Medooze MCU Video Multiconference Server
Installation and configuration guide
Ubuntu 12.04 LTS

Written by Adham Zeidan

01.09.2013

Index

1 Software tools installation

1.1 Installing Wireshark

To install the latest stable release (1.10.0) on a Windows machine, open a web browser and navigate to the website http://www.wireshark.org/download.html, and then click on the desired Windows stable release (32-bit or 64-bit). After the Windows installer has been completely downloaded, double click it and follow the steps of the installation wizard. To install Wireshark on Ubuntu follow these steps:

1. First some dependency packages have to be fetched to ensure a successful source-based installation of Wireshark. To do that, run the following commands from a terminal:

   ```
   sudo apt-get update
   ```

   ```
   sudo apt-get install autoconf bison flex libtool libgtk2.0-dev libpcap-dev libc-ares-dev libsmi2-dev libgnutls-dev libgcrypt11-dev libkrb5-dev libcap2-bin libgeoip-dev libortp-dev libportaudio-dev
   ```

2. Open a web browser and navigate to the website http://www.wireshark.org/download.html and download the source code of the stable release 1.10.0. Another possibility is to run the following command from a terminal:

   ```
   sudo wget http://wiresharkdownloads.riverbed.com/wireshark/src/wireshark-1.10.0.tar.bz2
   ```

3. The next step is to change to the directory where the Wireshark compressed tarball file is saved. After that unpack the compressed tarball file by running the following command:

   ```
   sudo tar xf wireshark-1.10.0.tar.bz2
   ```

4. Change to the directory wireshark-1.10.0 and then run the autogen.sh script to configure the build directory

   ```
   cd usr/local/src/wireshark-1.10.0
   ```

   ```
   sudo sh ./autogen.sh
   ```

5. Run the configure script to check the dependencies on your Ubuntu system

   ```
   sudo ./configure --prefix=/usr --enable-setcap-install
   ```

6. Run the make command to compile the source code and to create the executable files

   ```
   sudo make
   ```

7. After the compilation is done, install Wireshark

   ```
   sudo make install
   ```

8. To run Wireshark run one of the following command:

   ```
   wireshark &
   ```

   ```
   sudo wireshark &
   ```

1.2 Installing Java Development Kit (JDK)

To install Java SE 6 update 45 on 64-bit Ubuntu machine using a self-extracting binary file, follow these steps:

1. Download the JDK installation file (jdk-6u45-linux-x64.bin) from Oracle website (http://www.oracle.com). You need first to accept the license agreement from Oracle before you can download it

2. Copy the downloaded binary file to the location where you would like the files to be installed (for example /usr/local/src), and after that run the command

```
sudo chmod a+x jdk-6u45-linux-x64.bin
```

3. Run the self-extracting file by executing the command

```
sudo ./jdk-6u45-linux-x64.bin
```

4. After the file has been completely extracted, the directory jdk1.6.0_45 is created. The next step is to define the JAVA_HOME variable, which tells the OS where the JDK files are located. Additionally the path to Java executable files has to be set. To do that, edit the global environment file by run the following command from a terminal:

```
sudo gedit /etc/environment
```

After that add a line that defines the location of the JDK (JAVA_HOME) and then extend the PATH variable by adding JAVA_HOME/bin (see Figure 1)

```
environment
PATH="/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/src/jdk1.6.0_45/bin"

JAVA_HOME="/usr/local/src/jdk1.6.0_45"
```

Figure 1

5. Finally save the changes and reboot your machine. To make sure that the JDK has been successfully installed and the correct path to Java executable files has been correctly set, run the command **java -version**, which shows the installed and used version of Java on your Operating System (see Figure 2)

```
adham@adham-VirtualBox:~$ java -version
java version "1.6.0_45"
Java(TM) SE Runtime Environment (build 1.6.0_45-b06)
Java HotSpot(TM) Client VM (build 20.45-b01, mixed mode, sharing)
adham@adham-VirtualBox:~$
```

Figure 2

6. In case you have other Java versions installed on your Ubuntu machine, then you have to define Oracle JDK as the standard Java version to make sure that it will be used for running and developing Java applications. To do that, run the following commands from a terminal:

```
sudo update-alternatives --install "/usr/bin/java" "java"
"/usr/local/src/jdk1.6.0_45/jre/bin/java" 1

sudo update-alternatives --install "/usr/bin/javac" "javac"
"/usr/local/src/jdk1.6.0_45/bin/javac" 1

sudo update-alternatives --install "/usr/bin/javah" "javah"
"/usr/local/src/jdk1.6.0_45/bin/javah" 1

sudo update-alternatives --install "/usr/bin/jar" "jar"
"/usr/local/src/jdk1.6.0_45/bin/jar" 1

sudo update-alternatives –-set java /usr/local/src/jdk1.6.0_45/jre/bin/java

sudo update-alternatives --set javac /usr/local/src/jdk1.6.0_45/bin/javac

sudo update-alternatives --set javah /usr/local/src/jdk1.6.0_45/bin/javah

sudo update-alternatives --set jar  /usr/local/src/jdk1.6.0_45/bin/jar
```

1.3 Installing NetBeans IDE (Integrated Development Environment)

To install NetBeans IDE 7.3.1 on Ubuntu, follow these steps:

1. Navigate to the website https://netbeans.org/downloads/index.html
2. Download NetBeans IDE 7.3.1 with Java EE bundles

3. Change the directory to the location of the downloaded NetBeans installer, and then run the command

   ```
   sudo sh netbeans-7.3.1-javaee-linux.sh
   ```

   This command will extract the installation files and starts the installation wizard

4. Follow the steps provided by the installation wizard. Here you have to accept the terms in the licence agreement. After that choose the directory, in which NetBeans should be installed (default is /usr/local/netbeans-7.3.1) and make sure that the right Java JDK is used for NetBeans (see Figure 3). Finally click on install

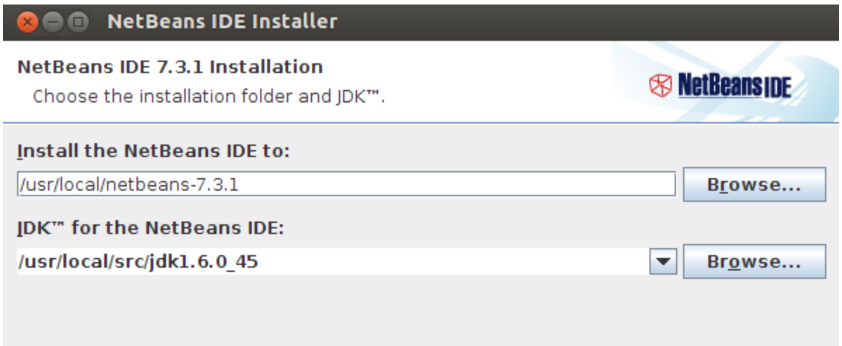

Figure 3

5. To start NetBeans, run the commands

   ```
   cd /usr/local/netbeans-7.3.1/bin/
   sudo ./netbeans &
   ```

2 Videoconferencing system implementation

This section represents a step by step installation and configuration guide of the Medooze MCU videoconferencing system on Ubuntu 12.04 LTS. The first step is to install the Media Mixer Sever and the mcuWeb application. After that GlassFish, JBoss and Tomcat Application Servers will be downloaded, installed and configured to deploy the mcuWeb application.

