Application Note BBAN01
Revision 1- November 13, 2006

Assessing end-to-end VoIP readiness of a network

ByteBlower@excentis.com
www.byteblower.com
www.excentis.com

Contents

1 Introduction

Excentis' ByteBlower is an advanced TCP/IP traffic generator and analyser. A ByteBlower set-up consists of one or more servers, to actually generate and analyse traffic, and a PC running the ByteBlower GUI to control the different servers. This distributed architecture allows traffic testing over large distances. Different servers can synchronize their clocks by means of optional GPS receivers to allow accurate one-way delay and jitter measurements. ByteBlower servers can use commercially available high-end Ethernet switches as multiplexers to attain large numbers of traffic ports.

This application note shows how you can use Excentis' ByteBlower to verify the end-to-end readiness for Voice-over-IP (VoIP) of any network. This can be a cable operator's hybrid fibre-coax network, an xDSL network, a leased line between company sites, or any other network that can carry VoIP.

Both readiness for carrying the actual voice (media stream) as the signalling can be tested.

2 Objectives

For assessing VoIP readiness, the important factors to consider are packet loss, one-way latency, and jitter (which is the variation of latency between successive packets). In general it is also useful to study the evolution of these quantities over time, as the load of other traffic will also vary in time and hence can influence VoIP performance. This should not only be done for the actual voice traffic, but also for signalling traffic, to ensure that calls can be set up in a timely and reliable manner.

As a first step, these quantities need to be measured between the VoIP endpoints, for example between an Analogue Terminal Adapter and a media gateway in a DSL network, or between two cable modems in an HFC network (or actually, Embedded Multimedia Terminal Adapters or E-MTAs as they are called when a phone can be attached).

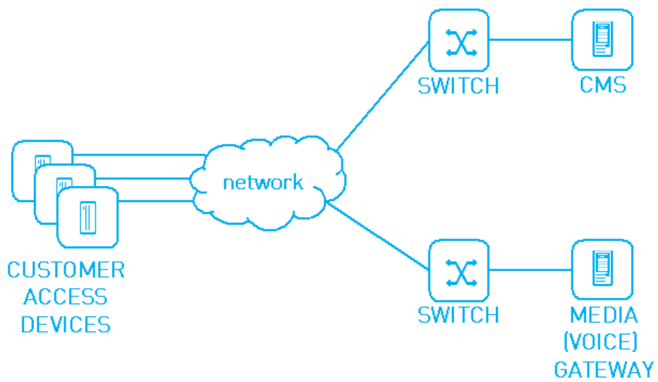

Figure 1: Example network setup

We will use ByteBlower to measure the aforementioned quantities (packet loss, latency and jitter), and this for each direction separately (i.e. one-way), and as well for voice as for signalling traffic. For useful measurements, it is a prerequisite that the network is configured to provide voice and signalling with a certain Quality-of-Service (QoS) level. ByteBlower can then be configured to generate traffic that is as realistic as possible, to make sure that the QoS will also apply to ByteBlower's test traffic.

Of course, in case detailed results for individual links are desired, or in case there is a performance bottleneck that needs to be identified, ByteBlower is the right tool to also do these measurements between any two points in the network.

3 Test description and set-up

3.1 Placement of ByteBlower servers and GUI

Since the goal of our testing is an end-to-end assessment of the network, the definition of the "ends" of the network will determine where to connect the ByteBlower servers. For example, in a cable network, a number of cable modems can be installed in the head-end, connected to different upstream and downstream channels. One ByteBlower server can be used in that head-end location, connected to each of the cable modems. Of course, there is always more than one "end" on a network. The second end, as far as voice is concerned, is typically near a PSTN or media gateway. As far as signalling is concerned, the other end of the network is near the softswitch. In case softswitch and media gateway are located in the same physical neighbourhood, one ByteBlower server can be placed there. In case they are in physically distant locations, there needs to be a ByteBlower server in both of these locations.

