other functions (legal intercept, firewall, confidentiality, multilevel security, and multilevel priority and preemption) is also described.

Assurance of Availability/Fault Tolerance — A network is highly available when there is high probability that a path with sufficient available capacity exists for a given flow. Availability depends directly on the reliability of routers and links along a path. The standard benchmark for router reliability is "five 9s", or 99.999 percent; router hardware and software is expected to average approximately 5 min of downtime/year.

Even if a network were composed of routers and links with 99.999 percent reliability, *network* availability would not be as high. One reason is theoretical: the availability of a path through a network is computed as the product of the reliabilities of the nodes along the path. The other reason is pragmatic: operator errors and unanticipated catastrophic events account for a majority of node failures, which means that even if router hardware and software were 100 percent reliable, there is still a nonzero probability of network unavailability.

Although service providers typically do not publish operator error statistics, a large-scale independent experiment suggests that node reliability is about 99.7 percent [10]. The expected reliability of a 10-node path is therefore $0.997^{10} = 0.97$, or 97 percent. The connectionless routing strategy of IP networks was designed specifically to handle such path unreliability by rerouting traffic around failed nodes or facilities. While rerouting delays may be acceptable for some non-real-time traffic, it can be unacceptable for time-inelastic mission-critical traffic, examples of which were mentioned earlier in this article. For example, in telephony networks synchronous optical network (SONET) technology is designed to provide a 50 ms restoration time upper bound; such fast restoration has yet to be demonstrated in conventional IP networks, where routing convergence times are currently measured — at best — in tenths, not hundredths of a second.

Packet sequencing networks can achieve high network availability through a straightforward itinerary redundancy approach based on an efficient branch-and-merge functionality. Each row in a sequencer's forwarding table can have multi-

ple egress {port, appointment} pairs, which function as a signal to replicate (branch) the associated sequenced packet. Classifying a sequenced packet as replicatable incurs very little processing overhead, in contrast to, for example, a multicast-based replication approach, which typically uses separate classification and routing logic. *At the merge point, the SSR knows precisely when the replicated packets will arrive, which not only eliminates the need to filter every ingress packet but also eliminates uncertainty in waiting (delay) times.*

Such an approach is shown in Fig. 5, which depicts a five-node network similar to that illustrated in Fig. 2. In this instance, traffic from the foreign office is switched through branch offices 1 and 2 to the corporate office, with an itinerary determined by the SA. The SA similarly affords an alternative, branched itinerary path via branch office 3 to the same corporate office, where the two separate flows are merged.

As a heuristic rule, we have found that 99.999 percent network availability is likely achieved for any sequenced flow by using only three redundant nonintersecting itineraries. At first glance one might conclude that the network would then be overprovisioned by a factor of three. However, this is not necessarily the case under a reasonable assumption that not every sequenced flow requires high availability. Multiple redundant itineraries in a sequenced network can be efficiently allocated on demand to a single flow. When the flow terminates, all of the resources are immediately returned to the resource pool.

Furthermore, an alternative approach to ensuring high flow reliability is to reserve alternate paths through the network for any particular flow. Only in the event of a failure is the packet flow switched to the backup itinerary. During quiescent conditions of network operation, the reserved throughput is not wasted, although it is allocated for a scheduled flow and available in the event a sequenced packet arrives at its appointment. Without the arrival of a sequenced packet, backfill permits the routing of unsequenced flows.

As an aside, we remarked earlier on the functional use of a single SA in each sequenced network domain. For improved reliability, within a domain two or more SAs can be internetworked to provide database concurrency and redundancy, using well-known cluster-controller computer network architectures with failover fault tolerance. We have also added standard authentication and encryption protocols to the signaling architecture to increase protection from attacks, and we have designed the signaling architecture as a virtual out-of-band system to further protect it from DoS attacks.

Legal Intercept/CALEA — Many countries have laws and requirements for the legal interception of telecommunications traffic. In the United States, these are stipulated in the Communications Assistance for Law Enforcement Act (CALEA), signed into law 25 October1994. CALEA requires carriers to ensure law enforcement's ability (pursuant to court order or other lawful authorization) to intercept communications regardless of advances in technology. Compliance deadlines have been extended for packet mode communications because the intercept function is especially difficult to implement using conventional IP technology.