2.1 Installing Medooze Media Mixer Server

The Media Mixer Server is responsible for providing all media handling functionalities such as audio, video and text encoding and decoding, media mixing, web broadcasting and finally recording and playback of files. The first step is to download the source code of the videoconferencing project from the hosting website http://sourceforge.net/projects/mcumediaserver

```
cd /usr/local/src
sudo apt-get install subversion
sudo svn checkout svn://svn.code.sf.net/p/mcumediaserver/code/trunk medooze
```

Before installing the Media Server, some additional software and libraries are needed. The external software and libraries required are:

- SRTP development libraries (libsrtp0-dev)
- SSL (Secure Sockets Layer) development libraries (libssl-dev)
- GSM development libraries (libgsm1-dev)
- Yasm assembler
- x264 encoder library
- FFmpeg

- mp4v2 library and tools
- XML-RPC for C (Xmlrpc-c) library
- VP8 encoder library
- Speex codec library
- Opus codec library
- Google WebRTC source code to enable VAD support

The following is an explanation how to install the above mentioned software and libraries. First some dependency packages and libraries have to be fetched and installed including the SRTP, SSL and GSM development libraries

```
sudo apt-get install libgsm1-dev g++ make libtool yasm git automake libsrtp0-dev
libssl-dev ant libnss3-dev autoconf
```

a) Installing Yasm assembler

To download, configure, compile and install Yasm assembler version 1.2.0, run the following commands in the source directory:

```
sudo wget http://www.tortall.net/projects/yasm/releases/yasm-1.2.0.tar.gz

sudo tar xvzf yasm-1.2.0.tar.gz

cd yasm-1.2.0

sudo ./configure --prefix=/usr

sudo make

sudo make install
```

b) Installing x264 encoder library

To configure, compile and install the latest version of x264 codec library, run the following commands in the source directory:

```
sudo git clone git://git.videolan.org/x264.git

cd /usr/local/src/x264

sudo ./configure --prefix=/usr --enable-debug --enable-shared --enable-pic

sudo make

sudo make install
```

c) Installing FFmpeg

To configure, compile and install the latest version of FFmpeg and its software libraries, run the following commands in the source directory:

```
sudo git clone git://git.videolan.org/ffmpeg.git

cd /usr/local/src/ffmpeg

sudo git remote set-url origin git://source.ffmpeg.org/ffmpeg

sudo ./configure --prefix=/usr --enable-zlib --disable-stripping --enable-nonfree
--enable-gpl --enable-shared --enable-pic --enable-avresample --enable-decoder=png

sudo make

sudo make install
```

d) Installing MP4v2 library

To configure, compile and install the latest version of MP4v2 library and the corresponding command-line tools, run the following commands in the source directory:

```
sudo svn checkout http://mp4v2.googlecode.com/svn/trunk/mp4v2

cd /usr/local/src/mp4v2

sudo autoreconf -fiv

sudo ./configure --prefix=/usr

sudo make

sudo make install

sudo make install-man
```

e) Installing Xmlrpc-c library

To download, configure, compile and install Xmlrpc-c library version 1.16.44, run the following commands in the source directory:

```
sudo wget http://downloads.sourceforge.net/project/xmlrpc-c/xmlrpc-c%20Super%20Stable/1.16.44/xmlrpc-c-1.16.44.tgz

sudo tar xvzf xmlrpc-c-1.16.44.tgz

sudo su

cd /usr/local/src/xmlrpc-c-1.16.44

./configure --prefix=/usr

make

make install

exit
```

f) Installing VP8 encoder library

To configure, compile and install the latest version of VP8 codec, run the following commands in the source directory:

```
sudo git clone http://git.chromium.org/webm/libvpx.git

cd /usr/local/src/libvpx

sudo ./configure --prefix=/usr --enable-shared

sudo make

sudo make install
```

g) Installing Speex library

To configure, compile and install Speex codec version 1.2, run the following script in the source directory:

```
sudo wget http://downloads.xiph.org/releases/speex/speex-1.2rc1.tar.gz

sudo tar xvzf speex-1.2rc1.tar.gz

cd /usr/local/src/speex-1.2rc1

sudo ./configure --prefix=/usr

sudo make

sudo make install
```

h) Installing Opus library

To configure, compile and install Opus codec version 1.0.2, run the following script in the source directory:

```
sudo wget http://downloads.xiph.org/releases/opus/opus-1.0.2.tar.gz

sudo tar xvzf opus-1.0.2.tar.gz

cd /usr/local/src/opus-1.0.2

sudo ./configure --prefix=/usr

sudo make

sudo make install
```

i) Enabling VAD functionality

To enable the VAD based positioning functionality, which is used to detect the presence or absence of voice, some libraries from the Google WebRTC implementation are needed. Therefore the first step is to download and install the Google depot_tools, which is a package of scripts used to manage checkouts and code reviews.

```
sudo su

cd /usr/local/src/

svn co http://src.chromium.org/chrome/trunk/tools/depot_tools

export PATH="$PATH":/usr/local/src/depot_tools
```

After that download the Google WebRTC source code

```
mkdir webrtc

cd webrtc

gclient config http://webrtc.googlecode.com/svn/trunk

gclient sync --force

gclient runhooks --force

cd trunk
```

Finally build the required libraries by compiling WebRTC VAD and the corresponding signal processing libraries

```
ninja -C out/Release/ common_audio

exit
```

j) Compiling and running the MCU Media Mixer Server

Finally after installing all previous software and libraries mentioned in this section, the Media Server can be compiled and installed. To do that, navigate to the media mixer source directory /usr/local/src/medooze/mcu and edit the configuration file “config.mk” (see Figure 4). To enable or disable a specific configuration, change the value of the corresponding parameter by typing “yes” or “no”. For example to enable debugging modus, set the value of DEBUG to yes.

```
sudo gedit /usr/local/src/medooze/mcu/config.mk
```

Figure 4

Finally save the changes and then compile and install the MCU Media Mixer Server

```
sudo su
cd /usr/local/src/medooze/mcu
make
make install
ldconfig
exit
```

To run the Media Mixer Server, navigate to the installation directory

```
cd /usr/local/src/medooze/mcu/bin/release
```

And after that run the command

```
sudo ./mcu [-h] [--help] [--mcu-log logfile] [--mcu-pid pidfile]    [--http-port port] [--rtmp-port port]
```

Table 1 explains the functionality of the different parameters that can be set for the Media Mixer Server.

Table 1

Parameter	**Function**
`-h,--help`	Print help
`-f`	Run as daemon in safe mode
`--mcu-log`	Set MCU log file path (default: mcu.log)
`--mcu-pid`	Set MCU pid file path (default: mcu.pid)
`--http-port`	Set HTTP XML-RPC API port
`--rtmp-port`	Set RTMP XML-RPC API port

For example to run the Media Mixer Server using port 9090 for HTTP XML-RPC API, run the following command

```
sudo ./mcu --http-port 9090
```

2.2 Installing Medooze mcuWeb application

mcuWeb is a web application that is responsible for SIP signalling, commanding the Media Server through the XML-RPC interface and providing an administration web interface for managing the service. The binary release of the mcuWeb application can be downloaded directly from the hosting website under http://sourceforge.net/projects/mcumediaserver/files/mcumediaserver/.