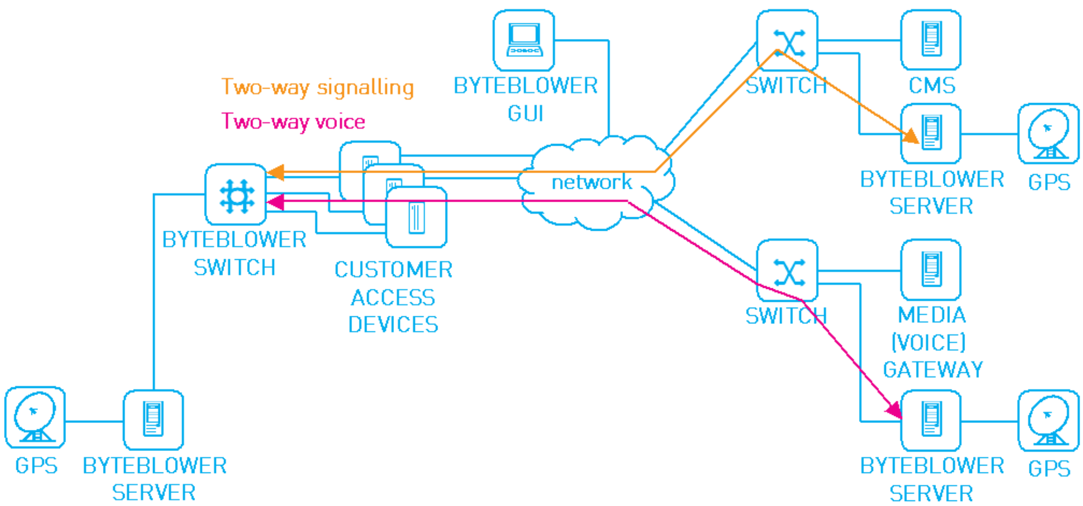

Figure 2: Example network setup

An example is given in the figure above, where one ByteBlower server is used on the side where customer's VoIP devices are simulated (ByteBlower switch behind multiple "customer access devices", e.g. cable modems in an HFC network or DSL modems in an xDSL network). On the other side of the network, there is a softswitch as the endpoint that terminates signalling traffic, and a media gateway, i.e. the endpoint where voice traffic is terminated. These two devices are in geographically distinct locations, necessitating the use of two ByteBlower servers. The

ByteBlower servers are connected to the network "in the same way" the softswitch and media gateway are connected, which typically means in the same IP subnet (and maybe VLAN), and on a port on the same backbone Ethernet switch.

All ByteBlower servers synchronize their clocks thanks to the GPS receiver, and this in turn makes it possible to perform one-way delay measurements with at least sub-millisecond accuracy. Note that the accuracy remains, even if the GPS signal is interrupted for short periods of time.

All ByteBlower servers are controlled by one easy-to-use graphical user interface. This ByteBlower GUI runs on a separate PC in an arbitrary location from where there is IP connectivity with the ByteBlower servers. Note that there is only one standard GUI; there is no need to have specially tailored applications or user interfaces for different types of measurements.

3.2 Defining the flows

The IP flows involved in VoIP are at least the following ones:

- "upstream" voice, i.e. from the customer to the network side;
- "downstream" voice, i.e. from the network side towards the customer;
- "upstream" signalling messages from the customer;
- "downstream" signalling messages from the network.

This means that per simulated CPE device, i.e. per ByteBlower port behind a customer access device, we need four flows.

Thanks to ByteBlower's configuration options, voice and data flows can be modelled in a Flow Template after their respective properties (e.g. 50 packets per second of PCMA-encoded voice in an RTP stream for voice flows, and bursty traffic for the signalling flows - accomplished by ByteBlower's Multiple Burst flow type with a Random Frame Size modifier). By choosing the required UDP port numbers and DSCP/ToS bits, we can make sure that ByteBlower's traffic undergoes the same QoS treatment as actual voice calls would.