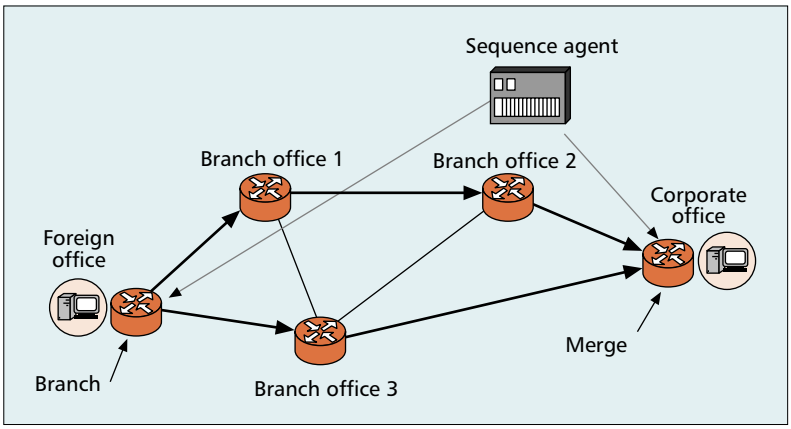

Figure 5. *An illustration of branch and merge for network reliability.*

The essence of the requirements for telecommunications carriers can be summarized in four key elements:

- Isolate call content transmitted by a carrier in its service area.
- Isolate the call-identifying information.
- Transmit intercepted content and identifying information to a law enforcement agency.
- Unobtrusively carry out intercepts.

As mentioned above, packet mode communications are particularly difficult for carriers to intercept. First, packets might not enter the network via the customary twisted-pair wires used in traditional circuit mode communications. Second, packet flows are probably connectionless, where each router decides what the next hop should be for each packet. Therefore, conventional routed networks would have to be designed so that every edge node (entry point) is capable of intercepting communications. Even if routers could intercept packets on demand — which they cannot — carriers would also need to manage control interfaces to each one.

In a sequence-enabled network, sequenced flows traverse a known path across the network. Consequently, this technology allows legal intercepts to be initiated at the most convenient node along the flow path. The SSR at that node can easily be instructed by the SA to unobtrusively replicate (using branching) the contents of the packet mode communications to or from the intercept target. The intercept can then be readily and unobtrusively forwarded directly to the proper law enforcement agency, or to an intermediate CALEA delivery server for further processing. The intercept can be constructed to either automatically occur as part of call setup, or intercept communications in progress.

SSRs do not need to examine each packet, searching for communications to or from an intercept target. Rather, a packet-sequenced network knows the complete itinerary (appointments and links) of every scheduled packet traversing the network. Therefore, it is a comparatively simple process to identify and intercept only specified packet flows, a capability we

Policy enforcement is relatively simple in connection-oriented systems such as TDM/SS7 telephony networks because individual low priority connections can be readily identified and torn down and because QoS is guaranteed regardless of the traffic state of the network.

have affirmed in laboratory testing. More important, this simplicity and precision allows to packet sequencing to scale to meet demanding governmental requirements.

The quality of the intercept is an additional important issue. If the intercepted flow were to suffer any packet loss, it might prove difficult to offer unequivocal evidentiary proof of claims. Sequencing the legal intercept provides the same QoS benefits described elsewhere in this article, most notably no packet loss regardless of network utilization or congestion.

Efficient Firewall — Assuming that itinerary requests are authorized and authenticated, an SSR intrinsically functions as a highly efficient firewall with no adverse impact on delay-sensitive traffic. Sequenced packets are completely identified by arrival time, which is difficult to spoof. Nonsequenced traffic can be shunted at an SSR to a conventional firewall for traditional filtering.