To compile the mcuWeb application from source, some additional libraries and software are needed. First you need to download the Apache XML-RPC library, which is a Java implementation of XML-RPC. To download and extract the library files of Apache XML-RPC release 3.1.3, which is compliant to the XML-RPC specification, run the following commands:

```
cd /usr/local/src/
```

```
sudo wget http://www.apache.org/dist/ws/xmlrpc/binaries/apache-xmlrpc-3.1.3-bin.tar.bz2
```

```
sudo tar -jxf apache-xmlrpc-3.1.3-bin.tar.bz2
```

The next step is to edit the project properties to match the Apache XML-RPC library directory. Therefore navigate to the XML-RPC client project directory and then edit the file “project.properties” (see Figure 5).

```
cd /usr/local/src/medooze/XmlRpcMcuClient/nbproject
```

```
sudo gedit project.properties
```

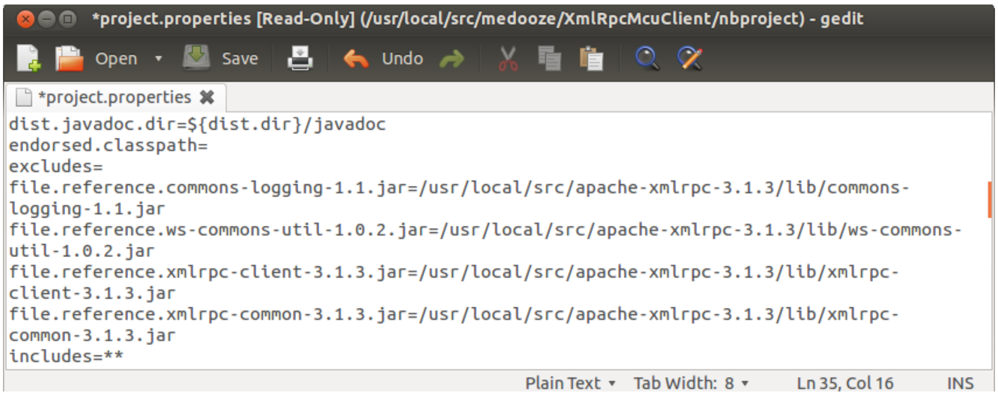

Figure 5

After that save the changes and compile the XML-RPC client project

```
cd /usr/local/src/medooze/XmlRpcMcuClient
```

```
sudo ant
```

The second important component needed to compile the mcuWeb application is the SailFin project, which is based on robust and scalable SIP Servlets technology on top of a Java EE-based GlassFish Application Server.

To download and install Sailfin version 2, which is the final build based on GlassFish AS version 2.1.1 and SIP Servlet API 1.1, run the following commands:

```
cd /usr/local/
```

```
sudo wget
http://download.java.net/javaee5/sailfin/v2_branch/promoted/Linux/sailfin-installer-v2-b31g-linux.jar
```

```
sudo java -Xmx256m -jar sailfin-installer-v2-b31g-linux.jar
```

After the installation of SailFin project, a new domain has to be created on the GlassFish Application Server. To do that, open and edit the installation file “setup.xml”. Important configuration parameters are the domain name, administrator username and password and HTTP port numbers for administrator and other users (see Figure 6).

```
cd sailfin
```

```
sudo gedit setup.xml
```

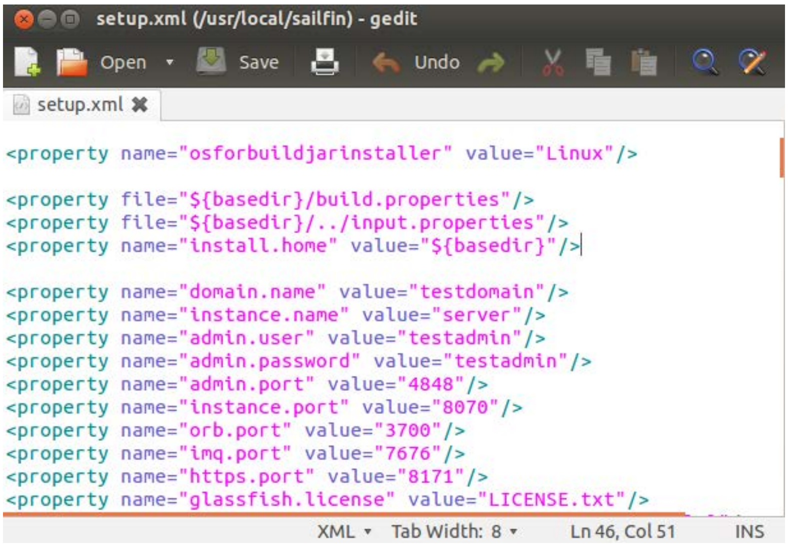

Figure 6

Finally save the changes and compile the file "setup.xml" to create the new domain

```
sudo chmod -R +x lib/ant/bin
```

```
sudo lib/ant/bin/ant -f setup.xml
```

Now it is time to compile the mcuWeb application. First navigate to the mcuWeb project directory, and then edit the file "project.properties" to match the Apache XML-RPC library and SailFin project directories (see Figure 7).

```
cd /usr/local/src/medooze/mcuWeb/nbproject
```

```
sudo gedit project.properties
```

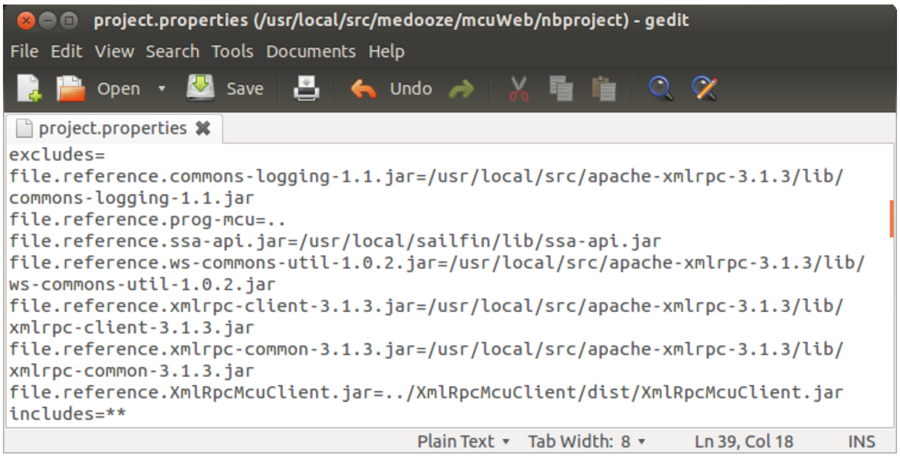

Figure 7

Finally compile the mcuWeb project

```
cd /usr/local/src/medooze/mcuWeb
```

```
sudo ant -Dj2ee.server.home=/usr/local/sailfin -Dlibs.CopyLibs.classpath=/usr/local/netbeans-7.3.1/java/ant/extra/org-netbeans-modules-java-j2seproject-copylibstask.jar
```

2.3 Deploying mcuWeb in GlassFish Application Server

Deployment is the process, by which a finished application or component is distributed to be installed on other computers, systems or servers. To deploy the mcuWeb application in GlassFish server, do the following steps:

1. first the created domain (testdomain) from last section (see section 2.2) has to be started

```
cd /usr/local/sailfin/bin/

sudo ./asadmin start-domain testdomain
```

2. After that login to the admin web interface of GlassFish AS by opening a web browser on Ubuntu and visiting the link http://localhost:4848, where 4848 is the administrator HTTP port number specified by the creation of the domain. Additionally you can login to the admin web interface using a web browser on other computers connected to the same network by visiting the link http://ipaddress:4848, where ipaddress is the IP address of Ubuntu. Enter the administrator username and password and then click on Login (see Figure 8)

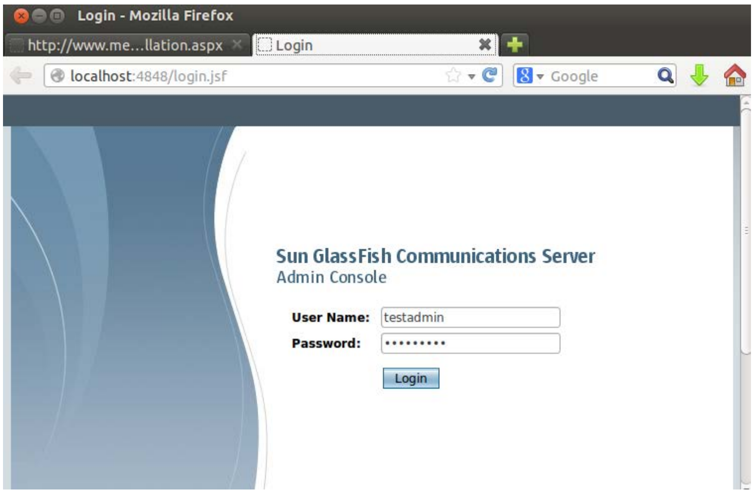

Figure 8

3. The next step is to change the listening port numbers of the SIP protocol. The default port numbers used for SIP are 5060 and 5061. But since the mcuWeb application will be deployed on several Application Servers running on the same machine, it is important that each AS is using different port numbers for the SIP signalling. To change the default listening port numbers, click on Configuration ➔ SIP Service ➔ sip-listener-1 and change the listening port to 5070 (see Figure 9). After that save the changes and repeat these steps to change the listening port on sip-listener-2 to 5071

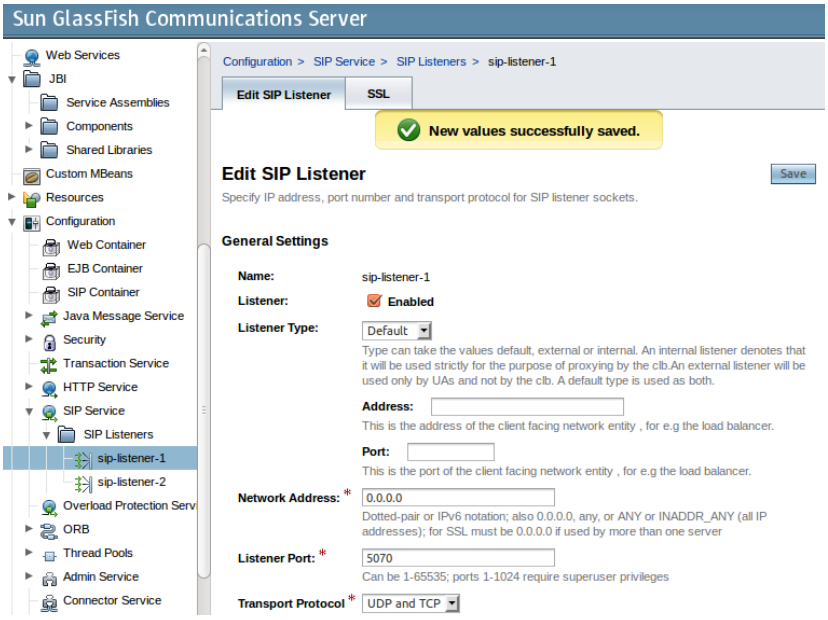

Figure 9

4. The default SIP port numbers in the SIP container have to be changed too. Therefore click on Configuration ➔ SIP Container, and then change the values under “SIP Port” and “Secure SIP Port” to 5070 and 5071 (see Figure 10). Finally save the changes

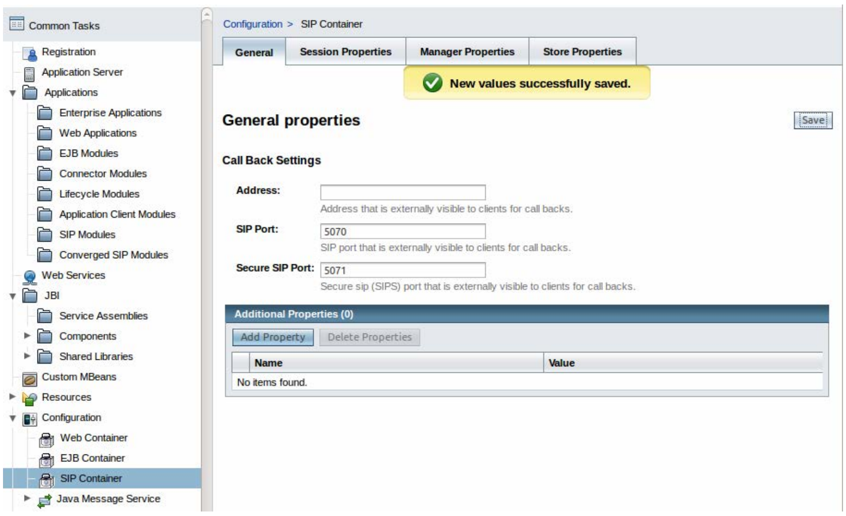

Figure 10

5. Now it is time to deploy the mcuWeb application. Under Application ➔ Converged SIP Modules click on "Deploy". Under Location click on "Browse" and select the mcuWeb.sar file, which is located under /usr/local/src/Medooze/mcuWeb/dist, and then click on "OK" (see Figure 11)

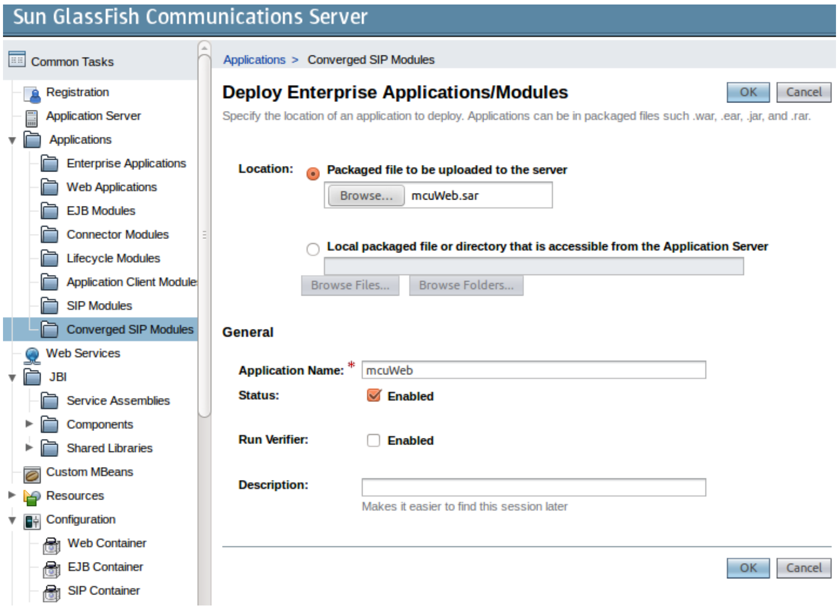

Figure 11

6. The deployed application can be found under Converged SIP Modules (see Figure 12). To run the application click on "Launch" or navigate with a web browser to the link http://ipaddress:8070/mcuWeb, where ipaddress is the IP address of Ubuntu