For all flows, it is important to measure one-way delay (latency), so we enable the "Latency Distribution" measurement (or alternatively, the more simple "Average latency" measurement). This will also give us the standard deviation of the latency distribution, which is the jitter we want to measure. Packet loss is of course also measured, but ByteBlower always does this by default.

3.3 Setting up ports

A "port" comes in two kinds: a logical ByteBlower port, and a physical network (Ethernet) port. As part of the port configuration step, ByteBlower ports need to be linked to physical ports on the ByteBlower server or switch. This is a process called "docking". While this may seem an awkward indirection at first glance, this concept makes it very flexible to define and later modify ports. This also makes it possible to assign multiple ByteBlower ports to one physical port.

For the ports that simulate customer CPE (VoIP) devices, it is typically very useful to turn on ByteBlower's DHCP feature. In some environments, this may even be a necessity. Settings like IP addresses or the use of DHCP are properties of the (logical) ByteBlower ports. Physical ports have no settings associated with them.

3.4 Running the test

To actually perform the test, the defined flows are assigned to a so-called Scenario, which is the unit of execution of a ByteBlower test.

To study the evolution of the measurements over time, ByteBlower's handy Batch mode is indispensable. The same scenario can be scheduled for automated batch execution (e.g. run the scenario every hour). This yields measurement results per run, and hence over time. This way, long-running tests can be executed without requiring manual intervention.

3.5 Gathering and interpreting results

When a running scenario is completed (or manually terminated), ByteBlower GUI generates a detailed report of all measurements. Per flow, important metrics as packet loss, minimum latency, average latency, maximum latency and jitter and more are reported. In addition, since we asked for a latency distribution measurement, ByteBlower has also kept track of a histogram of latency values.

Example measurement results are shown below.

