

## 10 Remote monitoring interface

The “Streaming Client” firmware supports two different ways of remote monitoring: SNMP and Barimon.

SNMP can send a trap on start-up and when switching the stream and can be requested at any time. Barimon sends periodic reporting as well as information on request. The features of these protocols are described in the following chapters.

### 10.1 Barimon Remote Monitoring

The device can be monitored using Barimon technology. Please visit <http://www.barimon.net> for detailed information and to create your own free Barimon project.

#### Barimon periodic report

With Barimon, the device's streaming status is sent actively by the device to the pre-configured server via HTTP. The report is sent in regular time intervals (configurable), at stream stop and at stream start (that means, when a stream switch happens, two reports are sent).

The Barimon server has to be specified in the configuration field “BARIMON URL”. Only the HTTP protocol is supported, including all its options and possibility of using the HTTP Proxy. The “BARIMON URL” syntax is:

**http://[<name>:<password>@]<address>/** (name and password can be omitted, e.g.

**http://www.barimon.net/**)

The frequency of Barimon reporting can be set in the configuration field “BARIMON Report Period” in minutes. The complete report will be sent in this defined time interval.

The following table show the content of the periodic report.

Value	Type	Description
<b>mac</b>	String (e.g.: <b>0008E1003D90</b> )	Devices MAC address used for Barimon “sensor” identification
<b>alarm</b>	Boolean: “ <b>true</b> ” or “ <b>false</b> ”	Alarm trigger: “ <b>true</b> ” on stream change, otherwise “ <b>false</b> ”
<b>Bitrate</b>	8bit unsigned decimal number	Bitrate of the played stream in kilobits per second
<b>BufferLevel</b>	16bit unsigned decimal number	Amount of bytes in the buffer
<b>Error</b>	8bit unsigned decimal number	Number of last error (see the definition in the <a href="#">File “BARIXAUDIOSNMP.MIB”</a> )
<b>FrameDrop</b>	32bit unsigned decimal number	Number of RTP frames dropped to correct long term clock drift. Resets with every RTP stream (reconnect or new sequence of frames).
<b>FrameDup</b>	32bit unsigned decimal number	Number of RTP frames duplicated to correct long term clock drift. Resets with every RTP stream (reconnect or new sequence of frames).
<b>FrameLoss</b>	32bit unsigned decimal number	Number of RTP frames lost on the network since the stream start. Resets with every RTP stream (reconnect or new sequence of frames).
<b>Latency</b>	16bit unsigned decimal number	Average latency of the RTP decoder; valid only for RTP streams.
<b>Reconnects</b>	16bit unsigned decimal number	Amount of reconnects due to loss of the stream source
<b>SoftErrorCount</b>	16bit unsigned decimal number	Amount of stream drop-outs (missed more than 5 frames in a row)

Value	Type	Description
<b>StreamNumber</b>	String (0,1,2,3 or "prio" for priority)	Number of played stream
<b>UpTime</b>	16bit unsigned decimal number	Up time of the device in seconds (since last reboot)
<b>URL</b>	String	URL of the currently played stream
<b>Volume</b>	8bit unsigned decimal number	Current volume level in percent

### Requesting Barimon report over UDP

Furthermore, the actual status of the device can be requested over UDP. The Port number used must be the same as specified in the configuration field "UDP Reporting Port". Setting the Port to 0 disables this function.

Sending a UDP datagram with the payload "MTELL\r\n", i.e. 7 bytes: **0x4D, 0x54, 0x45, 0x4C, 0x4C, 0x0D 0x0A**, will result in a UDP reply sent on same port to the IP address the request originated from. The reply is comma separated and contains no spaces and no line feeds (the table below is word wrapped).