Figure 12

2.4 Deploying mcuWeb in JBoss Application Server

To deploy mcuWeb application in JBoss Application Server, do the following:

1. The first step is to download JBoss Application Server with Mobicents SIP Servlets from the hosting website

```
cd /usr/local/
```

```
sudo wget
http://sourceforge.net/projects/mobicents/files/Mobicents%20Sip%20Servlets/Mobicents%20Sip%20Servlets%202.0.0.FINAL/mss-2.0.0.FINAL-jboss-as-7.1.2.Final-1349104459.zip
```

```
sudo unzip mss-2.0.0.FINAL-jboss-as-7.1.2.Final-1349104459.zip
```

```
sudo mv mss-2.0.0.FINAL-jboss-as-7.1.2.Final-1349104459 jboss
```

2. In order to deploy the mcuWeb application in JBoss AS, all needed is to rename the mcuWeb.sar file to mcuWeb.war and then copy it to the auto deployment directory

```
cd /usr/local/src/medooze/mcuWeb/dist
```

```
sudo cp mcuWeb.sar /usr/local/jboss/standalone/deployments/mcuWeb.war
```

3. The next step is to configure the JBoss server by editing the configuration file “standalone-sip.xml”. The HTTP listening port number 8080 and SIP listening port numbers 5080 and 5081 can be used, for example, for the JBoss AS (see Figure 13)

```
cd /usr/local/jboss/standalone/configuration
```

```
sudo gedit standalone-sip.xml
```

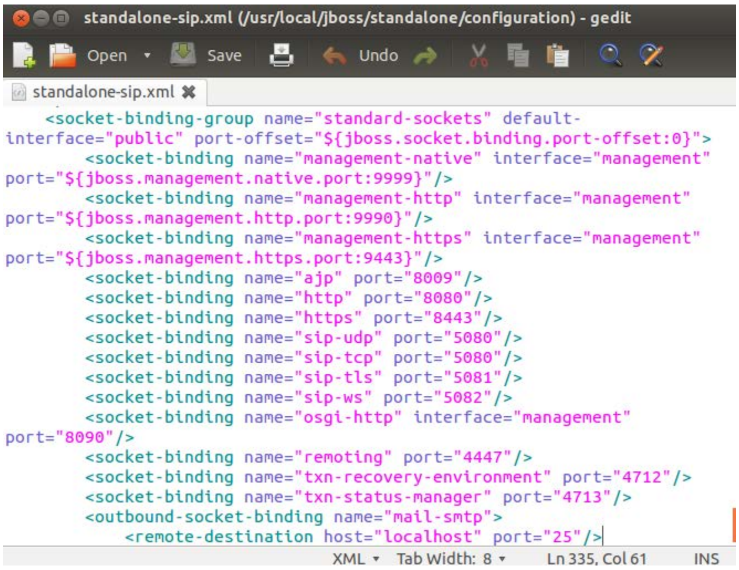

Figure 13

4. After editing the configuration file and saving the changes, run the JBoss server

```
cd /usr/local/jboss/bin/
```

```
sudo ./standalone.sh -c standalone-sip.xml -b ipaddress
```

Where ipaddress is the IP address of Ubuntu

5. Once the mcuWeb is deployed, the SIP Servlets routing routine has to be changed to make the mcuWeb application handle the SIP signalling. This is done via the Mobicents SIP Servlets management console available at http://ipaddress:8080/sip-servlets-management. Under INVITE choose the application “mcuWeb” and save the changes (see Figure 14)

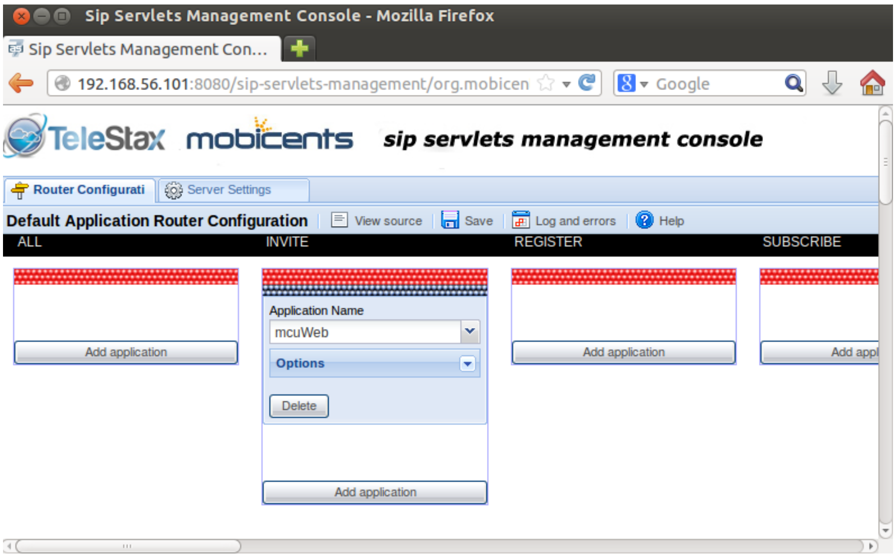

Figure 14

6. To configure the mcuWeb application navigate with a web browser to the link ipaddress:8080/mcuWeb, where ipaddress is the IP address of Ubuntu

2.5 Deploying mcuWeb in Tomcat Servlet/JSP Container

To deploy mcuWeb application in Tomcat Servlet Container, follow the description below:

1. The first step is to download Tomcat Servlet Container with Mobicents SIP Servlets from the hosting website

```
cd /usr/local/
```

```
sudo wget
http://sourceforge.net/projects/mobicents/files/Mobicents%20Sip%20Servlets/Mobicents%20Sip%20Servlets%202.0.0.FINAL/mss-2.0.0.FINAL-apache-tomcat-7.0.29-1210011535.zip
```

```
unzip mss-2.0.0.FINAL-apache-tomcat-7.0.29-1210011535.zip
```

```
mv mss-2.0.0.FINAL-apache-tomcat-7.0.29-1210011535 tomcat
```

2. In order to deploy the mcuWeb application in Tomcat Servlet Container, all needed is to rename the mcuWeb.sar file to mcuWeb.war and then copy it to the auto deployment directory

```
cd /usr/local/src/medooze/mcuWeb/dist
```

```
sudo cp mcuWeb.sar /usr/local/tomcat/webapps/mcuWeb.war
```

3. The next step is to configure the Tomcat server by editing the configuration file “server.xml”. The HTTP listening port number 8090 and SIP listening port numbers 5090 and 5091 can be used (for example) for the Tomcat server (see Figure 15)

```
cd /usr/local/tomcat/conf
```

```
sudo gedit server.xml
```