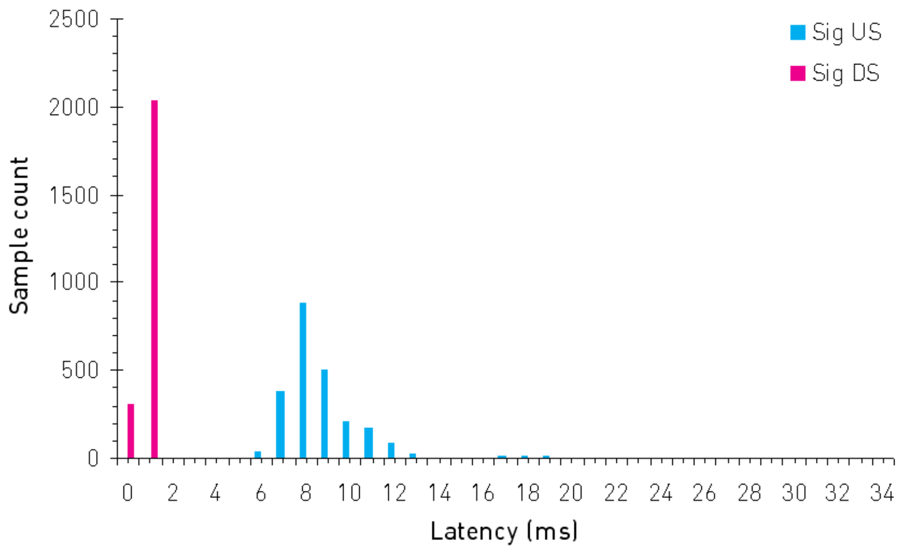

Figure 3: Upstream and downstream signalling latency distribution

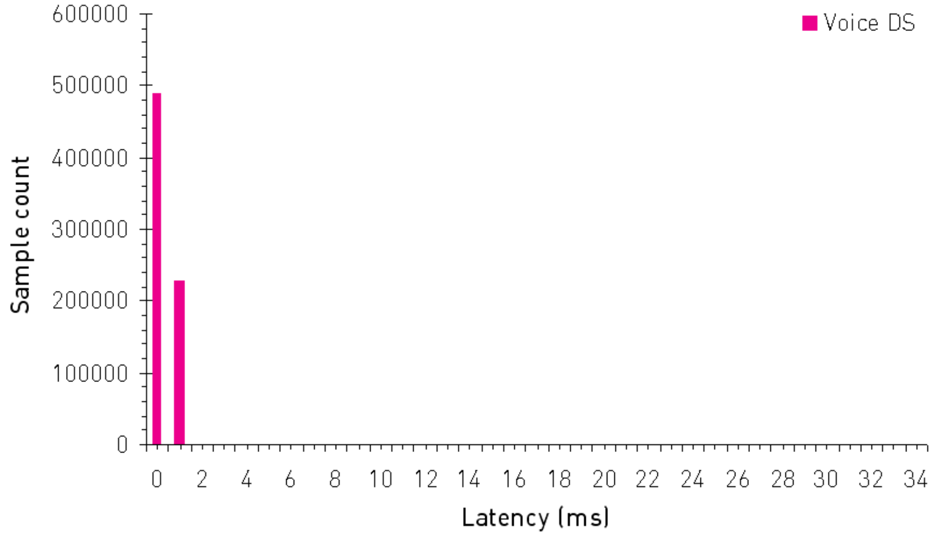

Figure 4: Downstream voice latency distribution

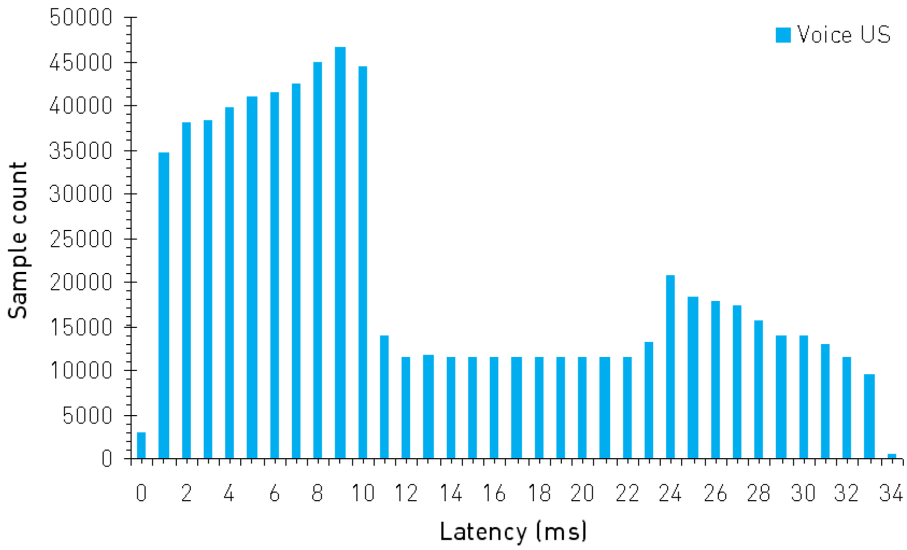

Figure 5: Upstream voice latency distribution

As indicated earlier, the Batch mode can be used to obtain measurement results over time. For example, packet loss over time may turn out to be as follows:

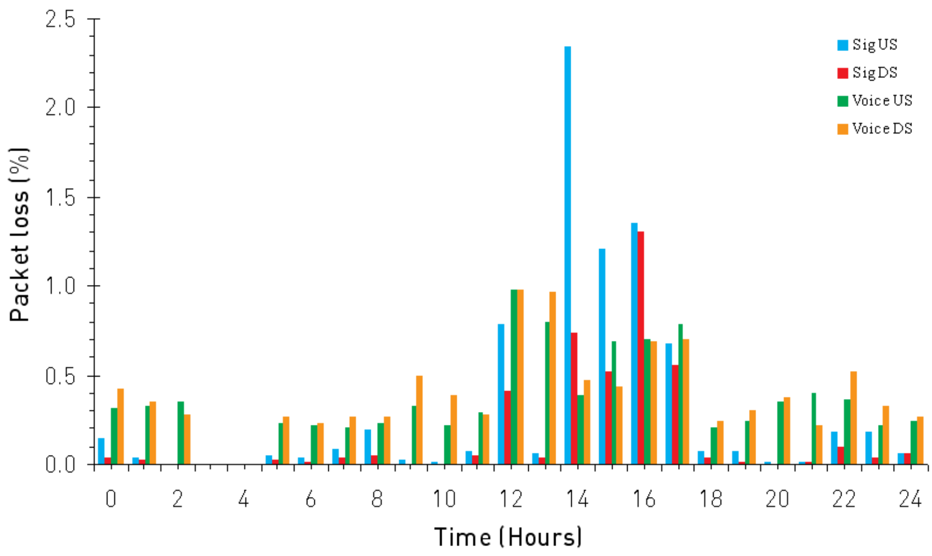

Figure 6: Packet loss over time

To determine whether your network is fit for the VoIP service, a number of criteria can be used, depending on how complex you want to make the assessment.

- In general, packet loss needs to be as low as possible, as well as latency and jitter. In case there are service level agreements that are expressed in these quantities, you could look at these figures only.
- In case there are no predefined limits on the aforementioned quantities, it is common to use the ITU's E-model [ITU-T] and related documents to try and estimate the conversational quality that is to be expected.

Conversational quality, speech quality or voice quality is generally expressed on the MOS scale (Mean Opinion Score).

MOS	Quality
5	Excellent
4	Good
3	Fair
2	Poor
1	Bad

The aforementioned E-model allows to estimate a MOS value based upon transmission parameters that can be derived from information such as the packet loss, latency, jitter and codec in use. This means that the ByteBlower measurements from the live network can basically be transformed into a MOS score prediction. As a rule of thumb, a MOS score of 4 is considered "toll quality" and is to be strived for. Of course, this calculation only applies to the measurements made on the (simulated) voice flows, and does not apply to the signalling flows.

Although ByteBlower returns quite a lot of useful information thanks to the latency distribution measurement, it is more typically the average latency (or sometimes the maximum latency) that is taken into account for MOS score calculations. In addition, in order to further simplify results, it is common to summarize all packet loss, latency and jitter figures of a certain measurement to one value for each of these quantities per link. Note that in case of a cable network, this can be an upstream or a downstream interface.

Example end results for the voice stream summarized this way are shown below. The MOS score calculations were done using a certain set of assumptions for the codec in use (taking into account its packet loss concealment capabilities and other parameters).

Link	Packet Loss (%)	Average Latency (ms)	Jitter (ms)	Estimated MOS
Node 1 downstream	1.19	3.1	0.6	4.18
Node 1 upstream	0.82	6.2	1.3	4.22
Node 2 downstream	1.56	3.7	0.7	4.14
Node 2 upstream	2.59	8.1	3.2	4.01

From this example set of results, we can conclude that expected voice quality is good, although for the upstream of node 2 the packet loss is rather high and a lower MOS is predicted. Thanks to packet loss concealment techniques, the codec can cope with it for now, but packet loss should not get any higher.

Example end results for the signalling flows are included below.

Link	Packet Loss (%)	Average Latency (ms)	Jitter (ms)
Node 1 downstream	1.51	2.3	0.57
Node 1 upstream	0.81	12.8	9.8
Node 2 downstream	1.71	2.7	0.63
Node 2 upstream	3.24	19.4	7.8

This leads us to the conclusion that our example node 2 upstream is relatively worse for conveying signalling packets, whereas voice still is transported fine. However, the observed packet loss and latency on the signalling packets would need more attention in this case.

3.6 Advanced measurement and monitoring set-ups

Although the ByteBlower GUI, with its flows, scenarios, and its batch mode, allows for quite complex set-ups already, it is sometimes desired to add some more intelligence to the measurement system. With ByteBlower, this can be done thanks to the availability of the open Application Programming Interface (API).

Thanks to this API, the most complex measurement set-ups can be programmed and automated. But there is more: because the API also gives access to measurement results, taking automated and real-time action upon certain measurements becomes possible! This means that ByteBlower can for example be used for automated, periodic monitoring of VoIP performance in a network. In case measurement results indicate a drop in performance, the Network Operations Centre can for example be notified automatically to allow immediate action to be taken.

