Remote Analysis of Human Voice – Lossless Sound Recording Redirection

Tomáš Kulhánek [1,š], Marek Frič [2], Milan Šárek [1]
[1] CESNET z.s.p.o., [2] Music and Dance Faculty of the Academy of Performing Arts, [3] First Faculty of Medicine, Charles University
tomaton@centrum.cz

Abstract. This paper introduces a new approach to lossless sound recording redirection over remote desktop protocol. This approach is also demonstrated on an application in the field of Analysis of Human Voice used by phoniatric examination developed in the Musical Acoustics Research Centre of the Music and Dance Faculty of the Academy of Performing Arts. There was developed plug-ins for Remote Desktop Protocol which adds functionality to redirect sound recorded on the client side to the remote application without loss of information. Using these plug-ins the analytic application can be deployed on the remote server and accessed via RDP clients from different locations connected to the Internet.

1 Introduction

The human voice can be analyzed with computation methods during the phoniatric examination and a result of such analysis is produced into a voice range profile, also called as a phonetogram. It shows dynamic range of the human voice in terms of both fundamental frequency and intensity. The analytic application introduced in this paper ParVRP is being developed in the Musical Acoustics Research Centre of the Music and Dance Faculty of the Academy of Performing Arts. It implements computation methods in MATLAB environment and analyses the human voice stored in a sound file or recorded with a local microphone attached to the computer. The ParVRP application allows segmentation of the recording into separated voice events and produces phonetogram. [1]

There is being prepared the complex system which should allow usage of the ParVRP application from different location. This system is planned to support phoniatric examination in more physical locations is being prepared. As a pilot project, the analytic application is deployed into a virtual server which is part of the virtual infrastructure built in the past for another project exchanging medical images using data grid techniques [2]. The access to the analytic application is provided with remote desktop protocol (RDP) following the concept of thin client. Among other features, sound recording redirection from thin client to the remote application was introduced since RDP version 7.0 (server side since Windows Server 2008 R2 and client side since Windows XP SP3). The older version of RDP protocol 5.2 (since Windows Server 2003) needs external modification by third party product. Anyway, when testing the sound recording redirection in both versions, it was founded that the used codecs in both versions degraded the sound characteristics when transferred to remote application and thus this type of sound recording redirection is not acceptable for exact voice analysis [3].

2 Methods

There were discussed an option to use dedicated independent TCP/IP connection between client and server to redirect sound recording over it. This will bring some more configuration effort for the client and remote Internet providers, which may need to configure other communication channel behind the institutional firewall. Thus it was decided to reuse the existing RDP connection and customize virtual channels over RDP to transfer binary data.

The whole schema of this solution is shown on Figure 1. The system consists mainly with RDP plug-ins for client and server side. The main part of these plug-ins were developed in C# using basic .NET libraries. Together with the Mono project, these plug-ins can be deployed on the most used platforms which are Microsoft Windows and Linux-like operating systems [7]. The minor part of the plug-ins were developed with dependency on platform specific libraries for Microsoft Windows as well as for Linux operating systems.

