VoIP Management Based on Extended ITU-T H.341 MIB Objects for Monitoring the Speech QoS

Ana Flávia Marinho de Lima, Francisco Raphael B. Ribeiro, Wellington Albano and José Neuman de Souza

Abstract — **Monitoring calls controlled by a gatekeeper in a voice over IP (VoIP) environment enables call management itself and the related speech QoS. The acquired information can be used to evaluate the speech parameters during the calls and make some actions for achieving agreed quality levels. ITU-T has standardized H.341 MIB objects that specify information to monitor and control VoIP calls. This article proposes an agent that implements the H.341 MIB (RAS and *Gatekeeper*) as well as some extension objects for call management in a gatekeeper-based VoIP system.**

***Index Terms*—*Gatekeeper*, Management, MIB, Voice over IP, SNMP Agent.**

I. Introduction

Voice over IP (VoIP) is a technology that allows voice transmission using the Internet as media between people throughout the world. The great contribution that VoIP gives to the users is the significant cost reduction in long distance calls. Since the user's location is transparent due to the use of the Internet, the cost of a call using this technology is not calculated by the distance between the caller and called points. This way a call considered local or long distance by a telephony carrier does not have distinction with the use of VoIP.

On voice over IP networks, there are some protocols that are responsible for call signaling, such as the H.323 recommendation - ITU-T (International Telecommunication Union - Telecom Standardization Sector) in[1], SIP (Session Initiation Protocol) in [2] IETF (Internet Engineering Task Force) and the ITU-T H.248.1 recommendation in [3]. Each one contains its own architecture and set of entities that are responsible for the signaling on the network.

Managing VoIP networks is a mandatory task, mainly for being a real time transmission network. This characteristic may be taken into account when the service is offered to the users who assume that it will always be ready to use. ITU-T standardized H.341 MIB in [4] contains a set of MIBs (Management Information Bases) for managing H.323-based multimedia systems. Each MIB contains a set of objects responsible for returning relevant information related to the implemented module.

This article presents the proposal and implementation of a management agent for monitoring a gatekeeper (entity responsible for authorizing call establishment between H.323 devices) and the calls in the VoIP environment at UFC (Federal University of Ceará). To reach this objective, an information base for this agent is currently under construction, with the managed objects specified by H.341 MIB (RAS and Gatekeeper). Moreover, a group of objects that return information (time, delay, R-factor, MOS and the number of lost packets) about the call quality will be extended.

Next section mentions some works related to this article. Section III describes the involved H.323 signaling and elements in the architecture. Sections IV and V present H.341 recommendation and the objects to be extended. The management proposal for VoIP networks is presented in Section VI. Finally, section VII presents the future works and conclusions.

II. Related Works

Some works on VoIP management have been developed based on H.323 signaling protocol. [5] carried through a survey of a management environment, analyzing the possible MIBs to be implemented, including ITU-T H.341 MIB, that is one of the subjects of the present article. Also, that work described a way of operation for accounting and the proposal of a tool for VoIP network management.

A study on two tools that monitor the evaluation of quality of VoIP networks in an active and passive way was made by [6]. The tool for active measurement is already implemented and in use at UFRJ's NCE (Electronic Computer Center). The architecture of the passive measurement tool was described in details in order to support the development of the collector tool module for the managed environment presented in this work.

A tool for the measurement of call quality based on the extended E-Model was developed by [7], where the returned information about the speech quality is being used to compose a new group of objects in the H.341 MIB.

III. H.323 Signaling Protocol

The H.323 standard provides a base for data, audio and video communication through packet-switched networks (particularly, IP technology). H.323 is a suite of protocols for multimedia communication over networks that do not support Quality of Service (QoS).

The work with H.323v1 started in May 1995 and that version was approved in June 1996. Version 2 was approved in February 1998 with three annexes: H.245 message used for H.323 endpoints; procedures for layered

. Ana Flávia Marinho de Lima is a Master's student at Department of Teleinformatics, oriented by José Neuman de Souza, at Federal University of Ceará (UFC), Caixa Postal 6007 – 60.755-640 – Fortaleza – CE – Brazil (anaflavia@deti.ufc.br).

Francisco Raphael B. Ribeiro is an undergrad student at Department of Computer Science at Federal University of Ceará (UFC) - Fortaleza – CE – Brazil. (raphael@ufc.br).