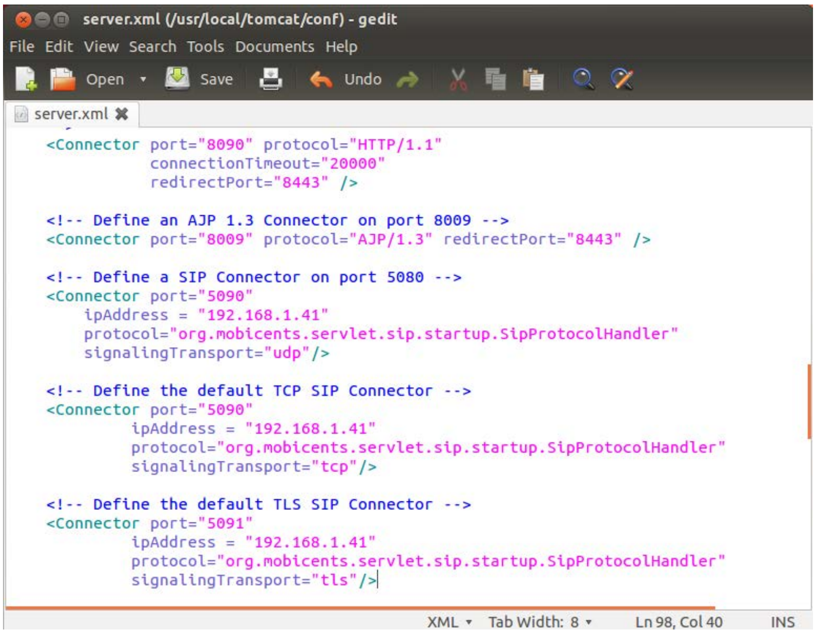

Figure 15

4. After editing the configuration file and saving the changes, run the Tomcat server

   ```
   cd /usr/local/tomcat/bin/
   sudo ./catalina.sh run
   ```

5. Once the mcuWeb is deployed, the SIP Servlets routing routine has to be changed to make the mcuWeb application handle the SIP signalling. This is done via the Mobicents SIP Servlets management console available at http://ipaddress:8090/sip-servlets-management. Under INVITE choose the application "mcuWeb" and save the changes (see Figure 14)

6. To configure the mcuWeb application navigate with a web browser to the link ipaddress:8090/mcuWeb, where ipaddress is the IP address of Ubuntu

2.6 Configuring media mixers

After deploying and running the mcuWeb application, navigate with a web browser to the link http://ipaddress:port/mcuWeb, where ipaddress is the IP address of Ubuntu and port is the HTTP listening port number used by the corresponding Application Server. The home page of the mcuWeb application will be displayed (see Figure 16).

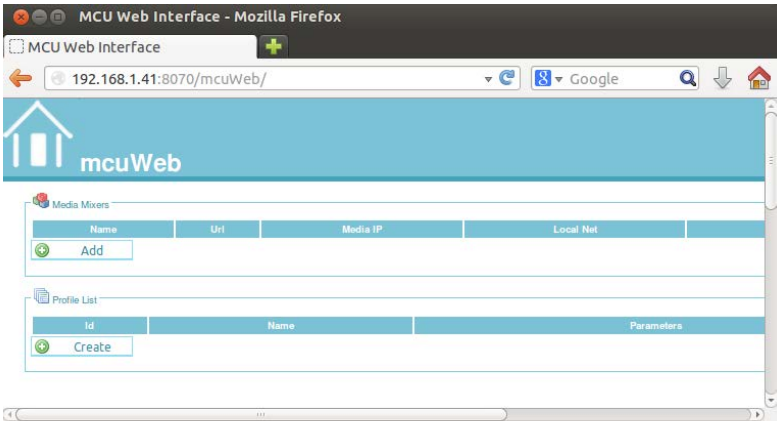

Figure 16

To create and configure a media mixer for videoconferences, click on the "Add" button under Media Mixers section. In the Media Mixer Data web page enter the following parameters and then click on the "Create" button (see Figure 17):

- Name: The name of the media mixer
- Url: Uniform Resource Locator (URL) of the MCU Media Server, which is used to send and receive XML-RPC requests and responses. If the Media Server and the Application Server are installed on same host, then you can enter http://127.0.0.1:port, where port is the HTTP XML-RPC API listening port number used by the Media Server, which can be changed via the --http-port argument
- Media Ip: The IP address of the Media Server that is used to transfer data (video, voice and text). This IP address will be included in the SDP answer sent form the Application Server to the SIP UA. Therefore it should be the IP address of the Media Server facing the Application Server
- Public Ip: The public IP address of the Media Server, which is used to create the RTMP (Real Time Messaging Protocol) URLs for flash streaming
- Local Net: This parameter is used to check whether the SIP UA is behind NAT or not. A SIP UA is behind NAT if the media IP in the SDP request is private and the IP address of the UA does not belong to the local net. The default value is 0.0.0.0/0, which means that all participants are not behind a NAT

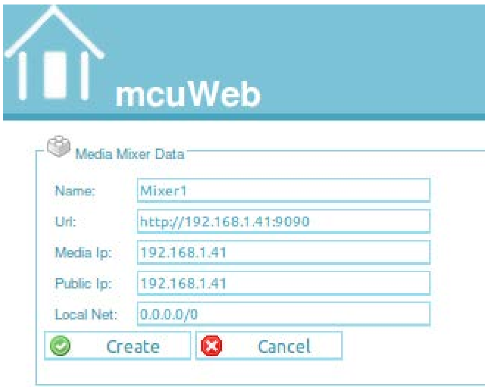

Figure 17

2.7 Adding video profiles

Before creating a video conference, it is important to define its features such as the used video resolution and the frame rates. This is done by adding video profiles, which define the quality of videoconferences. To add a video profile, click on the “Create” button under Profile List section and enter the following parameters (see Figure 18):

- Id: The identifier of the video profile
- Name: The name of the video profile
- Size: This parameter represents the video format resolution. Table 2 describes all supported video resolutions
- Video bitrate: The number of bits sent per second. Here the used unit is kbit/s
- Video FPS: Frame Per Second (FPS) or frame frequency, which represents the number of video images produced within a second
- Intra Period: Intra refresh period between video frames

Figure 18

Table 2

Format	Video resolution (width × height) in pixels
QCIF	176 x 144
CIF	352 × 288
VGA	640 x 480
PAL	678 x 576
HVGA	480 x 320
QVGA	320 x 240
HD720P	1280 x 720
WQVGA	400 x 240
W448P	768 x 448
448P	576 x 448
w288P	512 x 288
w576	1024 x 576
4CIF	704 × 576
4SIF	704 x 480
XGA	1024 x 768

After entering all these parameters, click on the “Create” button. The created video profile can be seen then on the mcuWeb home page.