4 Summary

This application note showed how ByteBlower can be used to test network readiness for the VoIP application. The main features that were covered, are summarized below.

- User-friendly, standard ByteBlower GUI allows easy creation of multiple types of flows per port (simulating voice and signalling traffic)
- Distributed architecture and GPS synchronization allow measurements of latency, jitter and packet loss over long distances
- Support for DHCP allows easy integration of test devices into a real-life network
- Automation of scenarios in batches allows long-running successions of short tests, in turn allowing analysis of performance over time
- Easily reconfigurable set-up allows to move the measurement points to different locations in the network, effectively narrowing down the scope of measurements to identify performance bottlenecks or track down sources of errors
- Even more complex (custom) scenarios and measurements can be automated by using ByteBlower's Application Programming Interface (API), which can even turn ByteBlower into an active performance monitoring tool

Glossary

A

Application Programming Interface (API) An application programming interface is the interface that a computer system, library or application provides in order to allow requests for services to be made of it by other computer programs, and/or to allow data to be exchanged between them.

D

DHCP (Dynamic Host Control Protocol) The Dynamic Host Control Protocol is a client-server networking protocol. The DHCP server provides configuration parameters like an IP address, default gateway, ... specific to the DHCP client host requesting information to participate on an IP network.

I

Internet Protocol (IP) The Internet Protocol is a data-oriented protocol used for sending data across a packet-switched internetwork.

M

Mean Opinion Score (MOS) The Mean Opinion Score provides a numerical indication of the perceived quality of received media after compression and/or transmission. The MOS is expressed as a single number in the range 1 to 5, where 1 is lowest perceived quality, and 5 is the highest perceived quality.

Q

Quality-of-Service (QoS) In the fields of packet-switched networks and computer networking, the traffic engineering term Quality of Service (QoS) refers to the probability of the telecommunication network meeting a given traffic contract, or in many cases is used informally to refer to the probability of a packet succeeding in passing between two points in the network within its desired latency period.

T

TCP (Transmission Control Protocol) The Transmission Control Protocol provides connections between two applications over which they can exchange data. The protocol garantees reliable and in-order delivery of sender to receiver data. The protocol number in the IP-layer for TCP is 6.

U

UDP (User Datagram Protocol) The User Datagram protocol provides a procedure for application programs to send messages to other programs with a minimum of protocol mechanism. The protocol is transaction oriented, and delivery and duplicate protection are not guaranteed. The protocol number in the IP-layer for UDP is 17 (=0x11).

References

ITU-T: ITU-T Recommendation G.107: The E-model, a computational model for use in transmission planning. ITU-T, March 2005

List of abbreviations

API	Application Programming Interface
CMS	Call Management Server
CPE	Customer-premises Equipment
DHCP	Dynamic Host Configuration Protocol
DSL	Digital Subscriber Line

E-MTA	Embedded Multimedia Terminal Adapter
GPS	Global Positioning System
GUI	Graphical User Interface
HFC	Hybrid Fibre Coax
IP	Internet Protocol
MOS	Mean Opinion Score
PCMA	Pulse Code Modulation A-law
PSTN	Public Switched Telephone Network
QoS	Quality-of-Service
RTP	Real-time Transport Protocol
TCP	Transmission Control Protocol
UDP	User Datagram Protocol
VoIP	Voice over IP

Additional information

Please contact ByteBlower@excentis.com for comments, further suggestions or additional information.

Ordering Information

ByteBlower is available in different configurations. All configurations support the use of 2 1-GigaBit interfaces for traffic of which one can be used for multiplexing traffic that is distributed by a switch. The different configurations differ in the number of ports that can be used on the switch. Current configurations are:

ByteBlower Version	# of ports	Article ID
12 Beaufort	12	BB12Bft
24 Beaufort	24	BB24Bft
48 Beaufort	48	BB48Bft