The ITB Journal

Volume 5 | Issue 1 Article 31

2004

Implementing Test Patterns to Dynamically Assess Internet Response for Potential VoIP Sessions between SIP Peers

Declan Barber
Institute of Technology Blanchardstown, Ireland., declan.barber@itb.ie

Implementing Test Patterns to Dynamically Assess Internet Response for Potential VoIP Sessions between SIP Peers

Declan Barber, Gavin Byrne & Conor Gildea
Institute of Technology Blanchardstown,
Declan.barber@itb.ie

Abstract

The capability of VoIP to provide internet telephony is limited by the lack of homogeneous quality of service (QoS) mechanisms in the Internet. Whereas approaches which reserve QoS resources will work well in an end-to-end managed environment, they are not automatically suited to the heterogeneous nature of the Internet. It may be possible to adopt the 'chirp-sounder' approach uses in establishing the optimal frequency channel for a high frequency (HF) radio transmission which dynamically samples a range of possible transmission channels and uses the echoing of a an established test pattern to ascertain the quality of the potential links. The optimal' channel' can then be selected for transmission. By repeating the process at intervals during the call, transparent handover can be achieved if the current channel deteriorates. This article asks if such an approach can be adapted to suit voice over IP telephony across the internet, specifically in relation to the Session Internet Protocol (SIP). SIP is an Internet-based protocol for establishing real-time end-to-end conference calls between peers. It already includes a mechanism, through the Session Description Protocol (SDP), of establishing the lowest common media capability available on both peers, but currently has no mechanism for establishing if the proposed media connection has adequate latency or packet loss performance to support real-time voice packets. This article asks if SIP should be extended to include such functionality and proposes the adoption of a client/server based measurement-based approach to control call admission.

Introduction

The Internet Engineering Task Force (IETF) Session Initiation Protocol (SIP) and the associated Session Description Protocol (SDP) are emerging as simple but effective protocols for establishing real-time single and multiparty voice or multimedia calls in IP networks. In terms of IP-based real-time multimedia transfer across the Internet, voice traffic is more sensitive to *loss* and *delay* than video, even though it requires far less bandwidth. If voice over IP (VoIP) is to become a realistic replacement for standard circuit-switched telephony services, users must experience the same consistently high-quality service they have become used to with traditional circuit-switched telephony. A lot of work is being done to develop quality of service (QoS) mechanisms for the Internet which will provide bandwidth guarantees and servicing priority for voice. A number of mechanisms have emerged for providing QoS for VoIP traffic. These mechanisms ultimately rely on marking voice packets so that bandwidth and improved packet servicing can be assigned to them as they cross an increasingly intelligent Internet, rather than merely providing 'best-effort' IP delivery. A related but significantly different approach, which draws on traditional circuit-switched telephony, is to try and establish if an adequate call-path is available and only to allow the call to occur if it is. This article asks if such an approach is suitable for SIP based VoIP calls, and if so, how the SIP/SDP protocol standard might be extended to include such a call admission mechanism.

SIP Overview

SIP is a relatively-new (1999) simple ASCII-based signalling protocol that uses application-layer requests and responses to establish one-to-one or conference multimedia communication between peer end-nodes on an IP network. It is an Internet Engineering Task Force (IETF) standard. It is compatible with existing Internet protocols and extensions such as HTTP, SMTP, LDAP and MIME and ultimately supports two important aspects of multimedia calls: ***call signalling*** and ***session management***. Once a session has been established, SIP the IETF standard Real Time Protocol (RTP) is used to transfer media. The operation of a SIP-based call can be summarised as follows:

- A calling node (A), determines the location of the desired peer node (B) using standard Internet address resolution, name mapping and redirection
- SDP is used to determine the lowest common media capability that both A and B have – this is primarily a matter of agreeing a common codec from a set of available codecs
- A uses SIP to identify if B is available or not
- If the call is possible, a two-way RTP session is established between A and B
- The voice or other media is transferred using RTP/RTCP unicast or multicast traffic.
- SIP supports the transfer, call-holding, addition of other peers or other conference type functions during the call
- SIP terminates the call when media transfer is complete

