

White Paper

Delivering Predictable IP Communications

Quality of Service for Voice over IP

Scott Heinlein
Solutions Marketing



Juniper Networks, Inc.
1194 North Mathilda Avenue
Sunnyvale, CA 94089 USA
408 745 2000 or 888 JUNIPER
www.juniper.net

Part Number: 200126-001 May 2005

Contents

Contents	2
Introduction	3
Challenges to Predictable Quality Delivery.....	3
Supporting Predictable IP Communications.....	4
Comprehensive QoS	4
Forwarding Classes and Loss Priorities	5
Packet Classifiers	5
Transmission Scheduling and Rate Control	5
ASIC-based Processing in Routing and Security Appliances.....	6
Leveraging MPLS for VoIP Communications	6
The Role of Diffserv	7
MPLS Diffserv Traffic Engineering.....	7
Efficiency Mechanisms for Low-speed Links	8
Link Fragmentation and Interleaving (LFI).....	8
Compressed Real Time Protocol (cRTP).....	9
Modular, Fault-Protected Design of JUNOS and ScreenOS	10
Conclusion	10

Introduction

Traditional IP networks were designed to support best effort data applications. The convergence of real-time applications, particularly IP communications, onto traditional IP networks creates new network requirements. One new requirement is the ability to provide acceptable levels of quality for delay sensitive, real-time applications – regardless of unexpected network events.

For Voice over IP (VoIP) communications to be successful, the IP network must be able to mimic the performance and Quality of Service (QoS) attributes of the traditional telephone network. This is a challenge for IP networks because they carry multiple traffic types – voice, video and many types of data. The traditional telephone network, on the other hand, is responsible for voice communications only – which makes delivering predictable voice quality much easier. However, the ability of one IP network to support multiple applications is one of the primary benefits of moving to a converged IP network, and allows the creation of a cost-efficient and flexible infrastructure.

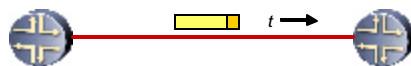
Challenges to Predictable Quality Delivery

Real-time applications are extremely sensitive to latency, jitter and packet loss. Too much of any can create significant quality challenges, rendering IP voice communications impossible. It is critical that IP networks are engineered correctly to ensure IP communications remain predictable, even when the network is reaching full bandwidth capacity. Figure 1 shows the recommended thresholds for delivering high-quality IP voice communications.

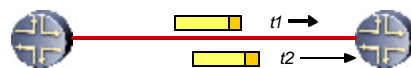
Figure 1: Acceptable Levels Of Latency, Jitter And Packet Loss For Voice

Based on ITU-T Rec. Y.1541)

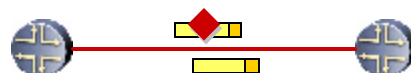
Latency: < 100 ms end-to-end



Jitter ($t_1 \neq t_2$) : < 50 ms



Packet loss: < 10E-3



To meet these requirements, IP networks must address the challenge at both the device level

and the network level. Device-level technology refers to technology built directly into security and routing devices. Network-level technology spreads across the entire network to separate sensitive traffic from non-sensitive for best-effort applications, and provide the means to offer and support high quality IP communications. Juniper Networks supports IETF and other standards to provide a robust and complete set of capabilities in both routing and security products to recognize and support IP communications traffic.

Supporting Predictable IP Communications

Juniper Networks offers a variety of security and routing platforms, purpose-built to handle the stringent requirements of real-time applications. The rest of this application note discusses the means by which Juniper Network implements the identification and processing of real-time application traffic with customer-specified QoS levels. The capabilities discussed include:

- Comprehensive, granular QoS functions on all platforms – enabling priority treatment of IP communications traffic end-to-end at scale.
- ASIC-based packet processing on firewall/VPN appliances and the M- and T-series routers – maximizing throughput because many packet-handling processes are handled by hardware and firmware, rather than software.
- MPLS DiffServ-aware Traffic Engineering – implementing DiffServ enables strict QoS while MPLS Traffic Engineering (MPLS TE) guarantees bandwidth and optimizes the use of network resources
- Enhanced protocol support for voice traffic on low-speed links – Link Fragmentation and Interleaving (LFI) and compressed real-time protocol (cRTP) reduce serialization delay on low-speed links, which increases throughput
- The modular and proven JUNOS operating system software for reliability

Comprehensive QoS

Juniper Networks offers a robust set of features to enable the highest level of voice quality. The Class of Service (CoS) application is used to modify voice packets traveling through the network to indicate the packet's priority and handling. Businesses can customize the CoS application to provide the required level of quality. The process for setting up QoS includes:

- Configure the forwarding classes for transmitting packets
- Define and place packets into the appropriate output queue
- Schedule the transmission service level for each queue
- Manage congestion using a Random Early Detection (RED) algorithm.

Voice or other real-time packets can be differentiated and given preferred treatment across the network, increasing the service's level of quality. Juniper Networks supports the following types of CoS mechanisms for this purpose:

- Differentiated Services – DiffServ as well as the IPv4 and IPv6 header ToS byte settings. The DiffServ code points (DSCPs) in the IP and IPv6 ToS fields determine the forwarding class and loss priority associated with each packet

- Layer 2 (IEEE 802.1p) to Layer 3 CoS mapping – mapping of Layer 2 packet headers to forwarding class and loss-priority values
- MPLS EXP –mapping of MPLS experimental (EXP) bit settings in the MPLS header to routing platform forwarding classes and vice versa

Many individual capabilities combine to provide Juniper Networks' superior QoS performance. Several of the most important of these are briefly discussed below.

For voice traffic to receive the appropriate treatment, a number of steps must be taken for each packet. Upon arrival the packet must be classified according to the information carried in the IP and/or MPLS packet header. The packet is then assigned to the appropriate forwarding class, where it is handled according to the scheduling parameters for that class. Each forwarding class is policed to ensure adherence to predefined limits. Once the packet is ready for transmission on the egress port it is marked to provide consistent treatment downstream.

Forwarding Classes and Loss Priorities

Forwarding classes affect the forwarding, scheduling, and marking policies that apply to all packets as they transit a routing platform; this is especially important for IP communications. The loss priorities allow one to set the discard priority of a packet. If a packet causes a service commitment to be exceeded, it is marked with a high loss priority. If at a later time a packet discard is necessary to avoid congestion the high loss priority packets are dropped first. The forwarding class plus the loss priority define the per-hop behavior (PHB). At least four categories of forwarding class are always supported: best effort, assured forwarding, expedited forwarding, and network control.

Packet Classifiers

Classifiers associate incoming packets with a forwarding class and loss priority and, based on the associated forwarding class, assign packets to output queues. Two general types of classifiers are supported. The first is the code point traffic classifier in which the code point determines each packet's forwarding class and loss priority. The forwarding class and loss priority of a packet is based on DiffServ codepoint (DSCP) bits, DSCP IPv6, IP precedence bits, MPLS EXP bits, or IEEE 802.1p bits. A second, more generalized, classifier is a multi-field classifier that sets the forwarding class and loss priority of a packet based on various fields in the IP packet, much like firewall filtering rules. Codepoint classifiers are most useful in the core of a network where codepoints have been explicitly set; multifield classifiers are used on the edge of the network where no explicit setting exists – for example, to classify traffic from a voice gateway that is unaware of the forwarding policies of the core network.