Wellington Albano is technical coordinator at the National Education and Research Network's Point of Presence in Ceará, Brazil (wellington@pop-ce.rnp.br).

José Neuman de Souza is researcher and professor at Departament of Computer Science at Federal University of Ceará (UFC) – Fortaleza – CE – Brazil. (neuman@ufc.br).

video codecs and H.323 over ATM (Asynchronous Transfer Mode). Version 3 was approved in September 1999, bringing three new annexes: communication between diverse administrative domains with H.225; a new call signaling mechanism based on UDP; the specification of a H.323 subgroup that may be implemented in small devices. The work with the H.323v4 was concluded in November 2000, bringing improvements to reliability, scalability and flexibility aspects in [8].

H.323 is an “umbrella” specification that describes in a complete way the architecture and the operation of a videoconference system on a packet network. This protocol is not specific for IP: it has sections on the use of H.323 over IPX/SPX or ATM. H.323 itself only provides the conception of a system and how multimedia communication systems must work. Particularly, H.323 still has a strong influence on the systems conception. The standard defines the Registration, Admission and Status protocol (RAS) for call routing , through H.225 protocol, responsible for the necessary resources to establish calls and through H.245 protocol, responsible for the necessary mechanisms for exchanging capacity and resource negotiation.

The H.323 signaling protocol is based on the Q.931 protocol, Integrated Services Digital Network (ISDN), that allows easy operation with legacy voice networks as PSTN (Public Switched Telephone Network) or SS7 (Signaling System Nº. 7), and this can be seen as an advantage to the system, since its integration with traditional telephonic systems is simplified. However, the very complexity of the protocol imposes limits to scalability that opposes this simplicity.

The H.323 standard specifies a system composed by Terminals, Gatekeeper, Multipoint Control Units (MCU) and Gateway in [9].

IV. H.341 Recommendation

ITU-T H.341 recommendation, published in 1999 with version 3 of H.323 protocol, defined a set of MIBs for the management of multimedia systems based on H.323 and H.320 protocols, as may be seen in Table 1. The objective of the MIBs presented in Table 1 is to standardize the available management information in multimedia conference systems, presenting objects that can be used to fault, performance and configuration management.

The use of this MIB is optional. However, if it is implemented in a device, all pertinent modules must be supported. The mandatory modules must be implemented in the referenced devices in [4].

As the objective of this article is to present an agent that monitors the calls and the configuration of the gatekeeper in use, only the MIBs of the Gatekeeper module (RAS and Gatekeeper) will be presented.

- **MIB RAS**

This MIB presents information related to RAS signaling – H.225, and is organized in five functional groups: *rasConfiguration*(1), *rasRegistration*(2), *rasAdmission*(3), *rasStatus*(4) and *rasEvents*(5).

The groups of objects that compose this MIB return information related to RAS signaling, such as call configuration, registration, admission, status and some other events that might occur if an admission is rejected.

Table 1 – Application of MIB H.341 modules according to H.323 device. Source: [4].

	Terminal	Gatekeeper	MCU	Gateway
Mandatory	Callsignalling	RAS	Callsignalling	Callsignalling
	H.245	Gatekeeper	H.245	Gateway
	Terminal		MC	RTP
	RTP		RTP	
Mandatory if the module is installed	RAS	Callsignalling	RAS	RAS
	MC	H.245	MP	MC
	MP	MC		MP
		MP		
		RTP		

- **MIB Gatekeeper**

This MIB presents information related to the gatekeeper and is divided in six groups: h323GatekeeperZone(1), h323GatekeeperSystem(2), h323GatekeeperConfiguration(3), h323GatekeeperStatistic(4), h323GatekeeperControls(5) and h323GatekeeperNotifications (6).

The groups of objects that compose this MIB return information related to the gatekeeper’s configuration.

V. Speech QoS Extension

With the implementation of the gatekeeper module of H.341 MIB it is possible to better monitor the calls and the gatekeeper’s configuration. Although there is much information returned by RAS and Gatekeeper groups of objects, it is not enough for a good management of the VoIP network, mainly when the goal is to get information about the call quality, an aspect not considered by the mentioned groups of objects.

To supply this necessity, a group of five objects in MIB RAS will be created to return the information related to the call quality. These objects are: time, delay, R-factor and MOS (as later explained), number of lost packets (due to loss in the network or discards in the jitter compensation buffer). Figure 1 shows the scenario for the generation and collection of input information for the E-Model.