SIP peers are called User Agents (UAs) and can operate as a client (call requesting party) or as a server (call responding party). A SIP peer would normally only operate as one or the other during any given session. Users are identified by unique SIP addresses based on the existing IETF formats but with a SIP flag e.g. *sip:joe.bloggs@itb.ie*. Users register their address (which is bound to the corresponding IP address of whichever terminal is used to register) with a local SIP Registration server which makes the information to a location server as required. Other types of SIP servers include proxy servers, for call-forwarding and SIP redirect servers, which call location servers to determine the path to the called peer and informs the calling peer when of the path when it learns it. Typical SIP peers include SIP-based applications or SIP phones. SIP gateways provide call control, translating calls between SIP endpoints and non-SIP endpoints (e.g. PSTN or GSM phones).

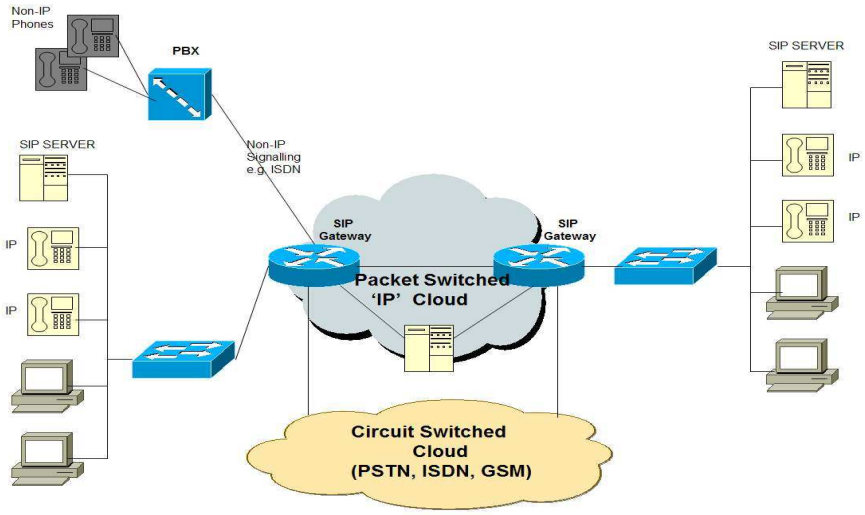

Figure 1: **VoIP (SIP) Infrastructure with Gateway Support to Circuit Switched Networks**

A transitional and convergent SIP-based VoIP Internetwork, incorporating support for existing local Private Branch Exchange (PBX) telephony and gives access both to packet and circuit switched wide area networks is shown below in Figure 1. This is the reference architecture we have adopted for this research.

QoS issues for VoIP

Voice over IP (VoIP) is extremely sensitive to loss and delay, even more so than video. In order for VoIP transmissions to be intelligible at the receiver, the voice packets should not be subject to excessive loss, (which is essentially a bandwidth issue) or variable delay (which is essentially a congestion issue). If VoIP is to become a realistic replacement for standard circuit-switched telephony services, users must experience the same consistently high-quality service they have come to expect from existing telephony. Some guidance in relation to VoIP includes:

- Voice packets are usually transmitted every 20ms
- Sufficient bandwidth provisioning should include consideration of both voice payload (e.g. 64 kbps for standard PCM voice) and associated IP header size (e.g. a further 16kbps).
- Voice packet round-trip time should not exceed 250 milliseconds
- Packet loss should be less than 1%
- Jitter (i.e. variable delay) must be minimised – delay variation should not exceed 100 ms, and ideally should be less than 10 ms

In general, VoIP can guarantee high-quality transmission of voice only if the bandwidth exists and voice packets are given priority over other less-sensitive traffic. The first step in providing QoS for voice packets is to mark and classify the packets so that network devices can discriminate between the time-sensitive voice packets and non-sensitive packets as they traverse the network and create different service levels for each class. Marking can be achieved in a number of ways. A common static method involves setting the IP *Precedence* bits in the IP header (the first three bits of the Types of Service (ToS) field in the IP header). This technique has now been extended to all marking using the first six-bits of the ToS field. These six bits represent the Differentiated Services Code Point (DSCP) and can be used to provide improved differentiated services to packet flows. The first three bits in DSCP are called the *class selector* bits and are compatible with precedence – the higher the decimal value, the higher the priority assigned to the packet. The next two bits are used to define the drop precedence (in the case that the network is congested and the final 'set' bit is used to indicate that the packet has been classified. If a device near the edge of a network has already identified a packet as being a VoIP packet (typically based on the protocol and port number in use) and marked it with an appropriate precedence or DSCP as it enters the internetwork, subsequent network devices can then classify the traffic by matching these bits. The appropriate QoS can then be applied.