Transmission Scheduling and Rate Control

A variety of tools are used to manage traffic flows. In general, customers can define the priority, bandwidth, delay buffer size, rate control status, and Random Early Detection (RED) drop profiles that are to be applied to a particular forwarding class for packet transmission. For voice traffic these options mean that the exact requirements for transmission priority, bandwidth, delay, and discard probabilities can be set relative to all other classes of traffic. This determines the order in which an output interface transmits

traffic from the queues. There are four levels of transmission priority ranging from *low* to *high*; packets in the higher-priority forwarding classes are transmitted ahead of packets in the lower priority queues as long as the higher-priority forwarding classes have enough bandwidth credit. Furthermore, one queue per interface can be assigned a “*strict-high*” priority, which works the same as *high* priority, but provides unlimited transmission bandwidth. This is an important concession to voice traffic when burst transmissions occur – the network exhibits an elastic response to sporadic burst traffic.

Policers for traffic classes limit traffic of a certain class to a specified bandwidth and burst size. Packets exceeding the policer limits can be discarded, or can be assigned to a different forwarding class or to a different loss priority, or to both. Schedulers do their best to correctly set the transmission characteristics for a class of traffic, such as voice traffic, and the policers enforce the “rules.” This combination ensures that the appropriate priority is given to the voice traffic and that there are adequate resources available to service the voice traffic queues; lower-priority traffic is not allowed to encroach.

When these capabilities are appropriately configured at each router and edge appliance, IP telephony applications are ensured the requisite resources to keep delay, jitter, and packet loss within the mandated bounds.

ASIC-based Processing in Routing and Security Appliances

As previously discussed, the JUNOS operating system provides a comprehensive set of QoS features to support the appropriate level of QoS for IP communications. To ensure QoS is performed without compromising router performance, packet processing ASICs are used on the M- and T-series routing platforms, and the J-series routers include protected CPU resources for specific QoS functions. These high-performance designs enable a Juniper Networks routers to process all packets at high speeds under all network conditions. Without this functionality, QoS features could degrade the performance of the router – eliminating the benefits of QoS and degrading overall network performance.

Leveraging MPLS for VoIP Communications

In traditional voice networks, traffic engineering is used to achieve performance objectives such as optimization of network resources and placement of traffic on particular links. To gain this same ability in an IP network requires computing a path from source to destination that is subject to a set of constraints, and then forwarding traffic along this path. The explicit routing capabilities of MPLS allow the originator of the LSP to do the path computation, establish the MPLS forwarding state along the path, and map packets onto that LSP. Once a packet is mapped onto an LSP, forwarding is done based on the label, and none of the intermediate hops makes any independent forwarding decisions based on the packet’s IP destination.

MPLS can provide additional benefits for supporting VoIP communications. While diffServ supports multiple classes of service for specific treatment but does not specify a route path through the network. In addition, DiffServ alone does not guarantee adequate bandwidth resources for a specific application. If voice traffic follows a network path with insufficient resources to meet the performance criteria for jitter and latency, for example, voice quality will not be adequate. In principle, this problem could be solved by over-provisioning resources to avoid congestion altogether. Unfortunately, besides being wasteful and

inefficient, this approach cannot provide any guarantees when congestion is caused by link or node failures.

The Role of Diffserv

DiffServ determines the QoS behavior of a packet at a particular node in the network. This is called the per-hop behavior (PHB) and is expressed in terms of the forwarding class that a packet experiences. The PHB translates to the packet queue used for forwarding, the resources (buffers and bandwidth) allocated to each queue, the frequency at which a queue is serviced, as well as the drop probability in case the queue exceeds a certain limit. The four general per-hop behavior categories are:

- Best effort (BE) traffic receives no special treatment.
- Expedited forwarding (EF) traffic encounters minimal delay, low loss, low jitter, and assured bandwidth end to end. From a practical point of view, this means a queue dedicated to EF traffic for which the arrival rate of packets is less than the service rate, so delay, jitter and loss due to congestion is unlikely. Voice and video streams can be mapped to EF: they have constant rates and require minimal delay and loss.
- Assured forwarding (AF) traffic offers finer CoS granularity. A queue number and a drop profile can define each PHB. The AF PHBs are applicable for traffic that requires rate assurance but not bounds on delay or jitter.
- Network control (NC) traffic to carry routing protocol exchanges. These packets cannot tolerate loss, but can accept delay.