Traffic Flow Confidentiality — Because the packet forwarding logic is based on packet arrival time, there is no need to examine IP headers. Hence, a high degree of confidentiality and nondisclosure of information transfer may be achieved by encrypting the IP header as well as the payload, thereby anonymizing routing information. Furthermore, note that encryption and decryption need only occur once (at network ingress and egress, respectively), which eliminates the need for encryption devices on internal links. Other techniques may be deployed to mask the transmission process altogether (i.e., to prevent an eavesdropper from inferring whether or not two organizations are communicating at all) by producing padded flows in which all packets are the same size, the flow rate is constant, and the payloads are encrypted.

Multilevel Security — Difficulty in supporting multilevel secure transmission in a conventional IP network is due to the difficulty of determining a priori when a shared resource will be used by a particular flow, as well as which shared resource will be used, which then makes it difficult to prevent flows/packets with different security levels from occupying shared resources at the same time. Techniques based on encryption may incur overhead that makes them ineffective for real-time flows. In contrast, for a sequenced flow the usage time is known precisely and is scheduled in advance. That is, for a precise and deterministic amount of time, SSR and link resources are dedicated to servicing a particular packet from a particular flow. These two attributes provide an effective and efficient basis for isolating individual flows with different security levels, thereby enabling a multilevel secure transmission capability for both conventional data and high-performance real-time flows.

Multilevel Priority and Preemption (MLPP) — MLPP uses policy management and enforcement modules to allow new high-priority flows to preempt lower-priority flows when there is no remaining capacity for the high-priority flows. Policy enforcement is relatively simple in connection-oriented systems such as time-division multiplexed/Signaling System 7 (TDM/SS7) telephony networks because individual low-priority connections can readily be identified and torn down, and QoS is guaranteed regardless of the traffic state of the network. In connectionless packet-switched networks with aggregated flows and statistical QoS, policy enforcement is a challenge not only because of admission control and policing issues but also because of the difficulty of determining when a new high-priority call will not receive sufficient QoS, and determining which flows to shut down in order to ensure statistically good QoS for the new high-priority calls. These latter issues speak to the challenge of providing a reliable "network busy" signal in connectionless packet-switched networks. In contrast, sequenced flows are connection-oriented and receive guaranteed QoS regardless of the traffic state of the network, and sequenced networks readily support a reliable network busy signal capability, thus making them an effective basis for an MLPP system.

Conclusion

This article provides an overview of packet sequencing, a technology that makes use of protocol determinism to provide exceptional QoS (no packet loss, minimal delay, and no jitter), reliability, and security in IP networks. As such, the technology makes possible the simultaneous delivery of (converged) services for which the original Internet was not designed, for example, voice, video conferencing and collaboration, collaborative computing, as well as legal interception of telecommunications traffic and private line (TDM) emulation. The technology can concurrently operate with alternative approaches to quality assurance, such as DiffServ and MPLS, which are being implemented today in traditional routers.

After a brief mention of QoS techniques, the fundamentals of packet sequencing are introduced. This includes an illustration of sequenced packet processing in an SSR, as well as discussion of a "vocabulary" especially pertinent to the approach. A sequenced network architecture is then described, after which packet flow from an endpoint through multiple SSRs is illustrated. This background material leads to a discussion of several significant system attributes, notably scalability, the application of sequenced TCP/IP in storage networks, reliability, and security (e.g., immunity to throughput DoS attacks, an efficient firewall capability, traffic flow confidentiality, multilevel security, and multilevel priority and preemption). Reliability and certain aspects of security are shown to make use of flow "branch and merge."

Packet sequencing is an especially attractive basis for attaining utmost quality, efficiency, and security in multiservice converged IP networks.

Acknowledgment

The authors are pleased to acknowledge Ilya Freytsis, Howard Reith, Steven Rogers, Paul Sprague, James Towey, and Dale Wisler for significant contributions to packet sequencing core technology and its applications.

References

[1] S. Giordano *et al.*, "Advanced QoS Provisioning in IP Networks: The European Premium IP Projects," *IEEE Commun. Mag.*, vol. 41 no. 1, Jan. 2003, pp. 30–36.

[2] P. Lorenz *et al.*, (guest editors), "IP-Oriented Quality of Service," *IEEE Commun. Mag.*, vol. 40, no. 12, Dec. 2002; and S. Giordano *et al.*., Guest Eds.), "Advances in QoS," *IEEE Commun. Mag.*, vol. 41, no. 1, Jan. 2003.