Once traffic has been placed into QoS classes based on their QoS requirements, you can then assure priority treatment through an intelligent and configurable queuing mechanism. For VoIP traffic, the key issue is to ensure minimal latency, packet-loss and variation in delay. A range of queuing techniques exist but the preferred option for VoIP is a *priority* queuing scheme where packet classes can be prioritised to be sent before other less sensitive traffic and yet the remaining bandwidth can still be managed to meet lower priority traffic requirements. Typically between three and four queues are configured for High, Medium, Default and low priority classes of traffic. The network device process scheduler services the priority queue first (this can be limited to a configured maximum packet-rate) and the remaining queues are then serviced, using the remaining bandwidth (to a configurable proportional use-level). When the remaining bandwidth is apportioned to the non high-priority queues using an algorithm such as the Weighted-Fair-Algorithm, and packet-class is used to decide which queue the packet is placed on, this queuing technique is known as Low Latency Queuing (LLQ).

Even when voice packets have been classified and queued using low latency techniques, problems may still arise by the voice packets get trapped behind larger data packets in an environment in which bandwidth constraints enforce even minimal service-levels to non-priority queued traffic. A key contributor to variable delay is the transmission of large data packets

between small voice packets on a network. If the configured data packet size is such that it may hold a voice-packet in a transmission queue and force it to exceed the acceptable delay for voice packet intervals at the receiver, then the data packets need to be fragmented. On low speed links, the data fragment should typically take less than 10ms to transmit but should never be lower than the VoIP packet size. The VoIP packets can then be interleaved between them, minimisimg variation in delay at the receiver.

A final technique worth referring to is the compression of the IP RTP header in point-to-point VoIP calls (e.g. between gateways). IP RTP can reduce the 40 byte header of a voice packet to just two bytes, thereby significantly reducing the bandwidth required to transfer voice. Although this comes at the price of increased processing, it can be of particular value on point-to-point low speed links.

In VoIP the amount of bandwidth required for a call will depend primarily on the codec selected and typcially ranges from 80 kbps (for 64kbps encoded speech using G711 a-law or u-law encoding) to 26 kbps (for 8 kbps encoded speech using G.729 encoding). This can be reduced somewhat if header compression is used for the RTP packet to the range 67 to 11 kbps.

Will the Internet Support his Call?

In traditional and mobile telephony, a placed call may be rejected by the local exchange if the circuit-switched connection cannot be made for resource shortage reasons. When a call is placed you either get a guaranteed dedicated connection or you get no connection at all. It is possible to transfer this rationale to packet-switched voice calls. In VoIP, it may be better to deny a VoIP call than to allow it to proceed in a network where the requisite bandwidth and QoS resources are not available at the time the call is placed. If the call went ahead, it would experience unacceptable and intermittent service, resulting in packet loss and excess latency. It could also affect other existing voice calls detrimentally. This is different from the QoS techniques discussed above insofar as it takes place *before* voice packets belonging to a requested call are allowed to be transmitted – it is basically a process to make an informed decision on whether to allow a call to proceed or not, or even to discern from a number of available routes the most feasible route for VoIP traffic. It is typically made based on one or a combination of local parameters, estimates of network congestion and the known shortage of requested QoS resources. As such, it could prevent excess voice traffic from getting onto the network and thereby also protect existing voice calls from being adversely affected by new calls.

