VIAVI Solutions

Application Note

Network Performance Testing for VoIP

Analyze network quality of service with real voice

Defining a network test strategy for voice

How accurate is your network test strategy when it comes to voice?

So many voice test strategies are closed! Closed means they concentrate on proving one narrow or specific parameter, almost always within a defined test condition. Take a test strategy based on proving the reliability of a network and the various network components to deliver packets 99.999% of the time with minimum delay. Now add the *condition*, with only one TCP/UDP packet present. Eureka, job done!

On reflection network engineers don't test like this or do they? When a test scenario is developed with a defined set of conditions, it only points to one outcome, the expectant result. Take the scenario of a macro traffic generator, simulating network traffic with layer 2-3 traffic, carrying stateless 4-7 loads i.e. the buffer is stuffed to build out the rest of the packet. The result, 99.999% reliability! Eureka, job done! Ok, there was some human intervention, some of the network/ components under went fine tuning, but none the less, five nines!

As the buffer stuffing contained a bit pattern of a RTP encapsulated voice call, Eureka job done! The network will reliably deliver voice traffic, with 99.999% reliability!

Network Performance Testing for VoIP

Is it this simple to test call performance and voice quality in converged networks? The macro traffic generator has served its purpose, to push traffic conditions into the network, from which network performance measurements can be ascertained. Eureka or is it?

With the macro traffic generator we have succeeded in predefining the *conditions*, in which the outcome can be predicted, the expectant result. A major limitation to this type of testing; is the bit-stuffing of the packet. Essentially, the call has no interaction with real or live network components.

As end-users of telephony services, we know it's a two-way thing. At the other end of the line is another person to whom we talk, laugh, shout, scream or share long pauses with. Every call unique! Therefore each packet is different, each traffic flow is different.

Is there a better way to test network performance with the delay sensitive application of voice? Can a test be created, in which no pre-defined condition influences the end result?

VoIP application note overview

In this application note we explore the concept of an open test plan for VoIP applications, in which the endpoint devices and the call management systems are included. The case for testing networks for VoIP has been outlined many times before, in all cases the focus has been on performance, in particular the reliability of the network to deliver packets.

To date these closed strategies have served well to define how the network and its components are fine tuned. They are a closed strategy as; they offer no insight into various unique elements that make up a phone call. In a converged world of networks, this can be very misleading.

Another case for a more open test strategy is convergence. Voice will share the network bandwidth with many other applications. These applications may include video or data, which means existing VoIP or voice only based test strategies offer no reflection of performance with real world network usage.

Defining an Open Test Strategy for VoIP

Defining call quality

What's different about VIAVI application note versus the thousands of existing and commonly themed VoIP test strategies available on the web? To try and answer this we will use two simple but yet powerful words "Per flow". This application note is a first and is completely unique as it adopts the concept of "Per flow" activities and performance measurements for voice.

Take the following table, STOP – PAUSE! now reflect for a minute on the information shown in the table:

	Call Connect time (ms)	Call Duration (mins)	Call Quality Score
Bob	500	5	3.0
Ann	1200	1	4.2
Tom	500	2	3.9
John	200	7	1.0
Lucy	1000	.05	4.5

Table 1: A sample of subscriber performance data

1. Who is the happiest and will they continue to subscribe to the service?

2. Who is likely to end their subscription?

3. Is an aggregate statistic of 3.4 a fair reflection of the Call quality?

4. Is 720ms an accurate assessment or view of call connection times?

5. Are users using the service exactly the same?

If you skimmed the information in the above table, stop and close this document now or go back up one more time and understand the questions being asked. This is the first step to understanding the relevance of performance using an open test strategy and understanding the relevance of per flow analysis.

Realizing each cell in the above table defines a unique performance value and when applied as an individual row represents a real end-user's quality of experience. It also helps seed the concept of how important per flow performance measurements are.

Traditionally, quality scores are assessed on a column basis i.e. defining an average score, for a particular performance measurement, clearly this approach offers no real value on what really is happening in the voice service.

The blinkered or column aggregate view is bad, a closed strategy. On reflection, if a large enough spread of numbers is averaged the result can be predicted. Bundled attempts at outlining quality based on aggregate views is not suffice, nor is it suffice to test with thousands of flows all with the same buffered packets and sizes.