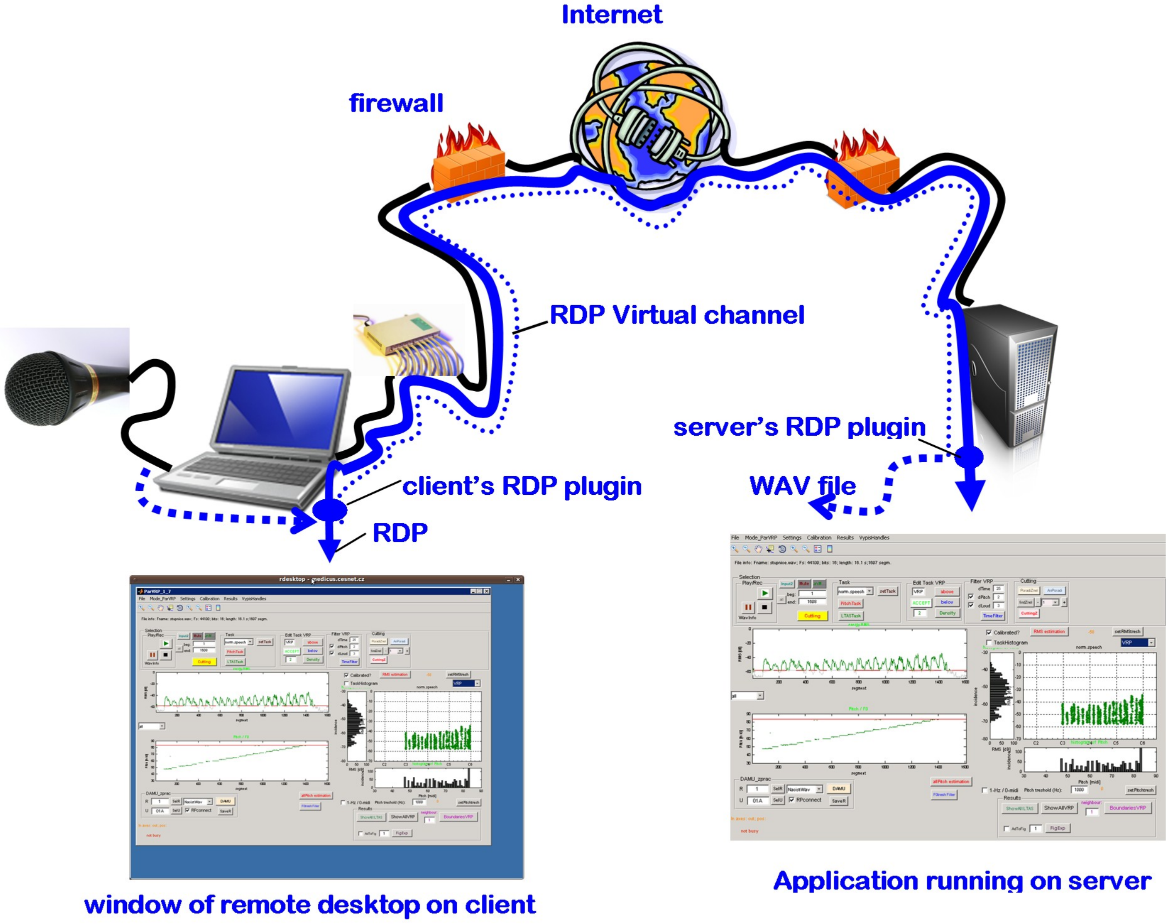
Internet
firewall
RDP Virtual channel
server's RDP plugin
client's RDP plugin
WAV file
RDP
rdesktop - medicus.cesnet.cz
window of remote desktop on client
Application running on server

Fig 1. Schema of the sound recording redirection with the client's and server's RDP plug-in

On the remote side, the server's plug-in resides on MS Windows platform and opens a named virtual channel to the RDP connection using Remote Desktop API (Wtsapi32.dll). The plug-in then send messages via this virtual channel to control recording on the client. The received data from the virtual channel is written to a file. A proprietary API is provided for the ParVRP application which controls the recording.

On the local side, the client's plug-in to RDP receives message, starts and stops the sound recording from the system default sound-card input and sends binary data of the recorded sound via virtual channel. In case of Linux (or Unix-like) operating systems, the client's plug-in uses an external patch for the rdesktop [5] which is usually part of a common Linux distributions. It adds an option for rdesktop to redirect named RDP channel communication to standard input and output of a external executable plug-in [4]. The client's RDP plug-in listens standard input and utilize another command-line application arecord [6] also available in a common Linux distributions. Arecord records sound from system default sound card input and returns binary data in WAV format. The plug-in streams the binary data into the standard output which is then redirected by the patched rdesktop to RDP virtual channel.

In case of remote desktop client within Microsoft Windows operating system, the Remote Desktop API (Wtsapi32.dll) is used to register a client's plug-in for receiving events regarding the named virtual channel. However the plug-in in .NET is a so-called managed code which however needs to be called from an unmanaged system API, thus e.g. Selvin proposes automated solution which decompiles managed code from .NET DLL and after modifications it compiles it back into DLL which is then usable from unmanaged API calls[8]. The The WINMM multimedia API (winmm.dll) is used to record sound from the system default sound card input into data stream which is then written to the RDP virtual channel.