Such decisions can be made based on local mechanisms such as the state of the local node and its interfaces, call volume restriction or some other locally known or configured parameters. Although a valuable decision-making component, to allow a call to proceed based solely on local mechanisms to and without any knowledge of network congestion is incomplete. A more evolved approach is to calculate the resources needed before a call is made and to request that the required resources are reserved for the call. This means that each network device along the call-path sets aside a subset of its resources to support the call and if any device cannot, the decision to abort the call may be made. Based on the response to the request for resources, a more informed decision can be made on whether to allow the call proceed or not. This approach is appropriate in an environment which is managed by a single administration from end-to-end but is not appropriate from a heterogeneous environment like the Internet. A compromise between using mechanisms using solely local information and resource reservation schemes could include a range of measurement-based techniques which gauge the network congestion in advance of making a call and make a decision based on the current *network* state. Unlike resource-based mechanisms, these techniques do not guarantee service resources and the measurement provides a basis for *estimating* if the state of the network will support the call. Although the local mechanisms are generally always pertinent to CAC decision-making, resource-based CAC is only possible if the calling/called parties are fully aware of the network topology and have some access to the intermediate backbone devices. This makes it impractical for the Internet, at least at the present time. Test pattern packets could be sent across the network to the destination node, which then return the pattern to the source node. By measuring the round-trip response, the source can estimate the loss and delay characteristics on the network path at the current time. Such techniques can essentially be independent of the network topology and will work transparently across the backbone without requiring any service management cooperation. For this reason, we are suggesting that any SIP-based VoIP deployment that uses the Internet as its backbone should adopt a combination of local and measurement-based techniques for decision-making. Based on the measured values and the estimation of network loss and delay characteristics, the call can either be allowed to proceed, refused or rerouted.

The Basic Research Idea

Although traditional 'ping' packets will give a rough evaluation of the network resources, a more appropriate approach, from a voice perspective, is to use a test pattern based on a series of realistic voice RTP packets which have been derived from the lowest common codec identified by the Session Discovery Protocol between the voice peers. Once the two RTP

channels have been established by SIP the test pattern would be sent from the source to the destination and 'bounced' back to the source. The returned test pattern could then be measured to establish the level of network congestion and the network's ability to support the call. The primary parameters measured to establishing the go/no-go decision or even route selection would include packet loss and delay characteristics. Measuring these characteristics would give a strong indication of bandwidth availability and congestion.

In VoIP, the amount of bandwidth required for a call will depend primarily on the codec selected and typically ranges from 80 kbps (for 64kbps encoded speech using G711 a-law or u-law encoding) to 26 kbps (for 8 kbps encoded speech using G729 encoding) for packet header and payload. This can be reduced somewhat if header compression is used for the RTP packet to a range of 67 to 11 kbps. Different approaches could be taken to assembling the test pattern packets: packets for a specified test pattern file could be dynamically generated by the selected codec scheme once SDP has identified the common codec or a library of prepared test packets could be available. Alternatively, the appropriate test pattern could be selected from a library of codec specific test patterns already existent on the peer. The client server character of a SIP user agent provides an appropriate underlying request/response paradigm between the peers. A call originating client transmit could send a series of RTP packets, with the appropriate precedence value 'typically 5 or '101' for voice) or DSCP value, requesting the server to echo the test pattern. The test pattern is delivered across the network to the destination SIP 'server' which responds by retransmitting what it has received to the source 'client'. The port numbers used for the connection correspond to the UDP port numbers already decided for the potential RTP voice session or a specific port could be used. Figure 2 below describes the basic process.

The bandwidth perceived by the client can be considered as:

$$\textbf{Network Bandwidth Perceived by SIP Client} = \frac{\textbf{Byte count transferred}}{T_{RT}}$$

Where T_{RT} = round-trip time

T_{RT} can be measured with a time that starts when the last byte of the test pattern has been transmitted and which stops when the last byte has been received back. This simple equation could be improved by subtracting the overhead taken by the destination to process the test pattern. This processing time will be dependent on the processing speed of the terminals in use e.g. it may be a workstation with a soft client or a SIP-enabled IP phone. For a given test pattern, it will be possible to derive good approximations based on the number of packets needed to create the test pattern for a specific codec, the number of bytes per packet, the

number of bytes required for data Link Layer de-encapsulation/re-encapsulation and whether compression is used or not.

$$\textbf{Network Bandwidth Perceived by SIP Client} = \frac{\textbf{Byte count transferred}}{T_{RT} - T_P}$$

Although not an accurate measurement of real network bandwidth, this perceived bandwidth measurement provides a reasonable basis for estimating of the network current availability. The second measurement to be taken requires the source client to compare the echoed test pattern sequence with the original test pattern in order to establish the packet loss and data corruption. Packet loss above 1% is considered to be unacceptable and would serve as a basis for attempting to reroute or cancel the call.