Example content of the UDP reply
<b>BufferLevel=10528,Latency=598,FrameLoss=0,FrameDup=0,FrameDrop=0,SoftErrorCount=0,StreamNumber=1,Bitrate=192,Reconnects=2,Error=11,Volume=25,UpTime=25,URL=rtp://0.0.0.0:4444/</b>

To test this we recommend the free PC software called "UDP Test Tool" from <http://www.simplecomtools.com>.

## 10.2 Own Monitoring Server using Barimon protocol

To run an own monitoring server you will need to write your own scripts depending on the server architecture and OS (PHP, ASP...).

The script has to be named "submit" and should be available in the folder "/sensors/data" as the "Streaming Client" firmware sends an HTTP GET request for "sensors/data/submit?..." to that server. This path is fixed and can not be changed. The information is included after the questions mark.

**GET sensors/data/submit?mac=<mac address>&alarm=false&info=<info> HTTP/1.0**

<mac address> is 12 hex character string without any delimiters: XXXXXXXXXXXXX e.g.: **0008E1003D90**

<info> is a string in the format: **BufferLevel=<int>,FrameLoss=<long int>,SoftErrorCount=<long int>,StreamNumber=<string>,URL=<string>,Bitrate=<int>,Reconnects=<long int>,Error=<int>,Volume=<int>,UpTime=<long int>**

<int> is an 8bit integer decimal number, <long int> is an 16bit integer decimal number, <string> is a string of characters

### Configuration Parameters for Barimon periodic report

The Barimon server has to be specified in the configuration field "BARIMON URL". Only the HTTP protocol is supported, including all its options and possibility of using the HTTP Proxy. The "BARIMON URL" syntax is:

**http://[<name>:<password>@]<address>/** (name and password can be omitted e.g. **http://www.myserver.com/**)

The frequency of Barimon reporting can be set in the configuration field "BARIMON Report Period" in minutes. The complete report will be sent in this defined time interval.

### Example "submit.php"

The submit PHP script can read the variables from the \$\_GET array e.g.:

```
$mac=$_GET["mac"]; // here you can check if the MAC address is registered in your database and decide to accept/ignore this request
```

```
$alarm=$_GET["alarm"]; // a is this an alarm ?
```

```
$info=$_GET["info"]; // comma separated list of "measured values"
```

The \$info variable will contain complete device info which is the string as described in the section above.

The GET variable handling is all standard, there's nothing "Barimon specific", you can access the variables as in any other web CGI script.

## 10.3 SNMP Remote Monitoring

The "Streaming Client" firmware supports SNMPv1 (Simple Network Management Protocol Version 1) which uses UDP for the transfer of information.

### SNMP trap sending

The IP address of the receiver of SNMP traps has to be specified in the configuration field "SNMP Trap Receiver". If set to 0.0.0.0 then no traps are sent. Traps are sent on UDP port 162 to the specified receiver.

The following traps are supported:

- **cold start** – sent at startup
- **private trap** – sent at stream stop or at stream start. System time and the stream number are sent in the trap.

### SNMP querying

The device can be queried using the SNMPv1 protocol on UDP port 161, the MIB (Management Information Base) version supported is 2.

The MIB file **BARIXAUDIOSNMP.MIB** is included in the "Streaming Client Update Kit" and can be found in the folder **update\_rescue**.

See the following print out of the MIB file for capabilities.

### File "BARIXAUDIOSNMP.MIB"

Content of the BARIXAUDIO.MIB file contained in the "Streaming Client Update Kit"

```
-- The Barix Audio MIB leaf  
-- The Barix MIB Registration Authority is barix.mib  
-- Version: 2.2  
-- Date: 07 March, 2006  
-- Copyright (c) 2004-2006 Barix AG  
  
-- Changes:  
-- 20050503 KPS Updated according to Barix MIB registration authority  
-- 20060116 KS/PK Added streaming variables  
-- 20060307 KS unit net, hostname added  
-- 20060307 KS instreaming levels added
```

```
BARIXAUDIOSNMP-MIB DEFINITIONS ::= BEGIN
```