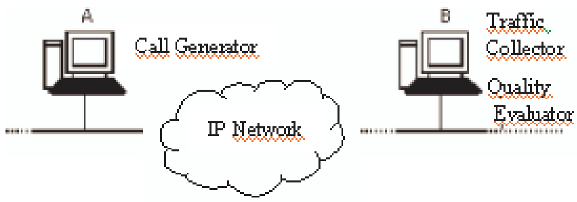

Figure 1 Traffic generation and collection of input information for E-Model. Source: [7].

- Call Generator

Located in machine A, this module artificially generates voice to be evaluated. The program OpenH323 Call Generation in [10], also known as CallGen323, allows the following functionalities: to generate an adjustable number of simultaneous calls; to execute a previously recorded voice file; to hang up a call after an established time; to wait for a new call during an adjustable time; to store a trace file containing information on the call; to select a preferred codec and to configure the number of frames per packet (voice packet duration).

CallGen323 works only with G.711 μ-law codec. In spite of restraining the possibilities of evaluation, the E-Model will not lose generality. Other open code applications may work with codecs such as G.711 A-law and GSM-FR. In a real call, however, this module is able to work with any codec, as presented in [11].

- Traffic Collector

Located in machine B, this module collects information about the received calls, such as delay, packet loss, codec type, among others. These data are the input to the evaluator module.

- Quality Evaluator

This module must be located in the same machine where the MIB is installed. Its functioning depends on the data received from the traffic collector module, which can be supplied to any machine in the form of a trace file.

The output from the call collector module generates a trace file with the following information:

- Time

Informs the moment that the packet arrived to machine B, according to its clock.

- Delay

Informs the time delay since the packet left machine A, considering the jitter compensation buffer delay and the delay for mounting and reconstructing the packet.

- R-factor

Informs E-Model's quality R-factor for the call period, between two consecutive moments.

- Number of lost packets

Informs the amount of packets that have been lost until the moment. The loss may be caused by two distinct reasons: Loss in the network, that may occur due to congestion and packet discard, and loss due to the delay of packets to arrive at the destination, since the jitter compensation buffer considers the delayed packets as lost.

One of the great advantages of this group of objects compared to RAS and Gatekeeper groups is that the objects will be able to return information on the speech QoS, independent of the signaling protocol, that might be H.323 or SIP, for instance.

Information regarding the first four objects can already be collected and observed through the call quality measurement tool developed by [7]. However, the object related to the number of lost packets is being added as a contribution to the work on the MIB's extension. The analysis on the number of lost packets due to loss in the network or in the jitter compensation buffer makes it easier for the manager to determine the causes of loss problems, so that they can be treated and possible solutions can be addressed according to the type of loss that occurred.

VI. Proposal for Management Using Extended ITU-T H.341.MIB

The management architecture for the VoIP network object of this article is composed of six functional entities, which exchange information as shown in Figure 2.

The management station is the interface between the network administrator and the management system, allowing the network to be monitored and controlled. The manager is responsible for communicating with the agent through SNMP messages in [12] and for obtaining the information available in the agent's MIB.

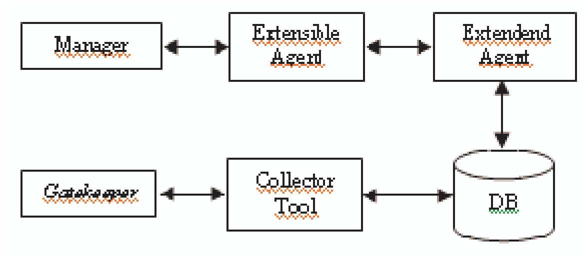

Figure 2: Management Architecture.

The agent is a piece of software executed on the device to be managed, and it may be a separate program or incorporated into the operational system. The agent provides the manager with information when requested. When something abnormal occurs on the agent side, it will be able to notify the manager on the occurrence by sending trap messages. Basically, there are two types of agents: the extensible agents and the extended agents in [13]. The extensible agents use the SNMP directly and utilize APIs (Application Program Interfaces) for the development of other agents. On the other hand, the extended agents indirectly use SNMP by way of an extensible agent and interact with it through the APIs.