[3] R. Bush *et al.*, "Some Internet Architectural Guidelines and Philosophy," IETF RFC 3439, Dec. 2002, http://www.ietf.org/rfc/rfc3439.txt.

[4] D. Gross and C. Harris, *Fundamentals of Queueing Theory*, New York: Wiley, 2nd ed., 1985.

[5] M. Leung *et al.*, "Adaptive Proportional Delay Differentiated Services: Characterization and Performance Evaluation," *IEEE Trans. Networking*, vol. 9, no. 6, Dec. 2001.

[6] S. Chen *et al.*, "On the Ordering Properties of GPS Routers for Multi-Class QoS Provision," *SPIE Int'l. Conf. Perf. and Control of Network Sys.*, 1998, pp. 252–65.

[7] P. Sevcik, "QoS: Show Me the Money," IEEE ICC 2002 Business Applications Series, "Internet QoS: Technology, Status and Deployment," NY, Apr. 30, 2002; http://www.icc2002.org.

[8] W. Stallings, "MPLS," *The Internet Protocol Journal*, vol. 4, no. 3, Sep. 2001; see http://www.cisco.com/ipj.

[9] S. Floyd and K. Fall, "Promoting the Use of End-to-End Congestion Control in the Internet," *IEEE/ACM Trans. Networking*, no. 4, Aug. 1999, pp. 458–72.

[10] V. E. Paxson, "Measurements and Analysis of End-to-End Internet Dynamics," Ph.D. thesis, Comp. Sci. Div., UC Berkeley, Apr. 1997; ftp://ftp.ee.lbl.gov/papers/vp-thesis/

Biographies

SEAN S. B. MOORE [SM] (smoore@cetacean.com) serves as Cetacean Network's chief scientist. He has developed both theory and technology across a range of disciplines including digital signal processing, global climate modeling, global-scale distributed databases, global-scale logistics and scheduling systems, e-commerce, genetic algorithms, automated hardware design, queueing theory, storage networking, and TCP technology. In the past he has served as director of business development, manager of advanced systems, and lead engineer at BBN Technologies; as senior director of R&D at MadeToOrder.com (now BrandVia); as manager of technical services at Tulane University Hospital; and as president of Avatar Consulting. He is a Technical Editor for *IEEE Communications Magazine*. He holds a B.S. in electrical engineering from Tulane University, an M.S. in mathematics from the University of New Orleans (recipient of the 1990 SIAM Applied Mathematics Award), and M.S. and Ph.D. degrees in computer science from Dartmouth College.

CURTIS A. SILLER, JR. [F] (csiller@cetacean.com) is chief technology officer at Cetacean Networks, Inc. He joined the firm after a long career with Lucent Technologies (previously AT&T), where he was a Bell Laboratories Fellow and Distinguished Member of Technical Staff. His experience spans several disciplines, including electromagnetic theory, communication theory, digital signal processing, transmission product conception and planning, and design of wide-area networks. He has played a principal role in designing terrestrial radio systems; satellite, optical, cable, public, and enterprise networks; and video-on-demand delivery architectures. He has participated in IEEE 802, ATM Forum, and IETF standards initiatives; written over 50 refereed papers; co-edited a book on SONET/SDH optical networking; contributed to other reference texts; and holds eight patents. He received an IEEE Third Millennium Medal. He earned B.S.E.E. (with Highest Honors), M.S., and Ph.D. degrees from the University of Tennessee, Knoxville. He is an active member of the IEEE Communications Society. He was a Technical Editor, Senior Technical Editor, and Editor-in-Chief (1993-1995) of *IEEE Communications Magazine*, after which he was appointed Director of Magazines. He was subsequently elected to two Vice Presidential terms (Membership Services and Technical Activities), and has received two service awards from the Society. He currently sits on the editorial board of four journals and is Director of Related Societies. He will become President of the IEEE Communications Society in 2004.

Packet sequencing is an especially attractive basis for attaining utmost quality, efficiency, and security in multiservice, converged IP networks.