Returning to the question list above, some of the answers are not apparent, which only serves to highlight the enormous gap between fact and fiction where aggregate performance measurements of macro traffic generators are used to indicate network performance.

Surely it cannot be the case that Ann with a call quality of 1 is happy, or even if the score of 1 reflects anything near the average score of 3.4? To continue to test with a blinkered approach and using static bit patterns offer limited detail terms of individual end-user quality of experience.

As already mentioned earlier, VoIP endpoints tend to have real people connected on the end. Therefore it's fair to conclude that no two endpoints activity is ever likely to be the same. As unique individuals we will interact with a handset differently, this may include the speed in which we dial a number, how fast we speak or even the duration of a call. The requirement is now for a completely configurable layer 4-7 entity which can accommodate different activities and scenarios.

Per flow, the relevance of individual endpoint performance

What is the relevance of per flow and real world users?

Per flow in its purest form means no two flows are the same, exactly how real world users use communications today. Per flow coupled with emulation enables testing with flows with precise details of the device type and properties, layer 2-3 configuration up to the SIP application request/response parameters through to the user activity in layer 7.

The true benefit to per flow testing for voice is the ability to provide performance metrics on each and every endpoint and the associated SIP/RTP flows, an example of which is shown in the data Table 1 (page 2).

Open test strategies, are without predefined conditions, as already suggested aggregate performance measurements inhibit open test strategies. A granular approach ensures that no single endpoint's condition influences the end result.

To further enhance the openness of the test strategy, the test should include a number of unique subscribers making calls, of varying lengths at varying times, all with unique properties.

Note: In reference to VIAVI TeraVM a flow is 5-tupple, in which all elements of the flow are configurable:

1. Source IP

2. Destination IP

3. Source Port

4. Destination Port

5. Protocol

Stateful Flows, real network traffic

In the real world, voice conversations are generally two way or bi-directional therefore it should be considered as a pre-requisite for an open test strategy that the traffic flows are stateful i.e. understand and respond to individual and unique control and transport layer requests.

The purpose of using stateful traffic flows in open test strategies include:

1) Real traffic, calls and activity.

2) Using fixed, stateless, bit patterns offer no reflection of real network conditions.

3) Stateful traffic means the end result is not pre-determined.

4) Enable analysis of the impact that network management and control signals have on an individual endpoint's performance.

5) Percentage of overall network bandwidth given over to control signals versus goodput of actual application flows.

Take the example scenario of network congestion, the traffic management policy may include TCP window shaping for improved VoIP application quality. It's essential that the emulated endpoint in the open test scenario is capable of conversing or negotiating this window re-sizing.

Voice Content

In open test strategies it's vital to add real voice samples to each flow. An ideal test will include different voice samples of female and male tones. To further enhance the test the use of different codecs to implement the calls should be included.

This not only helps to help network reliability but is also useful in testing SIP proxy or call handlers to block calls between endpoints which are incompatible.

In short the greater the variety the more open the test becomes, and less obvious the test results become.

Adding mixed traffic flows

Most networks today are utilized to deliver more than voice packets. An open test strategy will strive to include a mix of traffic from a range of applications. The range should not be limited to legal flows but should also include illegal flows such as unauthorized endpoints connecting to the network, security attacks such as DDoS.

Network Congestion or a component's ability to manage flows can be hampered under extreme traffic conditions. In a voice only test strategy this may go unchecked. In a real world deployment this may result in an irregular error caused when an unknown traffic flow is present.

A new and open test strategy for voice

As already discussed there are a number of test strategies available for VoIP, however in this paper we want to identify how we can improve on those old scenarios by utilizing per flow to deliver an open test strategy.

The traditional approach to testing VoIP applications include -

(If used in isolation the individual tests should be seen as a closed test strategy as the performance measurements are isolated to a specific set of measurement parameters.)

VoIP Acoustic Performance – Acoustic based measurements, with an emphasis on loudness, any distortion that may occur due to harmonics.