3 Results

Using the client's and server's RDP plug-in, there can be transferred audio data via the named RDP virtual channel without any loss of information, which is the main lack of the existing solution of sound recording redirections available for RDP version 5.2 and RDP version 7.0. The plug-ins redirect the binary data in a stream. However the binary stream is transferred in uncompressed WAV format with 44.1 kHz sampling rate and 16 bits per sample thus the bitrate of such recording is 705,6 kbit/s which needs to be transferred via the network. The overhead of base RDP protocol, which uploads events from client's keyboard and mouse to remote application is minimal. This bitrate doesn't cause any problems on the server deployed on the CESNET2 network, where the network bandwidth is above 1 Gbit/s. The bandwidth on the client's side location might be limitation if it is less than 1Mbit/s.

In contrast, the average bitrate of sound recording redirection of RDP 7.0 or Sound over RDP in RDP 5.2 is only 80 kbit/s thus this solution are not demanding on the network bandwith on client's side.

Protocol	**Upload to remote application**	**Type of sound transfer**
RDP v 5.2 + Sound Over RDP	80 kbit/s	Lossy
RDP v 7	80 kbit/s	Lossy
RDP v 5.2 + plugins for WAV via virt.channel	705,6 kbit/s	Lossless

Tab 1. Comparison of upload rate and type of sound recording redirection

The RDP plug-ins are distributed as DLL libraries which can be integrated into server's Windows platform. Proprietary .NET API is provided for ParVRP application developed in MATLAB. On the client side it can be integrated into general RDP clients on both Linux and Windows platforms.

4 Conclusions

In contrast with sound recording redirection of RDP 7.0 or Sound over RDP in RDP 5.2, the introduced redirection of sound recording over RDP doesn't provide a virtual sound card which may be used by a general remote application. This breaks a so-called “transparency” and any application like ParVRP needs modification to use the proprietary API provided by the RDP plug-ins.

Anyway the introduced sound recording redirection is acceptable for planned deployment in production environment to support phoniatric examination and provide high quality on-line analysis with a planned possibility of remote corroboration among more specialists.

The introduced approach of lossless sound recording redirecting over RDP to the remote application is also 10 times demanding for transfer upload rate than the mentioned options using lossy codecs, however still far bellow the bandwidth provided by CESNET2 network, which will generally connect the collaborating workplaces.

Comparing to the mentioned dedicated TCP/IP channel, which can be used to transfer sound recording instead of the introduced virtual channels in RDP, there would be needed to solve authentication and encryption of the communication together with additional effort needed to configure institutional firewalls to allow this channel. RDP protocol provides already authentication and encryption mechanism thus the RDP plug-ins don't need to solve that issues.

Acknowledgement

This work has been supported by the grant projects 1M0531 Musical acoustics research centre and by the research plan Optical National Research Network and its New Application, in association CESNET z.s.p.o.

References

[1] Frič M., Parametrizovaný fonetogram obecných řečových a hlasových projevů –ParVRP, MARC-Technologický list čís. 12, Zvukové studio HAMU, Praha 2007, http://web.hamu.cz/zvuk/vyzkum/dokumenty/TL12x.pdf

[2] Kulhánek, T. and Šárek, M. 2009. Processing of medical images in virtual distributed environment. In Proceedings of the 2009 EATIS'09. ACM, New York, NY, 1-3. DOI= http://doi.acm.org/10.1145/1551722.1551732

[3] Kulhánek T., Frič M., Šárek M.: Remote Access to Virtual Education and Research Application, MEFANET 2009 proceeding, Brno, ISBN: 978-80-7392-118-7, http://www.mefanet.cz/res/file/mefanet2009/prispevky/kulhanek_full.pdf

[4] Yakimenko S.,"Writing plugins for Rdesktop", The Code Project, Apriorit Inc, 1.12.2009, web, http://www.codeproject.com/KB/cross-platform/writing-plugins-forrd.aspx

[5] rdesktop: A Remote Desktop Protocol Client for accessing Windows Terminal Services, web: http://www.rdesktop.org

[6] Arecord, command line wav recorder, web: http://alsa.opensrc.org/index.php/Arecord

[7] mono, cross platform, open source .NET development framework, web: http://www.mono-project.com/Main_Page

[8] Selvin, How to Write a Terminal Services Add-in in Pure C#, The Code Project, 14.11.2006, web: http://www.codeproject.com/KB/system/TSAddinInCS.aspx