DiffServ provides differential forwarding treatment to traffic, thus enforcing QoS for different traffic flows. It is a scalable solution that does not require per-flow signaling or maintenance of the state parameters in the core. However, it cannot guarantee QoS if the path followed by the traffic does not have adequate resources to meet the QoS requirements.

MPLS Diffserv Traffic Engineering

MPLS Diffserv TE combines the advantages of both DiffServ and MPLS TE. MPLS Diffserv TE makes MPLS-TE aware of classes of service, allowing resource reservation with CoS granularity and providing the fault-tolerance properties of MPLS at a CoS level, thus ensuring adequate resources are available on a per-application level.

MPLS DiffServ TE furnishes the strict service guarantees that allow an enterprise to properly support converged, but radically different, services like IP communications, e-mail, web access, and mission-critical transaction support. With its JUNOS software, Juniper Networks provides a standards-based, interoperable, and scalable implementation, along with tools for ensuring that traffic stays within the limits of the resources reserved for it.

With an infrastructure based on Juniper Networks platforms, MPLS priorities can be established that ensure that IP telephony traffic follows paths with the proper resources for forward it and pass it efficiently through the network. These priorities can give voice MPLS LSPs greater importance than others, allowing them to use whatever resources are necessary to take the shortest path across the network or find the fastest alternative route in the event of a link failure. Priorities are assigned by indicating an LSP's importance relative to other LSPs. With MPLS DiffServ-aware Traffic Engineering, the relative proportion of

traffic on different links can be configured to stay constant. This is particularly useful for protecting voice traffic in the network.

With the Juniper Networks implementation of DiffServ MPLS TE, LSP policers can police traffic on a per-LSP basis. In this way LSP policers ensure that the traffic forwarded through an LSP stays within that LSP's bounds and prevents a misbehaving traffic source from degrading the QoS guarantees on other, well-behaved, LSPs.

MPLS Traffic Engineering (TE) enables resource reservation along the path, fault-tolerance of the path, and optimization of transmission resources. MPLS-TE sets up label switched paths (LSPs) along links with the necessary resources, thus ensuring that bandwidth is always available for a particular flow. LSPs are established only where resources are available, so over provisioning is not necessary.

Further optimization of transmission resources is achieved by routing LSPs to take longer paths if the available resources along the shortest path are not sufficient. An added benefit of MPLS is that built-in mechanisms such as link protection and fast reroute provide resilience in the face of failure.

Another MPLS enhancement that optimizes the bandwidth for IP telephony applications is MPLS auto-bandwidth. This enhancement of MPLS TE reduces the administrative overhead involved in setting up label switched paths and improves bandwidth efficiency. It enables the dynamic and automatic adjustment of bandwidth on a path to optimize network performance, which means that LSP capacities can dynamically adjust to voice traffic volume.

Auto-bandwidth adjustment allows MPLS TE tunnels to be set up initially with arbitrary bandwidth and then automatically and dynamically adjust the bandwidth based on traffic patterns, all without traffic disruption. Absent the auto-bandwidth feature a detailed traffic study would have to be conducted to determine the appropriate bandwidth values. These values are static so a poor calculation results in sub-optimal performance and, of course, can't respond to transient load conditions.

Efficiency Mechanisms for Low-speed Links

The delay and jitter performance for voice traffic must be examined for every link in the path, especially on low-speed links. Juniper Networks supports two important efficiency mechanisms for low-speed links: link fragmentation and interleaving (LFI) and compressed real-time protocol (cRTP). LFI addresses the latency problem that exists whenever voice traffic shares the same link with large packets, such as those associated with LAN traffic. LFI fragments large packets in order to reduce their serialization (transmission) delay.

Compressed RTP reduces the serialization delay incurred from the excessive overhead that appears in the protocols used for IP communications – RTP, UDP, and IP.