VoIP Voice Quality – Actual voice quality may be determined using an algorithm, an approach is to statistically analyze the packet headers around the delivered voice payload.

VoIP QoS – Network impairment (Latency, Jitter, Packet Loss/Duplication), Congestion, link or connection failures.

A mechanism to open these closed test strategies, is to implement a subsection of them all, which can then be used to influence an overall measurement of Quality of Experience (QoE).

Quality of Experience is an effective open test strategy as it essentially, says yes we want to see how the network will cope at various stages of the call lifecycle, with various user scenarios, but more importantly how will these performance measurements impact the experiences on an individual emulated endpoint?

Network benchmarking tests such as throughput or busy hour calling attempts are fine for defining the network limitations but offer no valid data in terms of actual endpoint quality of experience on a per individual call basis.

The relevance of unique endpoints in testing a Call Manager

Another example reason on how an open test strategy adds benefit and an indirect reason to using per flow comes in the form of call management system testing.

It's near to impossible for two endpoints to achieve registration with the same configuration (this should be considered as an actual test), or even more unlikely that it would cope with several thousand registration requests per second; therefore a more granular approach is required.

Can the network or device under test select and prioritize a lifesaving call?

Ultimately voice calls should be handled and prioritized with zero error, and open test strategies are used to determine how efficient the network or device under test is at call handling.

In many VoIP enabled services, the text "This application should not be used for emergency calls" is all too familiar sight. However, in the development of an open test strategy and using per flow it's possible to emulate real world scenarios with a high volume of network traffic flows and in that flow may be a single individual emergency call. Can your network connect a lifesaving call under extreme conditions?

Performance testing VoIP at scale

Open test strategies begin by assessing quality on a per individual endpoint basis to determine a single user's per flow performance.

Endpoint Assessment:

- Configuration of the endpoint – how fast can the call management system supply a TFTP file? Determine access and download times?
- Provisioning the endpoint – IP address assignment, how fast can the DHCP server respond with a valid address?
- Registration of the endpoint – how quick is the SIP proxy/registration authentication process?
- Inbound / Outbound Connectivity – Is the SIP proxy server routing configuration correct?
- Network Overload – What impacts have retransmits for SIP requests on the call manager?
- Media Encryption – Can calls be sustained with encrypted media flows (SRTP)?
- Call Media Quality – How much latency can an endpoint tolerate?

The above is a sample of just one endpoint's media session, in real world terms the process occurs over thousands of unique endpoints. Therefore testing needs to be scalable, not forgetting that communication is a two way flow, with various handshaking and negotiations e.g. (SIP Invite/Bye, TCP window resizes, etc).

Scalability Test Scenarios:

- Oversubscription of endpoints – Examine the fallout when all licences are in use, especially when new endpoints attempt to register?
- Call throughput – Determine the maximum concurrent calls possible, in a pure VoIP only situation. Re-examine when additional traffic types are added to the mix?
- Emergency Call handling – Using the above examples, examine how 911 calls are handled.

Testing VoIP applications with TeraVM

Introducing TeraVM

VIAVI TeraVM is a fully virtualized IP test and measurement solution that can emulate and measure millions of unique application flows. TeraVM provides comprehensive measurement and performance analysis on each and every applications flow with the ability to easily pinpoint and isolate problems flows. TeraVM is deployed on any industry standard hardware (e.g. Cisco, Dell, HP, IBM) with any major hypervisor (e.g. VMware ESXI, Hyper-V, KVM).

The cost benefits to a per flow test strategy

VIAVI per-flow emulation and test approach greatly simplifies the challenge of testing voice in converged networks in a reliable and repeatable manner. Using VIAVI's virtualized IP test solutions, service providers and network equipment manufacturers can delve down to the properties a single emulated VoIP endpoint to determine the performance in terms of an end-user's quality of experience, plus determine application level of quality on the voice calls in real time. The live or real time analysis is used by VIAVI to determine if any subtle change in the network configuration settings which may impact the latency sensitive voice application

Summary of TeraM's SIP and VoIP test functionality

1. SIP Registration – Test functionality and performance timing. Connect and register with the SIP registration server with unique details and passwords. Determine maximum number of registrations possible on a per second basis.