2.8 Creating and managing videoconferences

To create a videoconference, click on the “Create” button under the Conference List section, and then enter the following parameters in the shown Conference web page (see Figure 19):

- Name: The name of the videoconference
- DID: This parameter will be used to match an incoming SIP INVITE request with the corresponding conference. If the SIP username part matches the DID parameter, the participant will be connected to the conference
- MediaMixer: The name of the media mixer configured in section 2.6
- Composition: This parameter defines the number of participants in a videoconference. It defines as well the view of each participant on that conference
- VAD: To enable or disable the VAD functionality (None or Full)
- Mosaic size: Video resolution of each participant (see Table 2)
- Default profile: The used video profile for the conference (see section 2.7)

- Override Audio, Video and Text Codecs: These parameters can be used to define a specific audio, video or text codec to be used in the conference. If these parameters are not used, then the used codecs will be determined during the SIP signalling

Figure 19

After entering the above mentioned conference parameters, click on the “Create” button. As soon as a videoconference has been created, the conference web page will be displayed, which enables the user or administrator to manage the conference. For example the number of participants, the video resolution and the used video profile can be changed during a conference session by changing the corresponding parameters under the Conference section (see Figure 20).

Additionally participants can be invited to a conference by entering the corresponding SIP URI under the Add participants section, and then by clicking on the “Invite” button (see Figure 21).

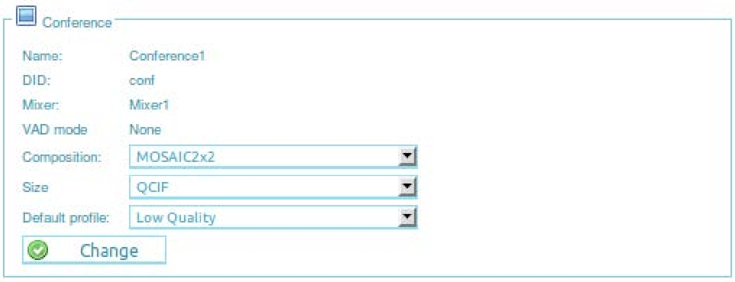

Figure 20

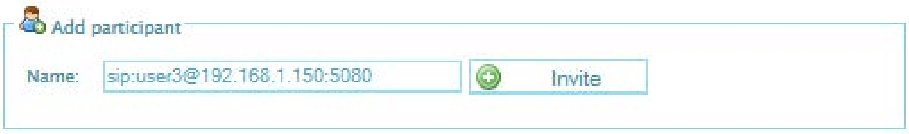

Figure 21

The Mosaic section allows the user to change the position of each participant (see Figure 4.22).

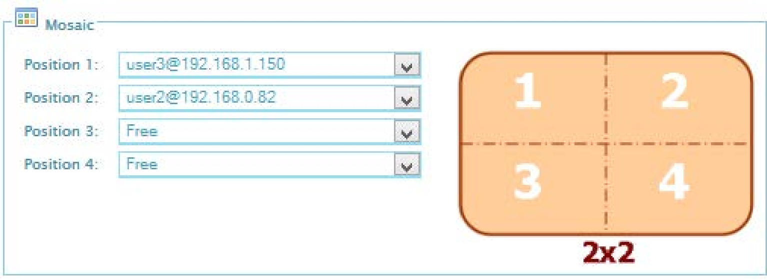

Figure 22

The Participant List section offers the ability of enabling/disabling audio and/or video by a participant. It allows as well the user to change the video profile of each participant. Finally a participant can be removed from a videoconference by clicking on the corresponding “Remove from conference” button (see Figure 23).

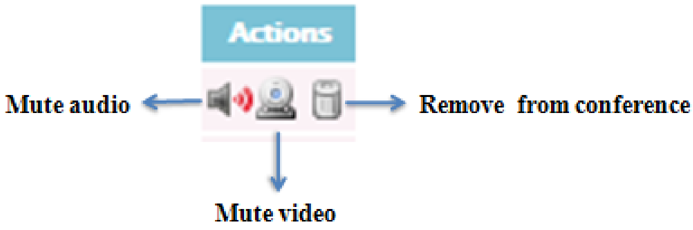

Figure 23

The Conference List section on the mcuWeb home page offers the following configuration options (see Figure 24):

- Remove conference: To remove an existing videoconference
- Conference detail: To change to the managing page of the videoconference
- Watch conference: To watch an existing videoconference on a web browser using flash streaming (see Figure 25)

Figure 24

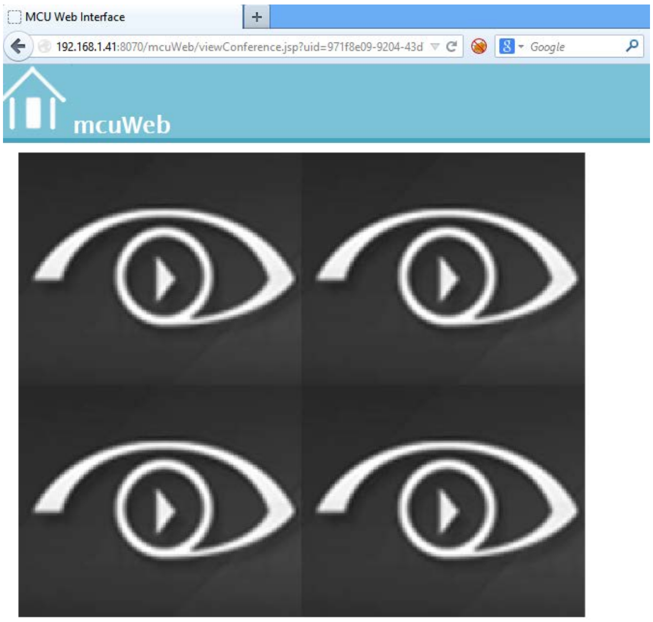

Figure 25

2.9 Joining a videoconference

To join a videoconference using a SIP UA (for example Linphone), enter the SIP URI of the videoconference in the field “SIP address or phone number” and then click on the “Call” button (see Figure 26).

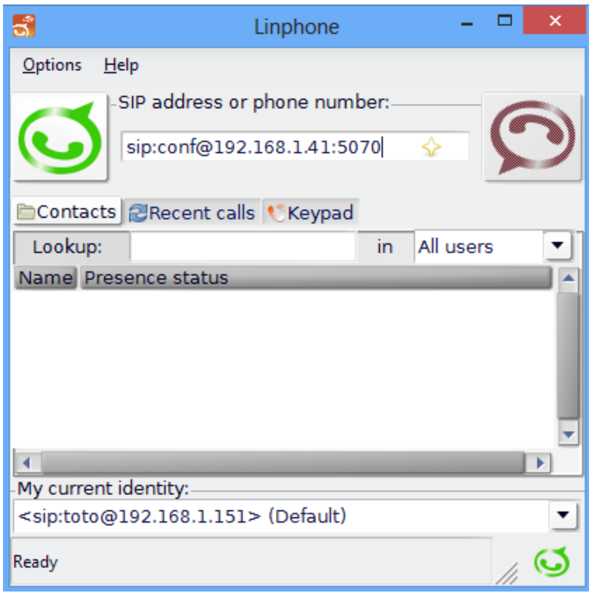

Figure 26

According to Figure 26 the SIP URI of a videoconference consists of the following parts:

- conf: The SIP username part that should match a conference DID parameter (see Figure 19)
- 192.168.1.41: The IP address of the Application Server, on which the mcuWeb application is running
- 5070: The SIP listening port number of the Application Server

2.10 Recording videoconferences

To enable the recording functionality, create a folder with the name “recordings” under the directory /var (see Figure 27). As soon as a videoconference is deleted, the conference will be saved in the recordings folder in the format FLV (Flash Video).