Link Fragmentation and Interleaving (LFI)

If a low-speed link is used to transmit only voice traffic, the LFI enhancement isn't necessary, because most of the packets are approximately the same size. However, if a link carries both traditional data traffic, such as LAN traffic, and voice traffic, then activating LFI is very desirable. The end-to-end delay target for IP telephony applications is about 100 ms. The typical maximum transmission unit for LAN traffic is about of 1500 bytes,

which takes approximately 215 ms to traverse a 56 kbps line. To illustrate the problem, assume there are two prioritized output queues: an expedited-forwarding queue for high-priority voice traffic (small packets) and a best-effort queue for the traditional LAN traffic (large packets). If packets are in each queue the high-priority traffic will be serviced first, and there is no problem. If the expedited-forwarding queue is empty then a packet is taken from the best-effort queue. Unfortunately, once the large low-priority packet begins transmission it cannot be stopped. If a voice packet arrives in the high-priority queue during the transmission it will have to wait until the transmission of the large packet has completed. In the worst case, where the voice packet arrives just as the data packet begins transmission, the latency is pushed to an intolerable level – the voice traffic may have to wait 215 ms before its transmission, which exceeds the voice delay target.

The link fragmentation and interleaving (LFI) enhancement fragments data packets of a specified length so smaller, high-priority voice packets can be interleaved among the fragments. When the peer interface receives the fragments, it reassembles them into the original packet. None of the real-time voice packets are fragmented. The net effect of this operation is to stabilize the overall delay and provide a bounded latency for the voice traffic at the expense of the data packets (which, by definition, are delay-insensitive on a best-effort basis). Practically speaking this means that once the fragmentation takes place all packets transmitted on the low-speed link are approximately the same size and therefore experience the same delay characteristics. The serialization delay is rendered approximately constant for all packets.

Compressed Real Time Protocol (cRTP)

The real-time protocol (RTP) is the Internet standard for the transport of real-time traffic, particularly voice and audio traffic. RTP provides QoS feedback from a receiver to the transmitter and supports synchronization of different media streams. The RTP protocol unit consists of a data portion and a header portion. The data portion supports the real-time properties of IP communications by providing timing reconstruction, loss detection, and content identification. For compressed-payload audio applications, RTP packets typically have a payload of only 20 to 160 bytes. The header portion of RTP, at a minimum of 12 bytes, is relatively large. This overhead, combined with eight bytes of overhead from UDP and 20 bytes of overhead from IP, brings the total overhead to about 40 bytes. On a small voice packet, 40 bytes of overhead is quite significant. This large overhead for voice packets increases the serialization delay for every transmission, which may seriously affect the end-to-end latency for the application.

The Compressed RTP (cRTP) protocol enhancements compress the 40-byte IP/UDP/RTP header down to 2 bytes, which greatly improves the quality of IP communications over low-speed links. How can a 40-byte overhead be reduced to two bytes and still recover the original information, as required by the receiver? Compressed RTP relies on the fact that although several fields change in the packet as it moves node to node the difference from packet to packet in the same flow is often constant. Taking into account these small changes the decompressor at the receiving end can reconstruct the original header with no loss of information by adding the incremental changes to the saved uncompressed header as each compressed packet is received. CRTP is implemented as a hop-by-hop compression scheme.

During compression of an RTP stream, a session context is defined between the transmitter (the compressor) and the receiver (the decompressor). For each context, the session state is established and shared between the compressor and the decompressor. Once the context state is established, compressed packets may be sent. The context state consists of the full

initial IP/UDP/RTP headers, plus several other values, such as a link sequence number. The context state must be synchronized between compressor and decompressor for successful decompression to take place. In practical terms the context state is established at the start of the voice exchange and is maintained throughout the call; compression can be done for the duration of the call, which greatly reduces the total latency.

Taken together the compressed real-time protocol and the link fragmentation and interleaving option greatly improve the QoS performance for IP telephony on low-speed links. Compressed RTP is used on a low-speed link even if only voice traffic is carried on the link. Both cRTP and LFI are used to optimize transmission characteristics when there is a mixture of traffic types on the same low-speed link. Either protocol is considered only for low-speed links; on high-speed links (e.g., faster than about 1.5 Mbps) the trade-offs are not worth the overhead imposed by either cRTP or LFI.