2. Interact with 3rd Party Servers – Emulated endpoints are fully stateful, and can interact with a number of leading vendor's equipment e.g. connect to a secure call manager through SIP/TLS sessions, plus make RTP and SRTP calls.

3. Call Quality - Establish a baseline reference call for voice quality – establish the most suitable codec and optimal buffer sizes for devices under varying networking conditions.

4. Bulk Call Capabilities – Establish the maximum number of calls possible, use a mix of traffic flows with signaling only and signaling plus media.

5. Load Testing – Test and measure voice quality under varying traffic conditions and network loads, on a per client/per flow basis. Mix several unique individual voice, multicast video and multiple data clients running service applications.

Run true, stateful TCP based application flows along with voice flows with multiple codecs. Exchange real voice, access real multicast streams, e-mail documents, URLs and attachments in order to emulate realistic, per client traffic flows.

6. IPv6 transition testing – Utilize unique layer 2/3 properties, concurrently examine performance of IPv4 versus IPv6 enabled voice applications. Examine performance of dual stack enabled UACs, investigate the ability to register/connect calls using IPv4 or IPv6.

   This provides a unique MAC and IP address per client. The flexible MAC address configuration and unique options assignment is key for validating security in many environments.

7. IMS enabled Multimedia calls – Analyze the SIP registration performance. Examine RTSP performance and the incoming media quality.

8. Secure calls through TLS/SSL – Measure and compare performance of secure (TLS/SSL) and unsecure voice calls. Determine if by varying the codec type has any influence on performance in TLS/SSL sessions.

9. NAT boundary traversal – Measure the effects of NAT boundary traversal (RFC 3261) on call quality.

10. Disruptive flows (P2P, DDOS, IGMP floods, spam, and viruses) – add to the existing test scenarios, test and verify any security and mitigation rules or functionality that may be available.

11. Network device QoS Settings – Run individual voice, multicast and application data on emulated hosts against external voice, multicast and application data servers. Test and verify appropriate QoS mechanisms to use at L2 and/or L3/4 to classify traffic into each service category. Assign VLAN priority (on single and tunneled QinQ) on emulated endpoints and DiffServ/TOS classification for each individual application/service.

Dedicated VoIP Statistics on a per emulated endpoint basis

VoIP UA related statistics

TeraVM's per flow architecture provides performance measurements for each and every emulated endpoint. The following is a summary of the VoIP UA related statistics.

VoIP UA Application Item	Description
UA In RTP Bits/sec	The number of RTP bits/second received in by this UA.
UA Out RTP Bits/sec	The number of RTP bits/second sent out by this UA.
UA In RTP Packets/sec	The number of RTP packets/second received in by this UA.
UA Out RTP Packets/sec	The number of RTP packets/second sent out by this UA.
UA RTP Out of Sequence Packets	The number of packets out of sequence sent out by this UA.
UA RTP Dropped Packets	The number of packets dropped by this UA.
UA Duplicate RTP Packets	The number of duplicate packets received in by this UA.
UA Out Calls Attempted	The number of calls out attempted by this UA.
UA Out Calls Established	The number of calls established by this UA.
UA Out Calls Rejected	The number of calls rejected by this UA.
UA In Calls Attempted	The number of incoming calls that this UA attempted to receive.
UA In Calls Established	The number of incoming calls established by this UA.
UA In Calls Rejected	The number of incoming calls rejected by this UA.
UA Calls Errored	The number of calls with errors logged by this UA.
UA SIP Out Messages	The number of SIP messages sent out by this UA.
UA SIP Messages Resent	The number of SIP messages resent out by this UA.
UA SIP In Messages	The number of SIP messages received by this UA.
UA In RTCP Packets	The number of RTCP packets received in by this UA.
UA Out RTCP Packets	The number of RTCP packets sent out by this UA
UA Registrations Attempted	The number of registrations attempted by this UA.
UA Registrations Successful	The number of successful registrations by this UA.
UA Registrations Rejected	The number of registrations rejected by this UA.
UA Registrations Errored	The number of registrations with errors logged by this UA.
UA Calls Received Ringing	The number of ringing calls received in by this UA.
UA Mean Time to Ringing (ms)	The average time for incoming calls to this UA to ring.
UA Min Time to Ringing (ms)	The minimum time for incoming calls to this UA to ring.
UA Max Time to Ringing (ms)	The maximum time for incoming calls to this UA to ring.
UA Calls Received RTP Packet	The number of messages with RTP packets received by this UA.
UA Mean Time to RTP Packet (ms)	The Mean time for this UA to receive the first RTP packet.
UA Min Time to RTP Packet (ms)	The Minimum time for this UA to receive the first RTP packet.
UA Max Time to RTP Packet (ms)	The Maximum time for this UA to receive the first RTP packets
UA RTP Jitter (RFC 3350) ms	The Jitter per ms.
UA RTP Max Jitter (RFC 3350) ms	The maximum Jitter per ms.