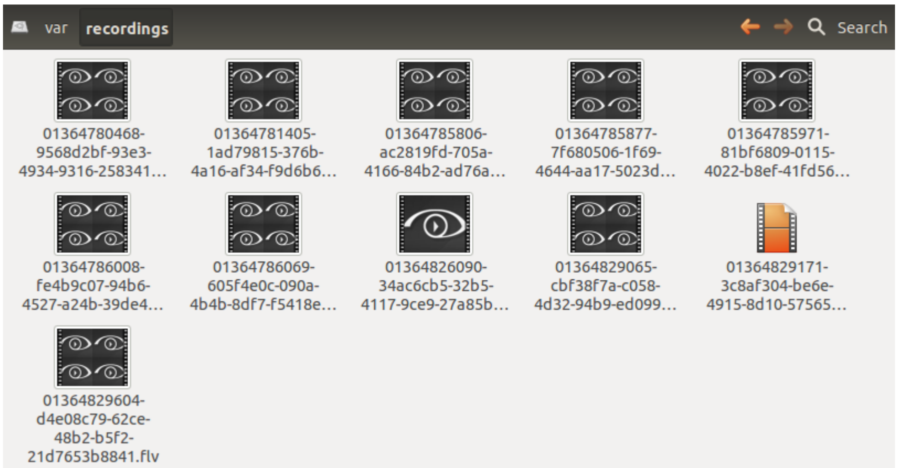

Figure 27

2.11 Adding conference templates

Ad-hoc templates allow the user to create videoconferences automatically with predefined parameters. When the first participant dials in and matches a DID parameter from a template, a videoconference is automatically created with the preconfigured parameters and the participant is joined to it. To create an ad-hoc template click on “Create” under the AdHoc Conference Templates section. The parameters of ad-hoc templates are similar to those used to create conferences (see Figure 28, see section 2.8). After entering the corresponding parameters click on the “Create” button.

Figure 28

2.12 Flash streaming and broadcasting

The media server supports as well live broadcasting and flash streaming using the RTMP protocol. To enable the broadcasting functionality, a live streaming channel hast to be created first on the Application Server using the mcuWeb application. Furthermore a media live encoder is required to encode audio and video in real time to the media server. The Adobe Flash Media Live Encoder 3.2 can be used. The software can be downloaded from the website http://www.adobe.com/.

To create a broadcasting live stream channel, click on “Create” under the Broadcast List section, and then enter the following parameters in the shown Broadcast web page (see Figure 29):

- Name: The name of the broadcasting live stream
- Tag: The identifier of the live streaming channel
- MediaMixer: The name of the media mixer configured in section 2.6

After entering the mentioned broadcasting parameters, click on the “Create” button.

Figure 29

To start live broadcasting and flash streaming, run the software Adobe Flash Media Live Encoder. As shown in Figure 30, the software offers many configuration options. It supports both VP6 and H.264 video codecs. The software allows the user to set the used video and audio codecs, the frame rate, the sample rate, the used bandwidth and finally the input and output video resolutions.

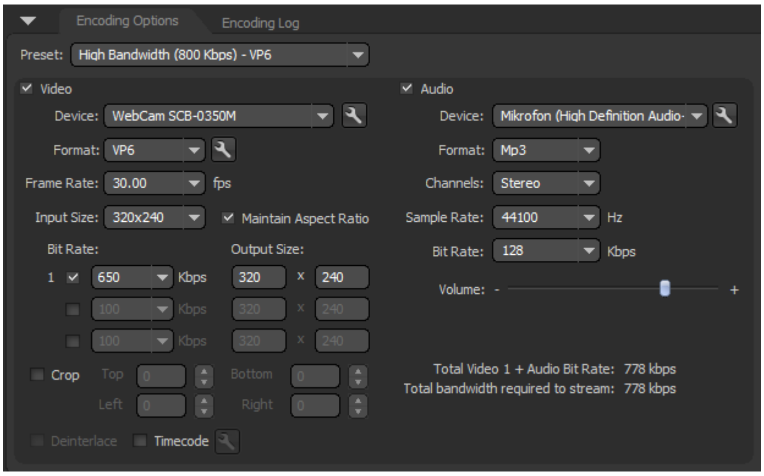

Figure 30

Under the Panel Options "Output" enable the option "Stream to Flash Media Server", and then enter the following parameters (see Figure 31):

- Under "FMS URL:" enter the URL of the Flash Video Streaming Service provided by the media server. The URL has to be entered in the following form:

  rtmp://ipaddress:port/broadcaster/publisher/identifier

  Where ipaddress is the IP address of the media server, port is the port number used for flash streaming (default is 1935), identifier is the identifier of the live streaming channel (Tag parameter)

- Under "Stream" enter the identifier of the live streaming channel (Tag parameter)
- Optionally you can enable and configure the auto adjustment options
- To save the live stream on your computer, enable the option "Save to File" and enter the name of the video, where the extension file has to be FLV (Flash Video)

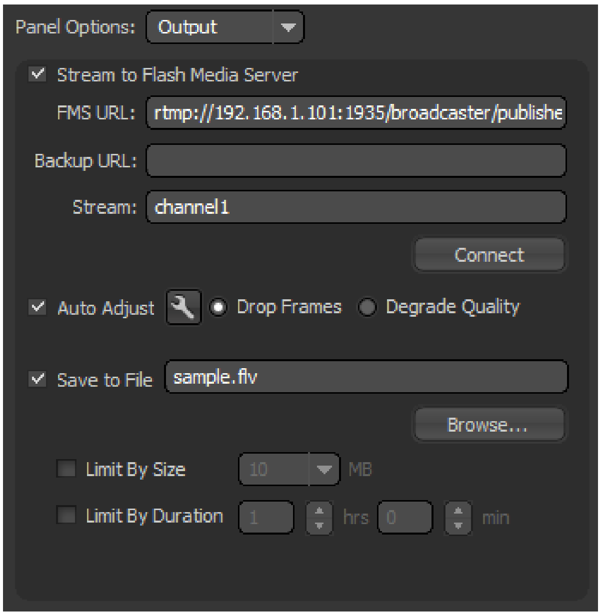

Figure 31

Finally to start video and audio encoding, click on the “Connect” button and then on the “Start” button. To view the live stream on the web, open a web browser and navigate to the mcuWeb home page, and then click on “Watch broadcast” under the Broadcast List section (see Figure 32).

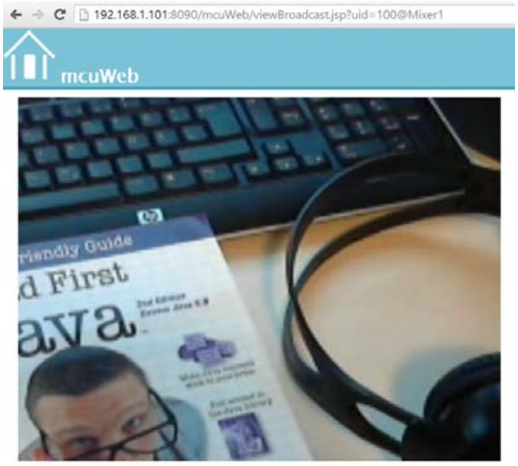

Figure 32