Modular, Fault-Protected Design of JUNOS and ScreenOS

Achieving suitable QoS goals for voice demands reliable, robust operating software. Juniper Networks J- and M-series routers for businesses utilize the JUNOS operating system. The top 25 service providers globally utilize the same JUNOS operating system in their networks. JUNOS is a proven operating system businesses can trust to run their mission critical applications. Juniper Networks' approach to reliable and available software to support differentiated services relies on three important software strategies.

First, the system architecture separates the control plane and forwarding plane. If the control plane should fail, the forwarding plane continues to support the differentiated services at the prescribed QoS levels. Second, the modular software architecture enables a controlled response when a software failure occurs, which often means that failure recovery doesn't require a system reset, and the handling of IP telephony traffic, for example, continues. A third important factor is that the number of variations in the software is minimized. Juniper Networks deploys a uniform code base for JUNOS. For example, each quarterly release of the JUNOS software runs consistently on all of Juniper Networks routing platforms. The same incarnation of the class of service application and support exists on all platforms.

Conclusion

Converging voice and data services over the same IP-based network and supporting each class of service in a manner consistent with their traditional requires the ability to offer and support differentiated services. Each service type has specific quality of service attributes that must be observed. Voice, in particular, demands low latency, low delay, and low loss if the IP telephony applications are to mimic the traditional offerings from the PSTN. Juniper Networks provides a number of capabilities that enable an enterprise to build and use an IP network in this manner. The DiffServ approach enables the creation and support for a variety of classes of service, ensuring that arriving packets are properly classified, scheduled, and transmitted node to node. MPLS Traffic Engineering is combined with DiffServ to further refine the network service. DiffServ-aware MPLS TE not only supports differentiated services but also guarantees priority LSPs that have appropriate bandwidth guarantees.

For low-speed links both link fragmentation and interleaving and compressed RTP are supported, which reduces the serialization delays on outbound links due to large data packets and excessive packet overhead.

Most important, all Juniper Networks platforms are purpose built to support the QoS requirements of IP communications and other real time applications.

Copyright © 2005, Juniper Networks, Inc. All rights reserved. Juniper Networks, the Juniper Networks logo, NetScreen, NetScreen Technologies, the NetScreen logo, NetScreen-Global Pro, ScreenOS, and GigaScreen are registered trademarks of Juniper Networks, Inc. in the United States and other countries.

The following are trademarks of Juniper Networks, Inc.: ERX, ESP, E-series, Instant Virtual Extranet, Internet Processor, J2300, J4300, J6300, J-Protect, J-series, J-Web, JUNOS, JUNOScope, JUNOScript, JUNOSe, M5, M7i, M10, M10i, M20, M40, M40e, M160, M320, M-series, MMD, NetScreen-5GT, NetScreen-5XP, NetScreen-5XT, NetScreen-25, NetScreen-50, NetScreen-204, NetScreen-208, NetScreen-500, NetScreen-5200, NetScreen-5400, NetScreen-IDP 10, NetScreen-IDP 100, NetScreen-IDP 500, NetScreen-Remote Security Client, NetScreen-Remote VPN Client, NetScreen-SA 1000 Series, NetScreen-SA 3000 Series, NetScreen-SA 5000 Series, NetScreen-SA Central Manager, NetScreen Secure Access, NetScreen-SM 3000, NetScreen-Security Manager, NMC-RX, SDX, Stateful Signature, T320, T640, and T-series. All other trademarks, service marks, registered trademarks, or registered service marks are the property of their respective owners. All specifications are subject to change without notice.

Juniper Networks assumes no responsibility for any inaccuracies in this document. Juniper Networks reserves the right to change, modify, transfer, or otherwise revise this publication without notice.