The extensible agent module is the interface between the manager and the requested resources. The extensible agent used for the VoIP management architecture is NET-SNMP v5.2 in [10]. This agent is compatible with the three versions of SNMP, and has diverse tools for trap generation, access and modification of available information in SNMP agents. Besides that, it has an API that can be used for the development of sub-agents or modules to be incorporated to the agent. NET-SNMP was chosen for being an open-code, free tool, and for having tools as mib2c, that deals with the conversion of ASN.1 code to C, so that it eases the implementation of MIB objects.

The extended agent is developed as a sub-agent on which the objects of extended H.341 MIB are implemented. The extensible agent calls an application running in parallel when a management station requests data objects implemented on this sub-agent. It accesses the database generated by the collector tool and sends the data requested to the extensible agent through the NET-SNMP API.

The Database - DB is composed by MySQL software in [14] that stores the information supplied by the collector module and forwards them to the MIBs module.

The tool for data collection is being developed to process and store the necessary data in MySQL. It uses the outputs from the protocol analyzer Ethereal in [15] and from the call quality analysis tool based on the extended E-Model, described in Section V. Despite Ethereal acting as a sniffer and capturing all the packets that pass through the network, with the use of filters it is possible to select only the packets related to the H.323 stack (e.g., H.225, H.245, Q.931, RTP and RTCP packets) without compromise to the security of the internal environment.

The gatekeeper used is the GnuGK package [16] version 2.0.7, for being more stable than the most recent versions. GnuGK is a H.323 gatekeeper based on the OpenH323 library [17], that is an open-code, free implementation of the H.323 stack. It supports authentication via database, H.235 (RAS authentication), LDAP (Lightweight Directory Access Protocol) and authentication based on the IP-User mapping. The latter is the method used in the environment in question. GnuGK is the source of information for the objects that are being developed, but as it does not support SNMP, it is necessary a collector tool for obtaining the data.

According to the H.323 Recommendation [1], a gatekeeper will have to provide the following services: address translation, admission control, bandwidth control, management of the administrative zone, signaling for call control, call authorization, bandwidth and call management.

A. Implementation of the Agent

The extended agent is being developed as a sub-agent, in C language, that it is the native language for NET-SNMP and its API. The sub-agent runs together with the extensible agent.

The request for information about the objects that arrive at the extensible agent is forwarded to the sub-agent and it sends such information to the agent.

All the communication between agent and sub-agent is intermediated by the AgentX protocol in [18]. This protocol is part of the implementation of NET-SNMP's AgentX framework, which was created with the intention of being the standard framework for the development of extensible SNMP agents. As defined in [18], all the communication is made through TCP port 705.

The code's basic structure, created from the ASN.1 definition for RAS and Gatekeeper MIBs, was generated through the tool mib2c that it is supplied together with the NET-SNMP package. The access to the database occurs through calls to the native MySQL access library, and the data are processed by data structures generated by mib2c. The agent and the sub-agent are in the same machine as the gatekeeper - GnuGK.

B. Implementation of the Collector Tool

This module is being developed in PHP. The output data from tEthereal, text mode of protocol analyzer Ethereal, and from the call quality analysis tool are passed to the collector tool by a pipe in order to extract relevant data for MIBs and their extensions and are stored in MySQL using PHP native communication interface. The collector tool is situated in the same machine as the other management elements.

The packets captured by tEthereal are filtered and only H.225, H.245, Q.931, RTP and RTCP packets are returned. Collected data are stored in PDML (Packet Details Markup Language), an XML-based format for visualization of data details in [15].

To test the other parts of the management system, a simple test collector tool was created, which collects just some information present at admission request messages of H.225 packets: call type (point to point, direct or routed by gatekeeper), destination alias, H.323 ID of source, call reference value and call ID. (This information is present on the following fields of admission request message, respectively: CallType, DestinationInfo, srcInfo, CallReferenceValue and CallIdentifier). In PDML format, this information is identified in the 'show' attribute of elements (tags) called 'field'.

The tool has five parts. The first one is tEthereal itself, which is used to capture the packets. Used with the parameters "- T h225", it collects only H.225 RAS packets and with "- R pdml", it formats the output in PDML.

The second part, the separator, is part of the PHP script. It opens a pipe to receive the tEthereal output at real time and separates it by packets. The third part is the parser. In the parser implementation, it was used expat, php's extension to analyze XML documents. The parser receives the packet from the separator and analyzes it for identification, and in case that it is identified as a call admission request, the parser takes the desired information from it. This analysis could not have directly been done on the output of tEthereal, since in order to analyze an XML document it is necessary to pass it entirely to the parser, justifying the need for the separator. The fourth part of the tool is the translator, run only when information is collected and responsible to translate it according to the MIB description.