Latency measurements per flow

Item	Description
RTP Latency Packets Measured	Number of RTP packets received on which latency has been measured
RTP Mean Latency (ms)	Mean Latency of the RTP Packets.
RTP Max Trip Time (ms)	The maximum trip time measured for an RTP packet.
RTP Min Trip Time (ms)	The minimum trip time measured for an RTP packet.
RTP Jitter (Latency) (ms)	The RFC 3550 inter-arrival jitter algorithm based on the latency timestamps inserted in the RTP packets.

Passive Analysis – R-factor statistics collected per voice stream

Item	Description
QmVoice MOS	MOS score for this voice stream.
QmVoice RFactor	R-Factor for this voice stream.
QmVoice Stream ID	The RTP SSRC of the audio stream being analyzed.
QmVoice Codec	Voice codec for this stream.
QmVoice In Packets	The number of voice packets received for the stream being analyzed.
QmVoice Dropped Packets	The number of voice packets lost for the stream being analyzed.
QmVoice Out Of Sequence Packets	The number of voice packets received out of sequence for the stream being analyzed.
QmVoice Duplicate Packets	The number of duplicate voice packets received for the stream being analyzed.
QmVoice Discarded Packets	The number of voice packets discarded for the stream being analyzed.
QmVoice Underrun Discarded Packets	The number of voice packets discarded due to under-run for the stream being analyzed.
QmVoice Overrun Discarded Packets	The number of voice packets discarded due to overrun for the stream being analyzed.
QmVoice Mean PDV ms (Packet Delay Variation)	The average instantaneous packet delay variation for the packets received on the stream.
QmVoice Max PDV ms (Packet Delay Variation)	The maximum instantaneous packet delay variation for the packets received on the stream.

Video Quality Metrics collected per video flow on multi-media calls

Item	Description
QmVideo Picture Quality	This is a CODEC dependent measure of the subjective quality of the decoded video stream (0-50).
QmVideo MOS	Mean Opinion Score representing video service picture quality. The score also considers the original video quality (before encoding and transmission) and the video content's sensitivity against video packet loss/discard.
QmVideo Transmission Quality	This is a CODEC independent measure related to the ability of the bearer channel to support reliable video (0-50).
QmVideo Multimedia MOS	A VQmon Mean Opinion Score representing video service multimedia quality. It takes video picture quality, audio quality and audio/video synchronization into account to generate the overall multimedia quality
QmVideo Mean PDV (Average Packet Delay Variation)	The average instantaneous packet delay variation for the packets received on the stream.
QmVideo Max PDV (Maximum Packet Delay Variation)	The maximum instantaneous packet delay variation for the packets received on the stream.
QmVideo Stream ID	Either the RTP SSRC or MPEG2-TS PID of video stream being analyzed.
QmVideo Codec	Video codec for this stream.
QmVideo In Packets	The number of video packets received for the stream being analyzed.
QmVideo Out Of Sequence Packets	The number of video packets received out of sequence for the stream being analyzed.
QmVideo Dropped Packets	The number of video packets lost for the stream being analyzed.
QmVideo Discarded Packets	The number of video packets discarded for the stream being analyzed.
QmVideo Underrun Discarded Packets	The number of video packets discarded due to under-run for the stream being analyzed
QmVideo Overrun Discarded Packets	The number of video packets discarded due to overrun for the stream being analyzed.
QmVideo Duplicate Packets	The number of duplicate video packets received for the stream being analyzed.
QmVideo In I-Frames	The number of I-frames received without impairments due to packet loss and/or discards of the frame itself for this stream.
QmVideo Impaired I-Frames	The number of I-frames impaired due to packet loss and/or discards for this stream.
QmVideo In P-Frames	The number of P-frames received without impairments due to packet loss and/or discards of the frame itself for this stream.
QmVideo Impaired P-Frames	The number of P-frames impaired due to packet loss and/or discards for this stream. This does not include frames impaired due to error propagation through temporal reference.
QmVideo In B-Frames	The number of B-frames received without impairments due to packet loss and/or discards of the frame itself.