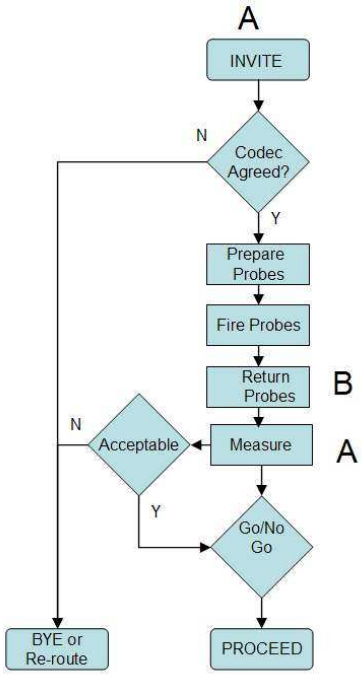

Figure 2: Process flow for establishing network state prior to permitting a voice call

Preliminary Conclusions: Potential Benefits & Limitations of this approach

The main advantage of this type of approach is that it is suitable for use across a backbone network such as the Internet, which some non-homogeneous QoS resources in place, but which you are not in a position to reserve or control without service provider intervention. It is an end-to-end solution, which has no reliance on QoS support in the intermediate service provider networks. These measurements can provide an initial basis for assessing whether a call should proceed. Implementing the process will only require a small amount of code to be added to the SIP application and would sit well with the client-server characteristics of the SIP user-agent. This is consistent with transitional convergent models where IP connectivity forms the lowest common service denominator. It is possible to conceive of this approach interacting with

routing intelligence in order to test multiple available routes (not just the best route indicated by the routing protocol which is usually bandwidth driven) on an application specific basis. In this way, a call with lower bandwidth but better delay characteristics might be chosen in preference to a route solely selected on bandwidth availability. A disadvantage is that it would only work between SIP peers and would not support calls between SIP and non-VoIP terminals. Furthermore, taking these measurements at the beginning of a call and establishing that conditions are satisfactory for the call to proceed at that point is no guarantee that the network will support the call for the entire duration of the call. It only serves as an *assessment* of the capability of the network to support the call at that time. One possibility would be to update the measurements at intervals throughout the call and if the measurements seem to be deteriorating towards a critical threshold, to seek an alternative connection to which the call can then be dynamically transferred in a manner that is transparent to the user. This is analogous to a mobile phone call channel hand-over when moving between cells or when experiencing difficulty. SIP currently doesn't support this functionality but as a protocol, which supports multi-party conferencing, it would not be difficult to extend it to do so. Averages could be maintained on a peer-to-peer basis to identify repetitive success and failure patterns to which different codec strategies could then be applied. For organisations with repetitive patterns of call traffic, the approach could be made proactive to determine network performance systematically between common call points. There is an obvious overhead in terms of delaying the actual call, while the assessment is made, and if subsequent test patterns are sent during a call.

References

Sugih Jamin , Peter B. Danzig , Scott J. Shenker , Lixia Zhang, A measurement-based admission control algorithm for integrated service packet networks, IEEE/ACM Transactions on Networking (TON), v.5 n.1, p.56-70, Feb. 1997

Matthias Grossglauser, David N. C. Tse, A framework for robust measurement-based admission control, IEEE/ACM Transactions on Networking (TON), v.7 n.3, p.293-309, June 1999

H. Schulzrinne et al., "SIP: Session Initiation Protocol", IETF DRAFT, November 2000.

H. Schulzrinne et al., "Centralized Conferencing using SIP", IETF DRAFT, November 2000.

Ulysses Black, Advanced Internet Technologies (Prentice Hall 2001)