The fifth part is the MySQL database, which is updated with the translated information to be accessed by the extended agent.

The tool was executed in the same machine where the Gatekeeper runs. Tests have been done successfully, through all the stages of the process as expected. The desired information is captured at real time and updates the database.

VII. Conclusion And Future Work

This article presents a proposal for the implementation of an extensible NET-SNMP agent based on ITU-T H.341 MIB (RAS and Gatekeeper). This agent monitors the gatekeeper and the calls managed by it.

The MIB RAS extension is also part of this work. As presented in Section V, the extended objects provide information about call quality based on the E-Model.

The agent implementation is based on the SNMP management model, which contains the entities: manager, agents and managed objects. The agents are divided in extended and extensible agents, since the gatekeeper does not support SNMP, and therefore the inclusion of the extensible agent (compatible with SNMP) is necessary. This agent is the interface between the manager and the extended agent (containing the Management Information Base - MIB). The extended agent gets information from the data stored at the MySQL database. The obtained data, related to H.323 signaling, was collected and filtered by Ethereal.

As future work the other modules of H.341 MIB may be implemented, thus allowing a better monitoring and control of the H.323 entities. Another interesting work would be to apply management to address access control security in H.323 entities. Also, it is a suggestion the creation of a tool for pro-active management that will notify the network manager about problems to the good functioning of the network, since in a VoIP environment it is very important that the network is always available.

Acknowledgment

The authors would like to thank RNP's Point of Presence in Ceará for the infrastructure and technical support. Specially, thanks to Camilo Camurça, and also to the support given by Regeane Aguiar (UFAM), João Carlos Peixoto and Fábio David (UFRJ).

REFERENCES

[1] ITU-T Recommendation H.323, "Packet-based Multimedia Communications Systems", 1999.
[2] RFC 2543. "SIP: Session Initiation Protocol". Available: http://www.ietf.org/rfc/rfc2543.txt?number=2543
[3] ITU-T Recommendation H.248.1, "Gateway Control Protocol", revised, 2002.
[4] ITU-T Recommendation H.341, "Multimedia Management Information Base", 1999.
[5] J. C. P. de A. Costa, "Implementação e Gerência de uma Arquitetura de Voz sobre IP", UFRJ, Rio de Janeiro, RJ: Master Thesis, Sep. 2003.
[6] F. David, "Ferramenta de Monitoração Ativa e Passiva para Avaliação da Qualidade de Redes VoIP", UFRJ, Rio de Janeiro, RJ: Master Thesis, Sep. 2003.
[7] L. S. G. Carvalho, "Implementação do Modelo E para Avaliação Objetiva da Qualidade da Fala em Redes de Comunicação VoIP", UFAM, Manaus, AM: Master Thesis, Sep. 2004.
[8] O. Hersent, D. Guide and J. P. Petit, "Telefonia IP – Comunicação Multimídia Baseada em Pacotes", São Paulo, SP: Addison-Wesley, 2002, pp. 4.
[9] R. Balbinot, J. G. Silveira, F. C. Castello, P. M. Dos Santos and A. da S. Quadra, "Voz sobre IP - Tecnologia e Tendências", in Proc. SBRC, 2003, pp. 321-363.
[10] Net-SNMP. Available: http://www.net-snmp.org
[11] ITU-T Recommendation G.113 Appendix I, "Provisional Planning Values for the Equipment Impairment Factor Ie and Packet-loss Robustness Factor", 2002.
[12] W. Stallings, "SNMP, SNMPv2, SNMPv3 and RMON1 and 2.", 3rd ed., Upper Saddle River, NJ: Addison-Wesley, 1999, pp. 71-83.
[13] D. R. Mauro and K. J. Schmidt, "SNMP Essencial", Rio de Janeiro, RJ: Campus, 2001, pp. 179-201.
[14] MySQL. Available: http://mysql.org
[15] Ethereal. Available: http://www.ethereal.com
[16] GNUGK – The GNU Gatekeeper. Available: http://gnugk.org
[17] OpenH323 Project. Available: http://www.openh323.org
[18] RFC 2741. "Agent Extensibility (AgentX) Protocol". Available: http://www.ietf.org/rfc/rfc2741.txt?number=2741