Video Quality Metrics collected per video flow on multi-media calls continued

Item	Description
QmVideo Impaired B-Frames	The number of B-frames impaired due to packet loss and/or discards for this stream. This does not include frames impaired due to error propagation through temporal reference.
QmVideo Frames/s	The frame rate of the video stream in frames per second.
QmVideo Frame Width	The width of the video frame in pixels.
QmVideo Frame Height	The height of the video frame in pixels.
QmVideo GoP Length	The Group of Picture length for the video stream. The GOP length is the number of frames between two full images (I-Frames).
QmVideo GoP Type	The Group of Picture type for the video stream.

Video metrics for MPEG2-TS transport enabled calls

Item	Description
QmMp2ts TS_sync_loss	TR 101 290 MPEG2-TS number of occurrences of transport stream sync loss for this video elementary stream.
QmMp2ts Sync_byte_error	QmMp2ts Sync_byte_errorTR 101 290 MPEG2-TS number of occurrences of sync byte error for this video elementary stream.
QmMp2ts Continuity_count_error	TR 101 290 MPEG2-TS number of occurrences of continuity counter error for this video elementary stream.
QmMp2ts Transport_error	TR 101 290 MPEG2-TS number of occurrences of packet with transport error bit set for this video elementary stream.
QmMp2ts PCR_repetition_error	TR 101 290 MPEG2-TS number of occurrences of time interval between two consecutive PCR values more than 40 milliseconds for this video elementary stream.
QmMp2ts PCR_discontinuity_indicator_error	TR 101 290 MPEG2-TS number of occurrences of the difference between two consecutive PCR values is outside the range of 0 to 100 milliseconds for this video elementary stream.
QmMp2ts PTS_error	TR 101 290 MPEG2-TS Number of occurrences of the presentation timestamp repetition period

Dual Hosted VoIP Application Results

The statistics gathered for a Dual Hosted VoIP UA are the same as those gathered for a (single host) VoIP UA. However, as the Dual Hosted VoIP UA can register or initiate calls using an IPv4 or an IPv6 external SIP Proxy, no differentiations will be made between the registration and call initiation results displayed for IPv4 and for IPv6.

Supported Video/Audio and Voice Codecs

TeraVM supports a number of pre-configured codecs, in addition TeraVM supports a configurable CODEC template with flexible Sample Period, Frame Size and Packet Rate.

Video/Audio Codecs

The following are a list of pre-configured video/audio codecs supported by TeraVM, note TeraVM supports concurrent measurement of both the video and audio in real-time:

Video Codec	Audio Codec
JPEG	MPEG-1 Layer 1
MPEG	MPEG-1 Layer 2
H.261	MPEG-1 Layer 3
H.263	MPEG-2 AAC
H.263+	AC-3
H.264	MPEG-4 AAC
MPEG-4	MPEG-4 Low Delay AAC
VC-1	MPEG-4 High Efficiency
MPEG2TS	

TeraVM support for Voice Codecs

The following are a list of pre-configured voice codecs supported by TeraVM:

Codec
AAC-LD
AMR-NB
AMR-WB
H.264
G.711 alaw
G.711 ulaw
G.722
G.723
G.728
G.729
GSM
iLBC (13.33)
iLBC (15.2)
MP4A

VIAVI Solutions

To reach the VIAVI office nearest you,
visit viavisolutions.com/contact

testingvoip-an-wir-nse-ae
30187430 900 0718